Intel® AI for Manufacturing Certification Course

Predictive Lead Conversion Using Metadata

Project Report

Organisation: Tara Metal Industries, Ahmedabad

Group ID: G00060

Submitted By:
Memon Mohammad Ayan Anwar
Paresh Gorakh Patil

Submission Date: 09 – 07 – 2025

Acknowledgement

We sincerely thank the organizing team of the **Intel® AI for Manufacturing Certification Course**, supported by **Intel**, **Gujarat Chamber of Commerce and Industry (GCCI)**, and the **Digital Readiness Team**, for providing us with the opportunity to work on this industry-relevant AI project.

We are grateful for the structured learning resources, hands-on labs, and real-world problem statements that helped us gain practical experience in applying artificial intelligence to manufacturing and marketing domains.

We would also like to express our appreciation to the faculty coordinators and mentors for their support, guidance, and valuable feedback throughout the development of this project.

Abstract

This project presents a machine learning approach to improve lead conversion rates for businesses by analyzing user metadata. Despite high website traffic, many businesses struggle with low conversion rates, often as low as 1.5%. This project leverages historical lead data including engagement behavior, source of traffic, time spent, and device usage to predict whether a lead is likely to convert. Using logistic regression as the predictive model, the solution was developed in Python, evaluated on real data, and deployed using Streamlit for user interaction. The resulting application allows sales or marketing teams to enter new lead details and receive an instant prediction on the likelihood of conversion. The project demonstrates how AI can provide actionable insights that improve marketing efficiency and reduce manual prioritization efforts.

Table of Content

Sr. No.	Title	Page No.
*	Acknowledgement	I
*	Abstract	II
*	Table of Content	III
1	**Project Overview**	**1**
	1.1 Project Title	1
	1.2 Project Description	1
	1.3 Timeline	1
	1.4 Benefits	1
	1.5 Team Members	1
	1.6 Risks	1
2	**Objectives**	**3**
	2.1 Primary Objective	3
	2.2 Secondary Objectives	3
	2.3 Measurable Goals	3
3	**Methodology**	**4**
	3.1 Approach	4
	3.2 Phases	4
	3.3 Deliverables	4
	3.4 Testing and Quality Assurance	4
	3.5 Risk Management	4
4	**Technologies Used**	**5**
	4.1 Programming Languages	5
	4.2 Development Frameworks	5
	4.3 Database Management Systems	5
	4.4 Development Tools	5
	4.5 Testing Tools	5
	4.6 Cloud Services	5
	4.7 Security	5
	4.8 APIs and Web Services	5
5	**Results**	**7**
	5.1 Key Metrics	7
	5.2 ROI	7
6	**Project Demonstration**	**8**

7	**Conclusion**	**11**
	7.1 Recap of the Project	11
	7.2 Key Takeaways	11
	7.3 Future Plans	11
	7.4 Successes and Challenges	11
8	**Project Specifics**	**13**
	8.1 Project URL	13
	8.2 GitHub URL	13
	8.3 Notebook URL	13
	8.4 Dataset URL	13
	8.5 Presentation URL	13

1. Project Overview

1.1 Project Title

Predictive Lead Conversion Using Metadata

1.2 Project Description

In the current business environment, companies often attract a high volume of leads through various digital marketing channels such as websites, social media, and online advertisements. However, the conversion rate of these leads into actual customers remains low, often around 1.5%. Identifying which leads are likely to convert is a complex challenge that consumes time and resources.

The objective of this project is to leverage artificial intelligence and machine learning to build a predictive model that can analyze lead metadata, including user behavior, engagement patterns, and demographic information, to estimate the likelihood of conversion. By using this model, businesses can prioritize high-value leads and improve overall marketing efficiency.

This project was undertaken as part of the Intel® AI for Manufacturing Certification Course, providing practical experience in applying machine learning to solve real-world problems.

1.3 Timeline

The project was executed over a structured timeline consisting of the following key phases:

1. Dataset selection, exploration, and problem understanding.
2. Data cleaning, preprocessing, and exploratory data analysis.
3. Model development, training, and evaluation.
4. Deployment of the predictive model as a web application using Streamlit.
5. Report writing, documentation, and presentation preparation.

