THIRD YEAR INDUSTRIAL TRAINING SEMINAR REPORT

INDUSTRIAL TRAINING
SEMINAR REPORT

Submitted in partial fulfilment of the Degree of Bachelor of Technology
Rajasthan Technical University

By

Naitik Pareek
(PIET21AD030)

DEPARTMENT OF ARTIFICIAL INTELLIGENCE
& DATA SCIENCE
POORNIMA INSTITUTE OF ENGINEERING & TECHNOLOGY, JAIPUR
(Academic Year 2023-24)

RAJASTHAN TECHNICAL UNIVERSITY

POORNIMA INSTITUTE OF ENGINEERING & TECHNOLOGY, JAIPUR

CERTIFICATE

This is to certify that THIRD Year Industrial Training Seminar Report entitled "**Sales Forcasting using Machine learning**" has been submitted by "Naitik Pareek (PIET21AD030)" for partial fulfilment of the Degree of Bachelor of Technology of Rajasthan Technical University. It is found satisfactory and approved for submission.

Date: 22/09/2023

Dr. Budesh Kanwar
Head,
Dept. of AI & DS
PIET, Jaipur

Dr. Dinesh Goyal
Director,
PIET, Jaipur

Company Certificate

DECLARATION

I hereby declare that the Industrial Training Seminar report entitled "**Sales Forecasting using Machine learning**" was carried out and written by me under the guidance of Ms. Shiwangi Sharma Assistant Professor, Department of Computer Science Engineering, Poornima Institute of Engineering & Technology, Jaipur. This work has not been previously formed the basis for the award of any degree or diploma or certificate nor has been submitted elsewhere for the award of any degree or diploma.

Place: Jaipur — Naitik Pareek

Date: 22/09/2023 — PIET21AD030

ACKNOWLEDGEMENT

A project of such a vast coverage cannot be realized without help from numerous sources and people in the organization. I am thankful to **Dr. Dinesh Goyal, Director, PIET** for providing me a platform to carry out such a training successfully.

I am also very grateful to **Dr. Budesh Kanwar (HOD, AI & DS)** for her kind support.

I would like to take this opportunity to show my gratitude **Mr. Kamal Saini & Ms. Shiwangi Sharma** who helped me in successful completion of my Third Year Industrial Training. They have guided, motivated & were source of inspiration for me to carry out the necessary proceedings for the training to be completed successfully.

I am also grateful to the **Learn & Build** for his/her guidance and support.

I am thankful to **Mr. Rishu Dwivedi** for his/her kind support and providing me expertise of the domain to develop the project.

I am also privileged to have **Mr. Ronak Jangir** who has/have flourished me with his/her/their valuable facilities without which this work cannot be completed.

I would also like to express my hearts felt appreciation to all of my friends whose direct or indirect suggestions help me to develop this project and to entire team members for their valuable suggestions.

Lastly, thanks to all faculty members of Computer Engineering department for their moral support and guidance.

Submitted by:

Naitik Pareek

ABSTRACT

The "**Sale forecasting**" is **the process of predicting future demand for goods and services by analyzing past and present market conditions**. It involves both informal and quantitative methods, such as the use of historical sales data or current data from test markets. Demand forecasting aims to answer questions like "when," "where," and "how much" demand by utilizing previously collected and saved data.[1] It is used to predict independent demand from sales orders and dependent demand at any decoupling point for customer orders. The enhanced demand forecast reduction rules provide an ideal solution for mass customization.

Keywords:
Preparing Your Budget, Sales Analytics, Data monitoring, Threats detection, data visualization, Cassandra database.

Implementation Software and Hardware:

- Programming Languages: Python
- Database: MySql
- Deployment: Streamlit
- Data Visualization: Matplot, Seaborn
- Network Infrastructure: High-speed internet for real-time analysis

Table of Contents

List of Figures

Chapter 1

Introduction to AI & ML

1.1 Introduction

Artificial Intelligence (AI) and Machine Learning (ML) are transformative technologies that have revolutionized various industries, including cybersecurity and network infrastructure. In this chapter, we will explore the significance of training in AI and ML and provide an introduction to Learn and Build Private Limited, a leading company specializing in network infrastructure, cybersecurity, and training services.

