ESG Certification Recommendation System

Georgia Institute of Technology

Applied Analytics Practicum

Team 2

November 23, 2023

Client

Stibo Systems

Team Members

Jackson Duke*

Sofia Laval*

Nathan Ruiz*

**Equal contribution by team members.*

ABSTRACT

Ecolabels play a pivotal role in guiding consumers towards sustainable purchasing decisions amid the increasing importance of environmental, social, and governance (ESG) factors. The current manual process of certifying products for certifications is both time-consuming and error-prone, posing risks of incorrect certifications and product recalls. In this paper, we will propose an innovative approach leveraging Large Language Models (LLMs) to streamline and enhance the certification process. We will compare the LLM assessments with traditional Machine Learning (ML) classification algorithms to evaluate the efficacy of these models as an Ecolabel recommendation engine.

LLMs have become increasingly popular for their ability to analyze large amounts of data while incorporating world knowledge and reasoning into language processing. Our results indicate that this unprecedented technology is suitable for assessing well-defined Ecolabel mandates. The LLMs demonstrate the ability to calculate metrics accurately while providing detailed explanations and transparent conclusions. While the ML algorithms classified the certifications with higher accuracy, they required extensive data preparation and are unable to provide reasoning behind the classification decisions.

By addressing the limitations and strengths of each approach, this research contributes to the ongoing discourse surrounding data science in ESG. The findings offer a foundation for future research aiming to strike a balance between the efficiency of automation and the precision required for reliable ESG certification.

1 CLIENT INTRODUCTION

Stibo Systems offers Master Data Management (MDM) solutions to centralize and manage critical data shared across business departments. Examples of master data types include customer, product, and sustainability data. By ensuring data consistency and transparency across an enterprise, this single platform approach unlocks business insights and enables efficient operations.

2 PROBLEM STATEMENT

Environmental, social, and governance (ESG) issues have become increasingly attached to purchasing decisions, with "78 percent of US consumers [stating] that a sustainable lifestyle is important to them."[1] This provides data companies with a unique opportunity to assist enterprises as they seek to build products sustainably and explains why the ESG software market is estimated to reach $2B in size by 2030[2]. One method for assessing products for sustainability is through Ecolabels, which are physical stickers or logos on a product indicating the adherence to an environmental standard or criteria.

A key challenge for sustainability master data lies in certifying and assessing products for these Ecolabels. The current process relies on sustainability experts who manually identify products that meet the requirements for ESG certifications using various Excel spreadsheets. This approach is not only time consuming, but error-prone, and can result in product recalls due to incorrect certification. There is also a great deal of inconsistency across ESG certification data. Product data can be incorrect or incomplete, and certification criteria can be vague or confusing. In addition, certifications are only valid for a limited time, and standards can be updated based on new environmental research, rendering past assessments and product data unusable. Our client is well suited to offer a system to verify these certifications, enabling enterprise ESG initiatives and becoming a forerunner in sustainability master data.

In this project, we aim to improve upon the manual process by leveraging Large Language Models (LLMs). LLMs are gaining popularity for the ability to process and analyze large amounts of data. Given product and certification data, these models use Natural Language Processing (NLP) methods to assess whether a product meets the criteria for an ESG certification (**Figure 1**).

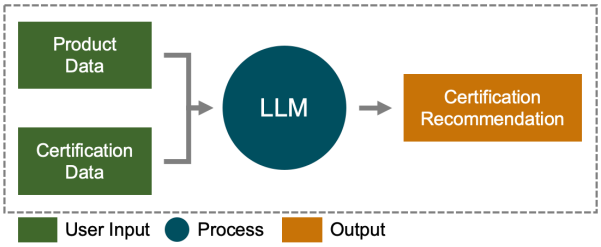

Figure 1. High-level recommendation engine architecture.

The world of LLMs is relatively new, so to assess the viability of the LLM assessments as an Ecolabel recommendation engine, we will compare this method with traditional Machine Learning (ML) classification algorithms.

[1] (2023, February 6). Consumers care about sustainability and back it up with their wallets. McKinsey & Company. Retrieved from https://mckinsey.com/industries/consumer-packaged-goods/our-insights/consumers-care-about-sustainability-and-back-it-up-with-their-wallets

[2] (2023, September 4). ESG reporting software market size to reach USD 1,215.9 million by 2028. Yahoo Finance. Retrieved from https://finance.yahoo.com/news/esg-reporting-software-market-size-125400262.html

3 EXPLORATORY DATA ANALYSIS

In this section, we will explore the two driving datasets for the recommendation engine.

3.1 Product Data

The product data originally comes from IceCat, a website where electronic product content data is stored across many different product categories. The diagram below (**Figure 2**) shows the hierarchal structure of the product data.