1.4 Benefits

- Enables data-driven decision-making for lead prioritization.
- Improves lead conversion rates, thereby increasing potential revenue.
- Reduces marketing and sales effort on leads with low conversion probability.
- Provides a quick and accessible tool for predicting lead outcomes.

1.5 Team Members

- **Memon Mohammad Ayan Anwar**
- **Paresh Gorakh Patil**

1.6 Risks

- **Data Quality Issues:** Incomplete or inaccurate data can negatively affect model accuracy.

- **Model Generalization:** The model may not perform equally well on unseen data from different industries or markets.
- **Deployment Limitations:** Free-tier deployment platforms may impose limitations on usage, scalability, and uptime.

---

2. Objectives

2.1 Primary Objective

The primary objective of this project is to develop an artificial intelligence-based predictive model that can analyze lead metadata and accurately predict the likelihood of a lead converting into a customer. The goal is to help businesses optimize their lead management process and improve conversion rates through data-driven insights.

2.2 Secondary Objectives

- To perform exploratory data analysis (EDA) to understand lead behavior patterns and key influencing factors.
- To clean and preprocess the dataset to ensure quality inputs for the model.
- To select and train a suitable machine learning model using real-world lead data.
- To deploy the predictive model as an interactive web application for non-technical users to access and use.
- To evaluate the model's performance using key classification metrics.

2.3 Measurable Goals

- Achieve an **accuracy score above 75%** for predicting lead conversion.
- Obtain **precision, recall, and F1-scores** in the acceptable range for both converting and non-converting leads.
- Successfully deploy the model on a **publicly accessible platform** for real-time predictions.
- Ensure the web application provides **instant predictions** with minimal processing time.

---

3. Methodology

3.1 Approach

The project followed a structured machine learning workflow combined with agile principles for iterative development. Each stage of the project was designed to progressively refine the dataset, the model, and the deployment to meet the objectives set.

3.2 Phases

- **Phase 1:** Data Collection and Understanding
- **Phase 2:** Data Preprocessing and Cleaning
- **Phase 3:** Feature Engineering and Model Training
- **Phase 4:** Model Evaluation and Metrics Analysis
- **Phase 5:** Web App Development and Deployment

3.3 Deliverables

- Cleaned and processed dataset ready for modeling.
- Jupyter Notebook demonstrating the complete workflow.
- Trained and saved machine learning model.
- Interactive Streamlit web application.
- Project presentation and report documentation.

3.4 Testing and Quality Assurance

The model was evaluated using standard classification metrics including **accuracy, precision, recall, and F1-score**. The confusion matrix was analyzed to assess class-wise performance. The deployed web application was tested to ensure correct predictions, error handling, and ease of use for end users.

3.5 Risk Management

- Addressed missing or inconsistent data through careful preprocessing.
- Mitigated overfitting by using simple, interpretable models and evaluating on unseen data.
- Ensured deployment compatibility through version control and thorough testing.
- Kept the application lightweight to maintain performance on free-tier cloud hosting services.

---

4. Technologies Used

4.1 Programming Languages

- **Python** was used as the primary programming language due to its extensive ecosystem of data science libraries and ease of use for machine learning development.
- **Python 3.10**

4.2 Development Frameworks

- **Scikit-learn**: Used for building and training the machine learning model (Logistic Regression).
- **Pandas**: Used for data manipulation and preprocessing.
- **NumPy**: Used for numerical operations and efficient data handling.

4.3 Database Management Systems

- The project did not require a dedicated database system as the dataset was stored and used in **CSV** format for simplicity and ease of access.

4.4 Development Tools

- **Jupyter Notebook**: Used for exploratory data analysis, model building, and result visualization.
- **Visual Studio Code (VS Code)**: Used for writing the Streamlit application script and managing the project files.
- **GitHub**: Used for version control and code hosting.