1.1.1 Significance of the Training

AI and ML have emerged as key drivers of innovation and efficiency in today's technology-driven world. These technologies have the potential to automate tasks, make data-driven decisions, and enhance security measures. As a result, training in AI and ML has become increasingly significant for individuals and organizations alike. The importance of AI and ML training can be summarized as follows:

Skills Development: AI and ML training programs equip individuals with the skills and knowledge required to work with cutting-edge technologies. This enables professionals to stay competitive in the job market and contribute effectively to their organizations.

Improved Decision-Making: AI and ML algorithms can analyze vast amounts of data to extract valuable insights. Training in these fields empowers professionals to make data-driven decisions that can lead to better business outcomes.

Cybersecurity: AI and ML are instrumental in cybersecurity for threat detection, anomaly detection, and network security. Training in AI and ML is crucial for cybersecurity professionals to stay ahead of evolving threats.

Innovation: AI and ML are at the forefront of technological innovation. Individuals and organizations trained in these areas can develop innovative solutions, products, and services that can disrupt industries and drive growth.

1.2 Company Profile: Learn and Build Private Limited

We are Learn and Build (LnB), a tech learning vertical by TechieNest Pvt. Ltd. Furthering our 10+ years of strong legacy and training over 200K candidates from 300+ premium institutions including IIT's & NITs, LnB aims at making quality technical education accessible to learners across the country. LnB works as the cornerstone for budding technocrats and stepping stone for working professionals, enabling India with technology creators. Here is an overview of the company's profile

Company Infrastructure: Learn and Build boosts state-of-the-art infrastructure equipped with advanced technology and resources to deliver top-notch services and training programs.

Number of Employees: The Company employs a dedicated team of experts with a deep understanding of Internetworking Technologies, Operating Systems, Relational Database Systems, Web Servers, and Security/Firewalls. The number of employees is substantial, reflecting the company's commitment to excellence.

Organizational Structure: Learn and Build follows a hierarchical organizational structure that promotes efficient communication and collaboration among its teams. This structure enables the company to streamline its operations and deliver high-quality services.

Branches: While headquartered in Rajasthan, India has established a network of branches and partner locations across the region to extend its reach and serve a wider customer base effectively.

Projects: Learn and Build has successfully executed numerous projects for enterprise customers, focusing on enhancing their network infrastructure, data security, and overall operational efficiency.

In addition to these core attributes, Learn and Build Private Limited places a strong emphasis on information security, with a team of dedicated security specialists who continuously monitor and safeguard the enterprise data of their customers.

Chapter 2

Technology Specification

In this chapter, we will discuss the key technologies involved in the AI and ML training programs offered by Learn and Build Private Limited. Additionally, we will explore the tools and technologies employed by the company in its projects and services.

2.1 Language Learned

Learn And Build AI and ML training programs cover a range of programming languages and technologies to provide participants with a comprehensive skill set. Some of the key languages learned in these training programs include:

1. **Python:** Python is a widely used language in AI and ML due to its simplicity, extensive libraries, and strong support for data science and machine learning frameworks like TensorFlow, PyTorch, and scikit-learn.

2. **SQL:** SQL (Structured Query Language) is crucial for data manipulation and management in AI and ML projects that involve databases.

3. **Jupyter Notebooks**: Jupyter Notebooks provide an interactive environment for writing and executing code, making them valuable for data exploration and sharing insights.

2.2 Tools and Company Technology

Learn And Build Private Limited leverages a range of tools and technologies in its projects and services, particularly in the domains of network infrastructure, cybersecurity, and IT consulting. Some of the key tools and technologies used by the company include:

1. **Virtualization Technologies:** Virtualization is fundamental to modern IT environments. Learn and Build utilizes virtualization technologies such as VMware vSphere and Microsoft Hyper-V for server consolidation and efficient resource allocation.