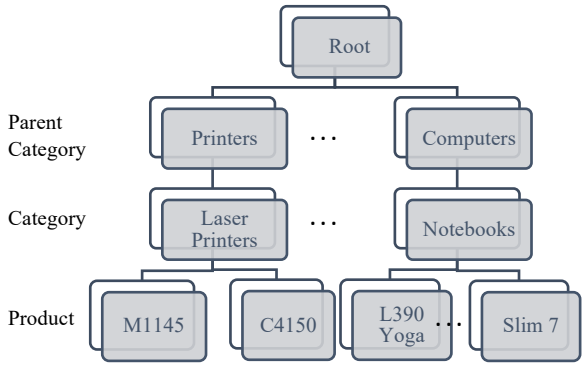

Figure 2. Product data hierarchy.

After exploring all 495 IceCat datasets provided, we selected the notebooks (laptops) dataset for our study, as this product category contained the most products and features from the 495 files: 233,669 notebook products (rows) and 2,462 features (columns). Below is a snippet of a Slim 7 notebook and its features:

Product Name	Processor Manufacturer	Processor Cache	…	Sustainability Certificates
Slim 7	Intel	3.0 MB	…	Energy Star

Table 1. Example notebook product.

The "Sustainability Certificates" column shows the ESG certification with which the product is compliant. We found that the certified product lists found on the Energy Star and TCO websites are not reliable, as certifications are for a fixed period, and only current generation products are listed. To proceed, we made the following informed assumptions:

Assumption 1: The "Sustainability Certificates" (SC) column captures the certifications at the same point in time as the product features.

Assumption 2: The SC column is reasonably well defined to delineate the products (i.e., SC = "Energy Star" indicates the product is not certified in TCO).

While **Assumption 2** is not as well-founded, this column is still the best candidate to be the dependent variable for this study. Removing duplicate rows and filtering for this column resulted in 110,225 products. The following table shows the different values and counts for the "Sustainability Certificates" column:

Sustainability Certificates	Row Count
RoHS	49,470
Energy Star	39,991
EPEAT	20,480
TCO	173
Other	111
Total	**110,225**

Table 2. "Sustainability Certificates" ("y" variable) counts.

35% of the columns were fully blank and removed from the dataset. We chose to exclude columns that had greater than 95% missing values, resulting in 346 features. The remaining columns contained 54% missing values and removing all rows with at least one missing value resulted in an empty dataset. Our final dataset consisted of 110,225 products with 346 features.

3.2 Certification Data

After researching different ESG certifications, we selected Energy Star and TCO for our initial study, as they are two of the most popular certifications for electronics and have specific criteria for notebooks. We then scraped the most recent requirement PDFs for mandates, which range from products to supply chain requirements, resulting in a final list for the LLMs to assess:

Certification	Total Mandates	Product Mandates	Avg. Characters in Mandate
Energy Star	22	**9**	391
TCO	41	**17**	310

Table 3. Summary of certification mandates.

We narrowed our scope to 9 and 17 product-focused mandates for Energy Star and TCO, respectively.

In the future, it would be ideal to automate this scraping process, so a product manager or data analyst would only need to upload a PDF of the certification requirements, and our tool would parse that document for the mandates. However, we opted to perform this task manually to save time and ensure accuracy. Below is an example mandate for TCO (**Figure 3**). The full requirement documents for Energy Star and TCO can be found in **Appendix A.1**.

TCO Mandate 5.2.1
Display Resolution

The display panel should have a pixel density of at least 100 PPI:

$$\text{PPI} = \frac{\sqrt{horizontal\ pixels^2 + vertical\ pixels^2}}{diagonal\ of\ the\ panel\ in\ inches}$$

Figure 3. Example certification mandate.

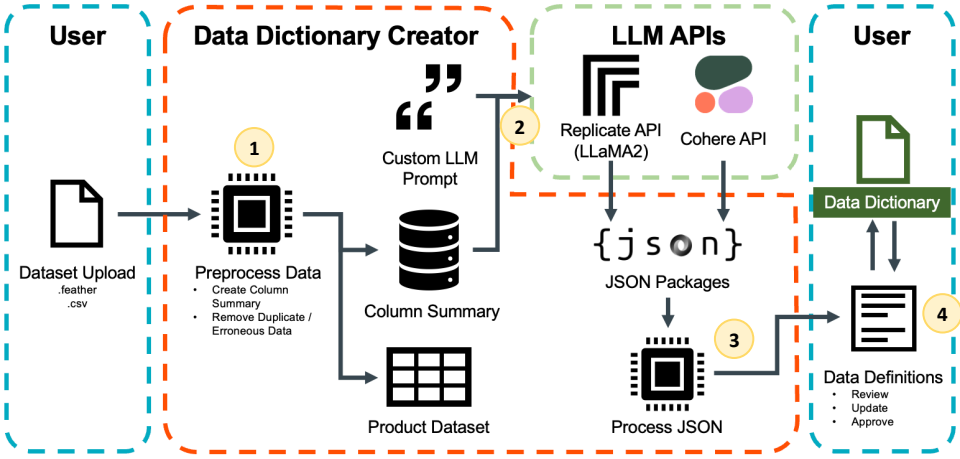

Figure 4. Architecture for the Data Dictionary Creator module in the application. (1) User uploads a product dataset, and the application removes duplicates and columns missing values for greater than 95% of rows. A column summary is created for the min, max, unit, and common values in each column. (2) The application queries Replicate (LLaMA2) and Cohere APIs to generate a definition for each column using the column summary. (3) The application processes the JSON output from the LLMs and displays the definitions to the user. (4) The user reviews, updates, and approves the best definition for storing in the Data Dictionary.