4.5 Testing Tools

- Manual testing of the Streamlit web application was performed to verify input handling and output correctness.
- Model evaluation was conducted using **Scikit-learn's built-in evaluation functions** such as confusion matrix, accuracy score, and classification report.

4.6 Cloud Services

- **Streamlit Cloud** was used to deploy the web application, allowing public access to the predictive lead conversion tool without requiring local setup.

4.7 Security

- As the project was a prototype and did not involve sensitive or personal data, basic security considerations were sufficient. Streamlit deployment provides SSL encryption by default for data transmitted between the user and the web app.

4.8 APIs and Web Services

- The project did not utilize any third-party APIs or external web services. All predictions were generated using the trained model running directly within the deployed Streamlit application.

---

5. Results

5.1 Key Metrics

The predictive model was evaluated using key classification metrics on the test dataset. The following performance results were obtained:

- **Accuracy:** 77.86%
- **Precision:** 76%
- **Recall:** 72%
- **F1-Score:** 74%

A confusion matrix was generated to visualize the model's classification performance:

	Predicted: No	**Predicted: Yes**
Actual: No (Class 0)	459	96
Actual: Yes (Class 1)	122	308

These results indicate that the model provides balanced performance in correctly identifying both converting and non-converting leads.

5.2 ROI

While this project was conducted as part of an academic and certification requirement, the potential business return on investment (ROI) includes:

- **Increased Efficiency:** By focusing marketing and sales efforts on leads with higher conversion probabilities, businesses can optimize resource allocation.
- **Cost Reduction:** Reducing time spent on low-potential leads lowers overall lead management costs.
- **Revenue Improvement:** Even modest improvements in conversion rates can significantly enhance revenue over time.

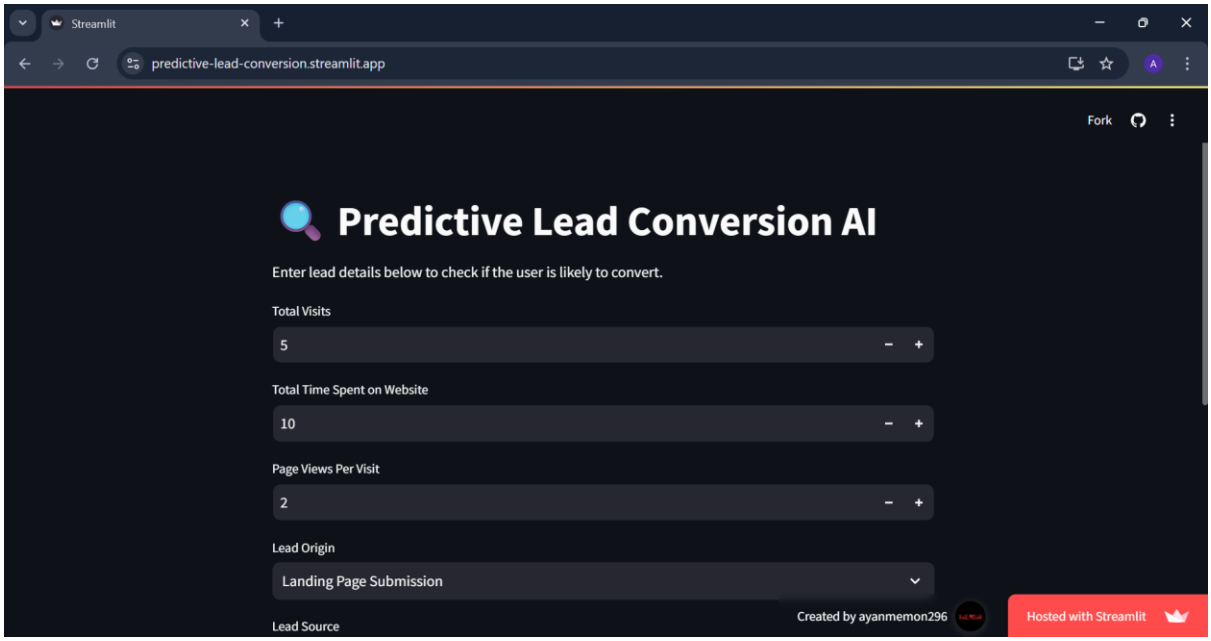
The deployment of this model as a web application allows instant predictions, enabling teams to make faster and smarter decisions without additional infrastructure investment.