2. **Cloud Computing Platforms:** The Company may leverage cloud platforms like Amazon Web Services (AWS) and Microsoft Azure for scalable and cost-effective solutions.

3. **Data Analytics and Visualization:** In projects involving data analysis and visualization, Learn And Build uses tools like Python libraries such as matplotlib and Plotly, Seaborn.

4. **Project Management Tools:** Learn And Build follows industry-standard project management methodologies and may use tools like Jira, Trello, or Microsoft Project for project planning and execution.

5. **Training and Learning Management Systems:** For its training programs, the company may employ Learning Management Systems (LMS) and e-learning platforms to deliver course content and track participant progress.

These technologies and tools empower Learn and Build to deliver high-quality services and training programs, ensuring that its clients and participants have access to the latest advancements in AI, ML, network infrastructure, and cybersecurity. In the following chapters, we will explore the specific training methodologies and services provided by the company in greater detail.

Chapter 3

Project Description/Technology Learned Description

In this chapter, we will delve into the detailed description of the AI and ML training programs offered by Learn and Build Private Limited. These programs cover a wide range of topics, from Python basics to advanced machine learning techniques. Let's explore each component of the training:

3.1 Python Basics

The Python Basics module serves as the foundation of the AI and ML training programs. Participants learn the fundamental concepts of the Python programming language, which is the cornerstone of AI and ML development. This module typically includes the following topics:

Introduction to Python: An overview of Python, its history, and its relevance in AI and ML.

1. Python Syntax: Learning the Syntax, Data Types, and Variables in Python

- **Syntax:** Python has a straightforward and readable syntax that uses indentation (whitespace) to define code blocks. It's known for its readability and clean structure, making it easy for beginners to grasp.
- **Data Types:** Python supports various data types, including integers (int), floating-point numbers (float), strings (str), lists, tuples, dictionaries, and more. Understanding data types is crucial for working with different kinds of data.
- **Variables:** Variables in Python are used to store data. They don't need to be explicitly declared; you can simply assign a value to a variable. For example, **x = 10** assigns the integer value 10 to the variable **x**.

2. Control Flow: Understanding Conditional Statements (if, else) and Loops (for, while) for Program Control

- **Conditional Statements (if, else):** Conditional statements allow you to make decisions in your code. "if" statements check if a condition is true and execute a block of code accordingly. "else" statements provide an alternative code block to execute if the condition is false.
- **Loops (for, while):** Loops allow you to execute a block of code repeatedly. "for" loops are used when you know how many times you want to iterate, while "while" loops continue until a specified condition is no longer true.

3. Functions: Writing and Using Functions to Organize Code

- Functions are blocks of reusable code that perform specific tasks. They allow you to break down your code into smaller, manageable parts. Functions take input parameters (if needed) and return results.

4. Data Structures: Exploring Data Structures Such as Lists, Tuples, Dictionaries, and Sets

- Python provides various data structures to store and manipulate data efficiently. Common data structures include:
    - **Lists:** Ordered collections of items, allowing for easy access and modification.
    - **Tuples:** Similar to lists but immutable (cannot be changed after creation).
    - **Dictionaries:** Key-value pairs that enable quick retrieval of values based on unique keys.
    - **Sets:** Unordered collections of unique elements.

5. File Handling: Reading from and Writing to Files in Python

- File handling allows you to work with files on your computer. You can open, read, write, and close files. Python provides built-in functions for these operations.

6. Error Handling: Dealing with Exceptions and Errors Gracefully

- Error handling is essential for managing unexpected issues in your code. Python uses try-except blocks to catch and handle exceptions.

Understanding these foundational concepts is essential for anyone learning Python, as they form the building blocks for more complex programming tasks and applications.