4 METHODOLOGY

This section will describe the methods for assessing ESG certification eligibility using LLMs as well as ML algorithms, which will be used as the baseline for comparison. This can be separated into three buckets:

1. Data Enrichment
2. LLM Querying
3. ML Classification

4.1 Data Enrichment

After the initial exploration of the dataset, we found that there were no column definitions associated to the product features. Column definitions allow us to take a certification mandate and rank product attributes by semantic meaning, thereby providing the LLMs with only relevant product information when assessing compliance. Thus, the first phase of this project was building an application to create definitions for the 346 product attributes (**Figure 4**).

This application was built using Streamlit, an open-source Python library for ML web applications. The end user uploads the product dataset (in our case, the notebooks dataset), and the application preprocesses the data and creates a summary for each column:

Name	Values	Unit	Min	Max
Processor Cache	'8.0', '6.0', '12.0', …	MB	1	12

Table 4. Layout of the column summary.

Using this column summary, the application queries the Replicate (LLaMA2) and Cohere APIs to generate a definition for the column. The user can then review, update, and approve the output that best describes the column. As an example, for the "Processor Cache" column:

> **Cohere Definition:**
> The size of the processor cache measured in MB.
>
> **LLaMA2 Definition:**
> The processor cache is a crucial component of a notebook's performance. It is memory located inside the processor that stores frequently used data or instructions. Processor cache sizes can vary, with larger caches providing better performance. This column contains the processor cache size in megabytes (MB).

Figure 5. LLM generated definitions for the column "Processor Cache" from the Data Dictionary Creator.

We chose the LLaMA2 response as the definition, as it captured the purpose of a processor cache in a laptop.

Repeating this process for the 346 columns resulted in the complete file of column definitions for the product dataset, called a data dictionary.

A demo of the data dictionary creator process can be found in **Appendix A.2.**

4.2 LLM Querying

With the data dictionary complete, the first step in the LLM process is mandate-column ranking, where each column definition is compared with each mandate description to rank the columns by relevance. We tested BERT and TF-IDF encodings to see which produced better similarity scores (defined in **Appendix A.3**):

Encoder	Min	Q1	Mean	Q3	Max
BERT	0.19	0.68	**0.73**	0.77	0.91
TF-IDF	0.00	0.01	**0.03**	0.05	0.49

Table 5. Mandate-column similarity score distributions using BERT and TF-IDF encodings.

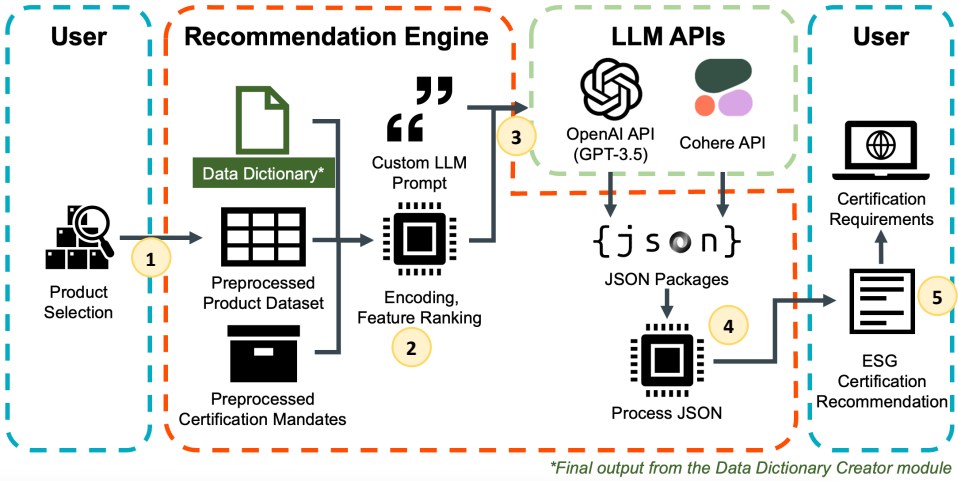

Figure 6. Architecture for the Recommendation Engine module in the application. (1) User selects a product to assess from a dropdown, and the application filters the dataset for that product's features. (2) The application loops through the certification mandates and prepares a payload for the LLM prompt. This process uses the mandate-column similarity ranking to search through the product features and select the five most relevant product attributes for each mandate. (3) The application loops through all mandates and queries the LLM APIs for an assessment on compliance. (4) The application parses the JSON output for the mandate recommendation. (5) The application displays the recommendation for each certification based on the LLM mandate assessments.