---

6. Project Demonstration

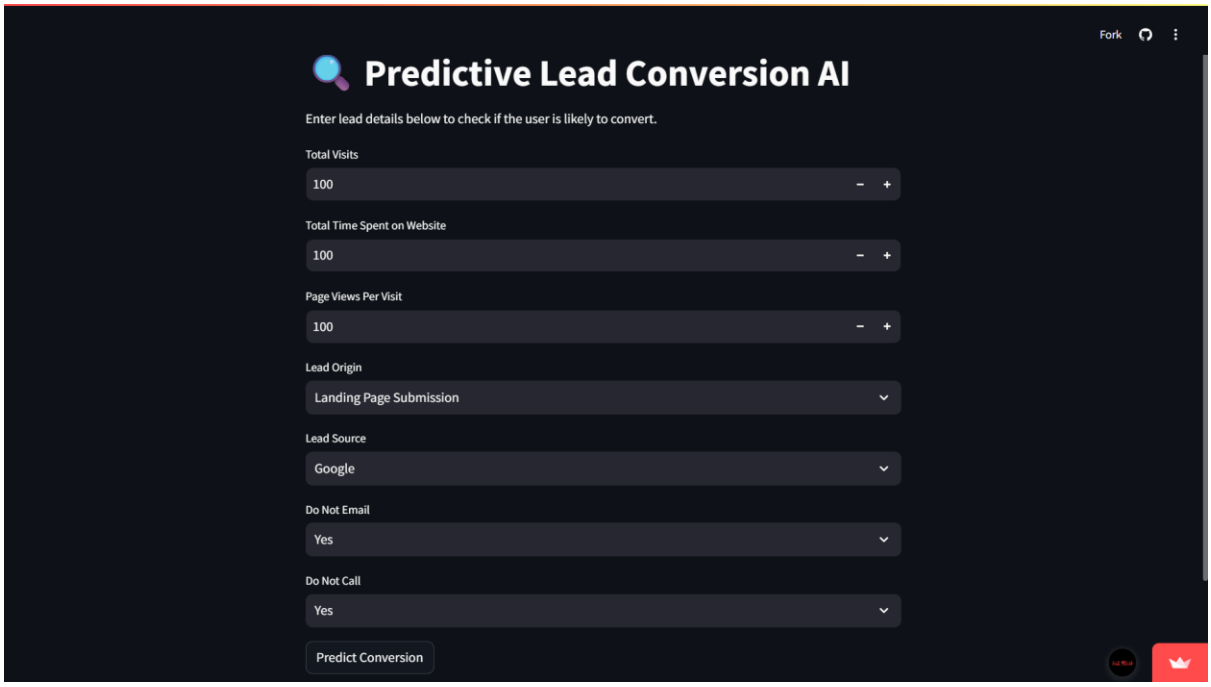
Step-by-Step Guide with Screenshots

Step 1: Accessing the Web Application



Caption:
Figure 1: Homepage of the deployed Predictive Lead Conversion web application where users can enter lead details.