3.2 Data Insights: Statistical Foundations for Data Science

This module focuses on the statistical foundations required for data analysis and data science. Participants gain a solid understanding of statistical concepts and their applications in data-driven decision-making. Key topics covered in this module include:

1. Descriptive Statistics: Analysing Data through Measures

Descriptive statistics involve using various summary measures to understand and describe a dataset. Some common measures include:

- **Mean:** The average of a set of values. It provides a central measure of the dataset.
- **Median:** The middle value when data is sorted. It is less affected by outliers compared to the mean.
- **Mode:** The most frequently occurring value in a dataset.
- **Variance:** A measure of how much data points vary from the mean. It indicates the dataset's spread or dispersion.

Descriptive statistics help analysts get an initial sense of the data's central tendency, variability, and distribution.

2. Probability Distributions: Understanding Common Distributions

Probability distributions describe the likelihood of different outcomes in a random process. Two common distributions are:

- **Normal Distribution:** Also known as the Gaussian distribution or bell curve, it's characterized by a symmetrical, bell-shaped curve. Many natural phenomena, like heights and IQ scores, follow a normal distribution.
- **Binomial Distribution:** Used for modelling binary outcomes (success or failure) in a fixed number of trials. It's relevant in scenarios like coin tosses or pass/fail experiments.

Understanding these distributions helps in making probabilistic predictions and modelling real-world phenomena.

3. Statistical Inference: Making Predictions and Drawing Conclusions

Statistical inference involves making predictions and drawing conclusions about a population based on a sample of data. It includes:

- **Point Estimation:** Estimating population parameters (e.g., mean, proportion) based on sample data.
- **Interval Estimation:** Creating confidence intervals to estimate the range in which a population parameter is likely to fall.
- **Hypothesis testing (as mentioned above):** Using hypothesis tests to draw conclusions about population parameters.

Statistical inference is essential for generalizing findings from a sample to the broader population, making predictions, and supporting decision-making.

4. Correlation and Regression: Analyzing Relationships between Variables

Correlation and regression are techniques used to study relationships between variables:

- **Correlation:** Measures the strength and direction of a linear relationship between two continuous variables. The correlation coefficient ranges from -1 (perfect negative correlation) to 1 (perfect positive correlation).
- **Regression Analysis:** Examines how one or more independent variables predict or explain changes in a dependent variable. Linear regression models the relationship with a linear equation.

These techniques are crucial for understanding dependencies between variables and predicting outcomes based on them.

3.3 Visual Pandas

Visual Pandas, or Visualization with Pandas, is an essential component of data analysis and presentation. In this module, participants learn how to use Pandas, a popular Python library for data manipulation and analysis, in combination with data visualization tools to create informative and compelling visuals. Key topics include:

1. Data Cleaning: Cleaning and Pre-processing Data Using Pandas

Data cleaning is the process of identifying and correcting errors, inconsistencies, and missing values in datasets to ensure they are accurate and ready for analysis. The Python library Pandas is a powerful tool for data cleaning and pre-processing. Key tasks in data cleaning include:

- **Handling Missing Data:** Identifying and filling in missing values or removing rows/columns with missing data.
- **Removing Duplicates:** Identifying and removing duplicate records from the dataset.
- **Data Transformation:** Converting data types, scaling features, and encoding categorical variables.
- **Outlier Detection:** Identifying and handling outliers that may distort analysis results.

Data cleaning ensures that the data is of high quality and can be used effectively for analysis and modelling.

2. Data Visualization: Using Libraries like Matplotlib and Seaborn

Data visualization involves representing data graphically to reveal insights and patterns that may not be apparent in raw data. Libraries like Matplotlib and Seaborn in Python provide a wide range of tools for creating various types of charts, plots, and graphs. Common visualizations include:

- **Bar Charts:** Used for comparing categorical data.
- **Line Charts:** Used to show trends and changes over time.
- **Scatter Plots:** Used for visualizing the relationship between two variables.
- **Histograms:** Used to display data distribution.
- **Heat maps:** Used to show correlations between variables.