The BERT encodings produced much stronger scores, averaging a cosine similarity of 0.73, while TF-IDF averaged 0.03.

Proceeding with the column rankings obtained by BERT encodings, the application runs through all mandates (9 for Energy Star, 17 for TCO), and passes the mandate description along with the five most relevant product features to the LLMs, which output an assessment and reasoning:

LLM Response	Meaning
TRUE	The product is compliant with the mandate.
MORE INFO NEEDED	The provided attributes do not provide the necessary information to assess compliance.
FALSE	The product is not compliant with the mandate.

Table 6. LLM recommendations.

The percentage of mandates passed:

$$Percentage\ Passed = \frac{TRUE}{TRUE + FALSE}$$

serves as the final output or confidence that a product is eligible for certification. This, along with the LLM reasoning, is output to the user for review.

The architecture can be found in **Figure 6**, and a demo of the product recommendation engine process can be found in **Appendix A.2**.

4.3 ML Classification

We used six traditional machine learning (ML) classification algorithms:

1. Logistic Regression (Log. Reg.)
2. Support Vector Machine (SVM)
3. K-Nearest Neighbor (KNN)
4. Random Forest (Ran. For.)
5. XGBoost
6. AdaBoost

to predict whether a product adheres to the ESG certification requirements for Energy Star and TCO. After training each model (based on training data), we evaluated the model's performance on test data using a range of key performance metrics: accuracy, ROC, AUC, precision, recall, and specificity (as defined in **Appendix A.3**). Figure 7 below presents a flow chart of the model selection process.

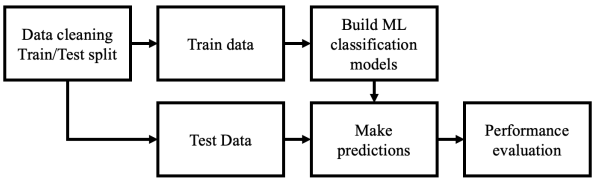

Figure 7. Model selection flow chart.

4.3.1 Data Cleaning

ML models such as logistic regression cannot handle missing or categorical data. To mitigate these limitations, we turned to label encoding and K-Nearest Neighbor imputation. Label encoding converts categorical data to numerical values, where each category in a categorical column is assigned a unique integer value. For each data point with missing values, KNN imputation identifies similar data points, typically found using a distance metric such as the Euclidean distance. It computes the weighted average of these

similar values and assigns this average as a replacement for the missing value.

After data preprocessing (**Section 3.1**), our dataset consisted of 110,225 rows and 346 columns. The data was randomly split into train and test sets, where the test sets contained 500 samples each, with 181 certified notebooks in the Energy Star set and 100 in TCO's. The tables below describe the final datasets used to train and test the ML models:

Dataset	Total Rows	Certified (1) Rows	Not Certified (0) Rows
ES Train	109,725	39,810	69,915
ES Test	500	181	319
TOTAL	**110,225**	**39,991**	**70,234**

Table 7. Energy star datasets.

Dataset	Total Rows	Certified (1) Rows	Not Certified (0) Rows
TCO Train	109,725	73	109,652
TCO Test	500	100	400
TOTAL	**110,225**	**173**	**110,052**

Table 8. TCO datasets.

5 MODEL EVALUATION

This section will present the testing results obtained by traditional ML models and LLMs.

5.1 Traditional ML Performance

Across the two selected sustainability certifications, six machine learning approaches were applied to the data. We measured their respective effectiveness in predicting whether a given product was Energy Star or TCO certified, allowing for prediction of outcomes with future products.

Below are the six models' performance on the respective 500-sample testing sets for each certification.

5.1.1 Energy Star:

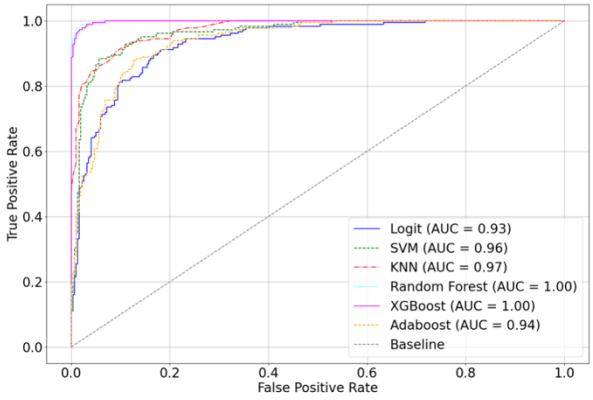

Figure 9. Receiver operating statistic (ROC) for six Energy Star classification methods.