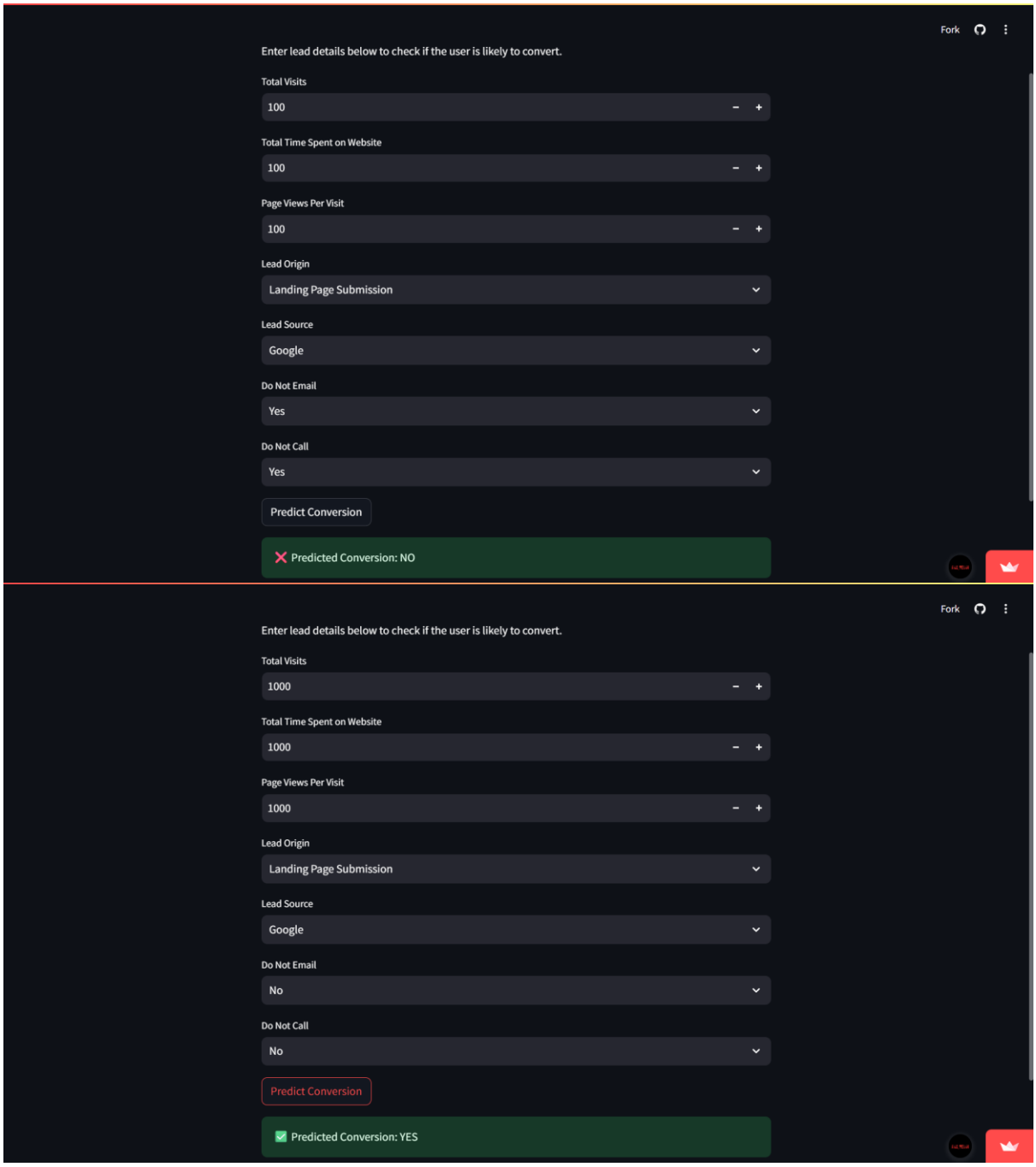
Step 2: Entering Lead Details



Caption:

Figure 2: User inputs lead information such as lead source, time spent on site, pages visited, and device type to predict conversion probability.

Step 3: Viewing Prediction Result



Caption:
Figure 3: Prediction result displayed instantly indicating whether the lead is likely to convert or not.

Step 4: Model Training and Evaluation

```
## 7. Evaluate the Model
y_pred = model.predict(X_test)
print("Accuracy Score:", accuracy_score(y_test, y_pred))
print("Confusion Matrix:\n", confusion_matrix(y_test, y_pred))
print("Classification Report:\n", classification_report(y_test, y_pred))
```

```
Accuracy Score: 0.7786802030456853
Confusion Matrix:
 [[459  96]
 [122 308]]
Classification Report:
              precision    recall  f1-score   support

           0       0.79      0.83      0.81       555
           1       0.76      0.72      0.74       430

    accuracy                           0.78       985
   macro avg       0.78      0.77      0.77       985
weighted avg       0.78      0.78      0.78       985
```

Caption:
Figure 4: Model evaluation in Jupyter Notebook showing accuracy, precision, recall, and confusion matrix.