Effective data visualization enhances data understanding and communication, making it a crucial step in data analysis.

3. Exploratory Data Analysis (EDA): Conducting EDA to Gain Insights

Exploratory Data Analysis (EDA) is a critical phase in data analysis that involves exploring and summarizing data to gain a deeper understanding of its characteristics. EDA typically includes the following tasks:

- **Descriptive Statistics:** Calculating summary statistics such as mean, median, and variance.

- **Data Distribution:** Visualizing data distributions through histograms, box plots, and density plots.
- **Correlation Analysis:** Exploring relationships between variables using correlation matrices and scatter plots.
- **Identifying Patterns:** Detecting patterns and anomalies in the data.

EDA helps data analysts and scientists uncover initial insights, identify potential issues, and inform the next steps in the analysis process.

4. Interactive Dashboards: Creating Interactive Data Dashboards

Interactive data dashboards provide a dynamic and user-friendly way to explore and interact with data. These dashboards often use tools and frameworks like Tableau, Power BI, or custom-built web applications. Key features of interactive dashboards include:

- **Filtering and Selection:** Users can filter and select data points to view specific subsets of the data.
- **Real-time Updates:** Dashboards can update in real-time as new data becomes available.
- **Visualization Interactivity:** Users can interact with charts and graphs, zoom in, or click for additional details.
- **Data Exploration:** Dashboards allow users to explore data from various angles and perspectives.

Interactive dashboards are valuable for decision-makers and analysts as they provide a dynamic and intuitive way to explore and understand complex datasets.

3.4 Intelligent Mastery: Learning to Build ML Models and Deployments

This module marks the transition into the realm of machine learning. Participants gain hands-on experience in building machine learning models and deploying them. Topics covered in this module include:

1. Supervised Learning: Understanding Supervised Learning Algorithms for Regression and Classification Tasks

Supervised learning is a type of machine learning where the algorithm learns from labeled training data to make predictions or decisions without human intervention. It is called "supervised" because

the algorithm learns from a supervisor who provides it with the correct answers during training. There are two primary tasks in supervised learning:

- **Regression:** In regression tasks, the algorithm predicts a continuous numerical value. For example, predicting house prices based on features like square footage, number of bedrooms, and location.

- **Classification:** In classification tasks, the algorithm assigns data points to predefined categories or classes. Common examples include spam email detection (classifying emails as spam or not) and image recognition (classifying images of animals into specific species).

2. Model Training: Training Machine Learning Models Using Real Datasets

Model training is the process of teaching a machine learning algorithm to make predictions or classifications based on input data. This involves:

- **Dataset Preparation:** Gathering and pre-processing a dataset that contains examples with known inputs and corresponding correct outputs (labels).
- **Algorithm Selection:** Choosing an appropriate machine learning algorithm for the task at hand. For example, linear regression for regression tasks or decision trees for classification tasks.
- **Feature Engineering:** Selecting relevant features (input variables) and transforming data if necessary to improve model performance.
- **Training:** Using the prepared dataset to train the chosen algorithm, allowing it to learn patterns and relationships in the data.

3. Model Evaluation: Assessing Model Performance and Selecting the Appropriate Evaluation Metrics

After training a machine learning model, it is essential to assess its performance to ensure it is making accurate predictions or classifications. Model evaluation involves:

- **Choosing Evaluation Metrics:** Selecting appropriate metrics for measuring the model's performance. For example, Mean Absolute Error (MAE) for regression tasks or accuracy, precision, recall, and F1-score for classification tasks.
- **Testing the Model:** Using a separate dataset (testing or validation set) that the model has

never seen before to assess its performance. This helps estimate how well the model will perform on unseen data.

- **Fine-Tuning:** Adjusting the model's parameters or making changes to improve its performance based on evaluation results.