Method	Accuracy	Precision	Recall	Specificity
Log. Reg.	85.2%	75.8%	86.7%	84.3%
SVM	90.2%	83.0%	91.7%	89.3%
KNN	91.2%	90.1%	85.1%	94.7%
Ran. For.	**98.0%**	**98.3%**	**96.1%**	**99.1%**
XGBoost	97.6%	**98.3%**	95.0%	**99.1%**
AdaBoost	86.6%	83.1%	79.0%	90.9%

Table 10. Performance metrics for six Energy Star classification methods.

Random Forest performed best across all performance metrics, misclassifying only 10 out of the 500 test products (98% accuracy). XGBoost also performed well (97.6% accuracy), while Logistic Regression and AdaBoost scored worse (85.2% and 86.6% accuracy, respectively). Detailed explanations of these metrics can be found in **Appendix A.4**.

5.1.2 TCO:

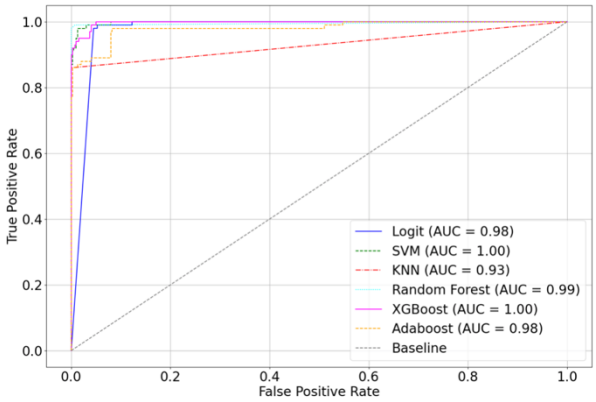

Figure 8. Receiver operating statistic (ROC) for six TCO classification methods.

Method	Accuracy	Precision	Recall	Specificity
Log. Reg.	95.6%	83.1%	**98.0%**	95.0%
SVM	**97.8%**	97.8%	91.0%	99.5%
KNN	91.4%	**100.0%**	57.0%	**100.0%**
Ran. For.	92.0%	**100.0%**	60.0%	**100.0%**
XGBoost	93.4%	**100.0%**	67.0%	**100.0%**
AdaBoost	92.8%	**100.0%**	64.0%	**100.0%**

Table 9. Performance metrics for six TCO classification methods.

SVM had the highest accuracy of 97.8%, along with a relatively high precision, recall, and specificity scores compared to the other models. KNN, Random Forest, XGBoost, and AdaBoost displayed lower recall scores, indicating difficulty in accurately predicting positive classes due to the imbalanced nature of the dataset. Despite efforts to adjust the models to account for the imbalance, the issue remained challenging because only 0.07% of the training data had a TCO certification.

5.2 LLM Performance

On average, it took Cohere 1m56s to assess a product for Energy Star, and 3m39s to assess a product for TCO. For GPT-3.5, it took 1m38s and 3m5s for Energy Star and TCO, respectively (**Table 7**). Cohere is free up to 1000 queries per month, while GPT-3.5 is priced per token. Using GPT-3.5 to assess one product for both Energy Star and TCO would cost just under $0.02.

Model	Certification	Time (s)	Cost
GPT-3.5	Energy Star	98	$0.006
GPT-3.5	TCO	185	$0.011
Cohere	Energy Star	116	Free*
Cohere	TCO	219	Free*
TOTAL	**389**	**205**	**184**

Table 11. Time and cost for Cohere and GPT-3.5 by certification.

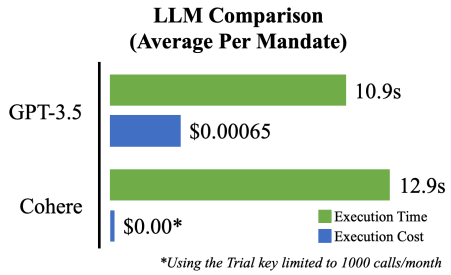

Figure 10. Time and cost for Cohere and GPT-3.5 per mandate.

Due to budget and time constraints, we could not run the LLM process on the complete testing datasets. However, we did run the process for 389 of the 1000 test products. There was no need to perform KNN imputation or label encoding, as the LLMs need only the original data from IceCat to assess compliance.

Dataset	Total Rows	Certified (1) Rows	Not Certified (0) Rows
ES Test	228	114	114
TCO Test	161	91	70
TOTAL	**389**	**205**	**184**

Table 12. Dataset breakdown for LLM testing.

5.2.1 Energy Star:

Using the ES Test dataset described in **Table 12**:

GPT-3.5	Predicted	
ES	**1**	**0**
True **1**	**86**	28
True **0**	69	**45**

Accuracy	**57.5%**
Precision	55.5%
Recall	75.4%
Specificity	61.6%

Cohere	Predicted	
ES	**1**	**0**
True **1**	**2**	112
True **0**	1	**113**

Accuracy	**50.4%**
Precision	66.7%
Recall	01.8%
Specificity	50.2%

At first glance, the accuracy of the GPT-3.5 assessments appears close to Cohere, but a closer look shows the gulf in quality between the two. Cohere appears to be overly cautious at recommending a product for certification, predicting only 3 of the 228 products to be compliant. While not always correct, GPT-3.5 has a better mix of predictions (155 positive, 126 negative). The recall for GPT-3.5 is strong at 75.4%, meaning that for Energy Star certified products, GPT-3.5 is correct in its assessment 75.4% of the time.