Summary of How the Project Works

1. The user accesses the deployed Streamlit web app through a browser.
2. The app presents a simple form where the user inputs lead-related information such as:
    - Lead Source (Website, Google, Referral, etc.)
    - Time spent on the website (in minutes)
    - Number of pages visited
    - Device used (Mobile or Desktop)
3. On clicking the **Predict** button, the trained Logistic Regression model processes the inputs and instantly predicts whether the lead is **likely to convert (Yes)** or **unlikely to convert (No)**.
4. The prediction assists businesses in focusing their efforts on high-probability leads, improving marketing efficiency and conversion success.

---

7. Conclusion

7.1 Recap of the Project

The project titled **Predictive Lead Conversion Using Metadata** aimed to address the challenge of low lead conversion rates in businesses by developing a machine learning model that predicts the likelihood of a lead converting into a customer. The project successfully covered all stages including data exploration, preprocessing, model training, evaluation, and deployment as a web application accessible to users.

Through this solution, businesses can gain actionable insights into their lead pool and focus their efforts on leads most likely to convert, resulting in improved marketing efficiency and potential revenue growth.

7.2 Key Takeaways

- Applying AI to real-world business problems can significantly enhance decision-making and operational efficiency.
- Even simple and interpretable models like Logistic Regression can deliver valuable insights when applied correctly to structured data.
- Clean, high-quality data is critical for building effective predictive models.
- Deploying models as user-friendly web applications can help bridge the gap between technical development and practical business usage.

7.3 Future Plans

- The current model could be further enhanced by experimenting with advanced algorithms such as Random Forests, XGBoost, or Neural Networks.
- Additional features and external data sources (e.g., customer interaction history, CRM data) can be incorporated to improve prediction accuracy.
- The web application can be extended to include visualization dashboards, lead scoring explanations, and integration with real-time lead databases.

7.4 Successes and Challenges

Successes:

- Achieved a functional AI-based lead conversion predictor with an accuracy of approximately 78%.
- Successfully deployed the model as an interactive web application accessible to users without technical expertise.
- Learned and applied end-to-end machine learning project development from data collection to deployment.

Challenges:

- The original dataset required significant cleaning and preprocessing due to missing and inconsistent values.

- Maintaining balance between model complexity and deployment simplicity was necessary to ensure the app runs smoothly on free-tier services.
- Achieving very high accuracy was limited by the inherent quality and size of the available dataset.

---

8. Project Specifics

8.1 Project URL

The deployed web application for the **Predictive Lead Conversion Using Metadata** project is accessible at the following link:

Web App:
https://predictive-lead-conversion.streamlit.app/

This web app allows users to enter lead metadata and receive an instant prediction on whether the lead is likely to convert.

8.2 GitHub URL

The complete project files, including code, dataset, model files, presentation, and report, are hosted on GitHub:

GitHub Repository:
https://github.com/AyanMemon296/Predictive-Lead-Conversion

8.3 Notebook URL

The full Jupyter Notebook containing data exploration, preprocessing, model training, and evaluation is available within the GitHub repository at the following path:

https://github.com/AyanMemon296/Predictive-Lead-Conversion/blob/main/LeadConversion_Final.ipynb

Notebook File:
`LeadConversion_Final.ipynb`

8.4 Dataset URL

The dataset is also included directly in the GitHub repository under the file name Leads.csv.

Dataset Link:
https://github.com/AyanMemon296/Predictive-Lead-Conversion/blob/main/Leads.csv

8.5 Presentation URL

Presentation (PPTX) Link:
https://github.com/AyanMemon296/Intel-AI-Certification/tree/main/Project/Predictive-Lead-Conversion/Predictive_Lead_Conversion_Presentation.pptx

---