5.2.2 TCO:

Using the TCO Test dataset described in **Table 12**:

GPT-3.5	Predicted	
TCO	**1**	**0**
True **1**	**86**	5
True **0**	49	**21**

Accuracy	**66.5%**
Precision	63.7%
Recall	94.5%
Specificity	80.8%

Cohere	Predicted	
TCO	**1**	**0**
True **1**	**39**	52
True **0**	7	**63**

Accuracy	**63.4%**
Precision	84.8%
Recall	42.9%
Specificity	54.8%

GPT-3.5 and Cohere both perform well, with GPT-3.5 performing slightly better. Like the Energy Star results, Cohere has a higher precision score, suggesting greater caution to recommend a certification. The recall score for GPT-3.5 is impressive at 94.5%, misclassifying only 5 TCO certified products.

6 DISCUSSION

From the results in **Section 5**, we see that traditional ML classification algorithms are better at predicting product certifications than LLMs. The worst performing ML algorithm, KNN, still scored a higher accuracy on identifying Energy Star products than GPT-3.5, the best performing LLM for TCO (85.2% vs 66.5%).

Many of the ML models performed exceptionally well when it came to identifying products that did not meet the requirements to be Energy Star or TCO certified ("0" class). For both certifications, most models produced few if any false positive outputs, meaning they would avoid improperly certifying a product that did not meet the qualifications.

That said, further analysis is needed to understand whether the observed exceptional performance of the ML models can be generalized to other products, or if it is a result of our specific methodology and dataset. Continued approaches could include alternative forms of imputation, cross-validation, variable selection, the use of more balanced data, integration of additional data resources, and other machine learning algorithms.

Furthermore, many ML algorithms lack clarity around the reason for classification and often require extensive data cleaning. The LLM approach addresses both drawbacks. Consider the following mandate and product features:

TCO Mandate 5.2.1
Display Resolution

The display panel should have a pixel density of at least 100 PPI:

$$\text{PPI} = \frac{\sqrt{horizontal\ pixels^2 + vertical\ pixels^2}}{diagonal\ of\ the\ panel\ in\ inches}$$

Product: E15

Display resolution: 1920 x 1080 pixels
Depth: 240 mm
Width: 365 mm
Height: 18.9 mm
Display brightness: 250 cd/m^2

Figure 11. Sample input to GPT for TCO 5.2.1.

This mandate is the kind of problem LLMs can solve. The requirements are clear and concise, and all terms are defined with equations. In addition, the product dataset has the values needed to assess compliance. Using, this information, GPT-3.5 was able to calculate the correct metric for assessment, come to the correct conclusion for the product, while showing work along the way:

GPT-3.5 Response:
Recommendation: TRUE
Reasoning:
To determine if the product is compliant with the TCO Certification Mandate 5.2.1, we need to calculate the pixel density (PPI) of the display panel.
First, calculate the diagonal of the panel using the width and height dimensions:
Diagonal = sqrt((240 mm)^2 + (365 mm)^2)
= sqrt(57600 + 133225)
= sqrt(190825)
≈ 436.84 mm
Next, convert the diagonal from millimeters to inches:
Diagonal ≈ 436.84 mm * 0.03937
= 17.2 inches
Finally, calculate the PPI:
PPI = sqrt((1920 pixels)^2 + (1080 pixels)^2) / 17.2 inches
= sqrt(3686400 + 1166400) / 17.2
= sqrt(4852800) / 17.2
≈ **191.74 PPI**
The calculated pixel density (PPI) of the display panel is approximately **191.74 PPI**. Since this value is greater than the required minimum of **100 PPI** specified in the mandate, the product "E15" is compliant with the TCO Certification Mandate 5.2.1 for display resolution.

Figure 12. Sample output from GPT for TCO Mandate 5.2.1.

Contrasting this with a mandate with which the LLMs performed poorly:

Energy Star Mandate 3.3.1.1
System Sleep/Alternative Low Power Mode

(1) Sleep/Alternative Low Power Mode shall be set to activate after no more than 30 minutes of user inactivity.
(2) The speed of any active 1 Gb/s or faster Ethernet network links shall be reduced when transitioning to Sleep Mode or Off Mode.

Product: E15

Password protection type: HDD
Storage media: SSD
Ethernet LAN: False
Ethernet LAN data rates: 100
Discrete graphics adapter: False

Figure 13. Sample input to GPT for Energy Star 3.3.1.1.

From this alone, we can expect some issues with the LLM assessment. These mandate requirements are much less explicit compared to TCO 5.2.1. To make matters worse, no column in the dataset addresses these requirements.

GPT-3.5 Response:
Recommendation: MORE INFO NEEDED
Reasoning:
1) The product "E15" does not provide any information regarding the sleep/alternative low power mode settings. It is not specified whether the sleep/alternative low power mode can be activated after no more than 30 minutes of user inactivity.

2) The product does not specify the speed reduction of 1 Gb/s or faster Ethernet network links when transitioning to sleep mode or off mode.

Based on the information provided, it is not possible to determine if the product meets the Energy Star Certification Mandate 3.3.1.1.

Figure 14. Sample output from GPT for Energy Star 3.3.1.1.

As expected, GPT is not able to come to any definitive assessment for this mandate. To improve the effectiveness of the LLM assessments, the user should be able to provide additional data and information about the products as necessary:

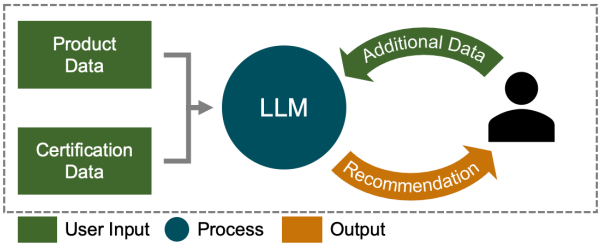

Figure 15. Updated recommendation engine architecture. This update allows for the user to interact with the LLMs, providing the additional data necessary to provide a complete assessment.

This would allow the LLMs to better handle "vague" mandates. In particular, the Energy Star certification contained numerous mandates that were not able to be answered by the data in the product dataset, which is why GPT-3.5 and Cohere performed so poorly (57.5% and 50.4% accuracy, respectively). In addition, this updated architecture would allow the tool to assess the complete certification, rather than limiting the scope to just product-related mandates.

Another improvement to the LLM process would be using "few shot learning," a method where model accuracy improves by including a small number of examples per class. In our process, we queried the LLMs using product data without providing examples of compliant products with which to compare. This is called "zero shot learning" and can cause the uncertainty observed in the LLM responses.

With these improvements implemented, we believe that LLMs can be an excellent tool for aiding in the ESG certification process. It should be noted that while the ML algorithms performed incredibly well on the product dataset, most enterprises do not have the luxury of 100k+ labeled datapoints to train. Our LLM solution would perform the same on one hundred products or one million, as there is no need for a large training set.

7 CONCLUSION

The findings presented in this paper explore the efficacy of traditional machine learning (ML) algorithms and large language models (LLMs) in assessing compliance with ESG certifications. The results indicate that overall, traditional ML algorithms outperform LLMs in predicting product certifications. Nevertheless, it is crucial to recognize the advantages offered by LLMs, particularly in addressing issues related to classification reasoning and extensive data cleaning associated with many ML algorithms.

The study emphasizes the suitability of LLMs, such as GPT-3.5, for well-defined and structured mandates. In such cases, where requirements are clear, concise, and precisely defined with equations, the LLMs demonstrate their capability to calculate metrics accurately and provide transparent conclusions. However, challenges arise when dealing with less explicit mandates, where the limitations of LLMs become evident. They struggle to derive definitive assessments due to the ambiguity of requirements and the absence of corresponding data. To enhance the effectiveness of LLM assessments, user involvement is recommended, allowing for the provision of additional data and information as necessary. Further, we recommend collecting up-to-date information on the product dataset, as gathering this data is crucial to ensure the model receives accurate input. Despite the superior performance of traditional ML algorithms in this specific context, the LLM approach remains a valuable tool, particularly in scenarios characterized by well-defined and structured mandates.

Appendix

A.1 Data

Title	Description	Link
Energy Star	Energy Star Program Requirements for Computers	link
TCO	TCO Certified, Generation 9 for notebooks	link

A.2 Web Application Demos

Title	Description	Link
Data Dictionary	Video demonstration for the data dictionary creator application	link
Product Recommendation	Walkthrough of the product recommendation application process	link

A.3 Performance Metrics

Here we will define the metrics used to assess performance.

Similarity Score:

When comparing two strings of text for semantic similarity, we used **cosine similarity** as out metric for closeness, which is the normalized dot product of two encoded string vectors:

$$K(X,Y) = \frac{X \cdot Y}{\|X\|\|Y\|}$$

https://scikit-learn.org/stable/modules/generated/sklearn.metrics.pairwise.cosine_similarity.html

Confusion Matrix:

Table that compares the predicted labels (classes) to the actual labels (classes). By default, the confusion matrix assigns label 1 to the probabilities that are greater than or equal to 0.5 and label 0 otherwise:

		Predicted	
		1	**0**
True	**1**	**True Positive (TP)**	False Negative (FN)
	0	False Positive (FP)	**True Negative (TN)**

https://scikit-learn.org/stable/modules/generated/sklearn.metrics.confusion_matrix.html

Accuracy:

Overall accuracy of the model:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

Precision:

Percentage of the predicted positive class that are actual positives:

$$Precision = \frac{TP}{TP + FP}$$

Recall:

Percentage of the actual positive class that were predicted positive by the model:

$$Recall = \frac{TP}{TP + FN}$$

Specificity:

Percentage of the actual negative class that were predicted negative by the model:

$$Specificity = \frac{TN}{TN + FP}$$

Receiver operating characteristic (ROC):

A graphical representation that demonstrates the performance of classification models across various thresholds, with false positive rate (1 – specificity) on the x axis and true positive rate (recall) on the y axis. Each point on the curve corresponds to a specific threshold's false positive and true positive rate.

https://scikit-learn.org/stable/modules/generated/sklearn.metrics.roc_curve.html

Area under curve (AUC):
A performance metric that summarizes the overall performance of an ROC curve by calculating the area under the curve, representing the integral of the curve from 0 to 1 across all thresholds.
https://scikit-learn.org/stable/modules/generated/sklearn.metrics.auc.html

A.4 Confusion Matrices

Confusion Matrices for classification of Energy Star Products:

Log. Reg.

True \ Predicted	1	0
1	**157**	24
0	50	**269**

Accuracy	**85.2%**
Precision	75.8%
Recall	86.7%
Specificity	84.3%

SVM

True \ Predicted	1	0
1	**166**	15
0	34	**285**

Accuracy	**90.2%**
Precision	83.0%
Recall	91.7%
Specificity	89.3%

KNN

True \ Predicted	1	0
1	**154**	27
0	17	**302**

Accuracy	**91.2%**
Precision	90.1%
Recall	85.1%
Specificity	94.7%

Ran. For.

True \ Predicted	1	0
1	**174**	7
0	3	**316**

Accuracy	**98.0%**
Precision	98.3%
Recall	96.1%
Specificity	99.1%

XGBoost

True \ Predicted	1	0
1	**172**	9
0	3	**316**

Accuracy	**97.6%**
Precision	98.3%
Recall	95.0%
Specificity	99.1%

AdaBoost

True \ Predicted	1	0
1	**143**	38
0	29	**290**

Accuracy	**86.6%**
Precision	83.1%
Recall	79.0%
Specificity	90.9%

Six machine learning (ML) classification algorithms to classify **Energy Star** certified products:

1. Logistic Regression (Log. Reg.)
2. Support Vector Machine (SVM)
3. K-Nearest Neighbor (KNN)
4. Random Forest (Ran. For.)
5. XGBoost
6. AdaBoost

Confusion Matrices for classification of TCO Products:

Log. Reg.

True \ Predicted	1	0
1	**98**	2
0	20	**380**

Accuracy	**95.6%**
Precision	83.1%
Recall	98.0%
Specificity	95.0%

SVM

True \ Predicted	1	0
1	**91**	9
0	2	**398**

Accuracy	**97.8%**
Precision	97.8%
Recall	91.0%
Specificity	99.5%

KNN

True \ Predicted	1	0
1	**57**	43
0	0	**400**

Accuracy	**91.4%**
Precision	100%
Recall	57.0%
Specificity	100%

Ran. For.

True \ Predicted	1	0
1	**60**	40
0	0	**400**

Accuracy	**92.0%**
Precision	100%
Recall	60.0%
Specificity	100%

XGBoost

True \ Predicted	1	0
1	**67**	33
0	0	**400**

Accuracy	**93.4%**
Precision	100%
Recall	67.0%
Specificity	100%

AdaBoost

True \ Predicted	1	0
1	**64**	36
0	0	**400**

Accuracy	**92.8%**
Precision	100%
Recall	64.0%
Specificity	100%

Six machine learning (ML) classification algorithms to classify **TCO** certified products:

1. Logistic Regression (Log. Reg.)
2. Support Vector Machine (SVM)
3. K-Nearest Neighbor (KNN)
4. Random Forest (Ran. For.)
5. XGBoost
6. AdaBoost

A.5 Workload Distribution

Task	Description	Team Member Contributions
EDA: Product Data	Analyze and understand given data to find any meaningful insights, select product category to proceed with.	Jackson, Sofia
EDA: ESG Certifications	Analyze, understand, and consolidate technical requirements across ESG certifications.	Nathan
Data Cleaning / Transformation	Preparing ESG requirements, cleaning product dataset, summarize and encode product summary and certification requirements.	Nathan – Certifications Jackson, Sofia - Products
Methodology: Data Dictionary Creator	Execute proposed methodologies for the Data Dictionary Creator.	Jackson
Methodology: Recommendation Engine	Execute proposed methodologies for the Recommendation Engine.	Jackson, Sofia, Nathan
Analysis and Results	Evaluate performance of models using labeled data.	Jackson, Sofia, Nathan
Midterm Report	PowerPoint Presentation Slides	Jackson, Sofia, Nathan
Final Report	Final report document	Jackson, Sofia, Nathan