4. Deployment Strategies: Learning How to Deploy Machine Learning Models in Real-World Applications

Once a machine learning model has been trained and evaluated, the next step is deploying it in real-world applications. Deployment strategies involve:

- **Integration with Systems:** Integrating the model into existing software systems or applications where it can make predictions or classifications based on new input data.
- **Scalability:** Ensuring that the deployed model can handle varying levels of workload and data input.
- **Monitoring:** Implementing mechanisms to continuously monitor the model's performance in production and to retrain it periodically with new data to maintain its accuracy.
- **Ethical Considerations:** Addressing ethical and legal concerns related to the model's deployment, especially if it deals with sensitive or personal data.

3.5 Advanced Machine Learning Techniques

In this advanced module, participants delve deeper into machine learning techniques. They explore complex algorithms and strategies for solving more challenging problems. Topics may include:

1. Unsupervised Learning: Exploring Clustering and Dimensionality Reduction

Unsupervised learning is a category of machine learning where the model learns from unlabelled data, attempting to find patterns or structures within the data. Two common techniques in unsupervised learning are clustering and dimensionality reduction:

- **Clustering:** Clustering algorithms aim to group similar data points together into clusters. The goal is to identify inherent patterns or similarities in the data without prior knowledge of the categories. Common clustering algorithms include K-Means, Hierarchical Clustering, and DBSCAN. For example, in customer segmentation, clustering can group

customers with similar purchasing behaviours.

- **Dimensionality Reduction:** Dimensionality reduction techniques aim to reduce the number of features (dimensions) in a dataset while preserving as much relevant information as possible. This is especially useful when dealing with high-dimensional data, as it can improve model efficiency and reduce noise. Principal Component Analysis (PCA) and t-SNE (t-distributed Stochastic Neighbour Embedding) are popular dimensionality reduction methods.

2. Ensemble Methods: Understanding Random Forests and Gradient Boosting

Ensemble methods are machine learning techniques that combine the predictions of multiple base models to improve overall prediction accuracy and robustness. Two widely used ensemble methods are:

- **Random Forests:** Random Forest is an ensemble learning method based on decision trees. It constructs multiple decision trees during training and combines their predictions through a voting mechanism (classification) or averaging (regression). Random Forests are known for their high accuracy, resistance to overfitting, and ability to handle both categorical and numerical data.
- **Gradient Boosting:** Gradient Boosting is another ensemble technique that builds an ensemble of weak learners (typically decision trees) sequentially. It corrects the errors of the previous model by focusing on the data points that were misclassified. This iterative process results in a strong predictive model. Popular variations include AdaBoost, Gradient Boosting Machines (GBM), and XGBoost.

Ensemble methods are favoured for their ability to handle complex relationships in data, reduce overfitting, and deliver accurate predictions.

3. Neural Networks: Introduction to Deep Learning and Neural Network Architectures

Neural networks are a class of machine learning models inspired by the structure and function of the human brain. Deep learning, a subset of machine learning, focuses on neural networks with multiple layers (deep neural networks). Key points about neural networks include:

- **Deep Learning:** Deep learning models, with their deep architectures, have demonstrated remarkable success in various tasks, including image recognition, natural language

processing, and speech recognition. They consist of an input layer, multiple hidden layers, and an output layer.

- **Neural Network Architectures:** There are various neural network architectures, each designed for specific tasks. For instance, Convolutional Neural Networks (CNNs) excel in image analysis, while Recurrent Neural Networks (RNNs) are suitable for sequential data like text and time series. Transformers have become the go-to architecture for natural language processing tasks.
- **Training:** Neural networks learn by adjusting their parameters (weights and biases) during a training process. They use optimization techniques like gradient descent to minimize prediction errors.

Deep learning has had a profound impact on various industries, driving advancements in computer vision, speech recognition, and natural language understanding, among others.

3.6 Text Processing and Natural Language Processing

Natural Language Processing (NLP) is a critical field in AI and ML, focused on understanding and processing human language. In this module, participants learn NLP techniques, including:

1. Text Pre-processing: Cleaning and Tokenizing Text Data

Text data often comes with noise, inconsistencies, and unnecessary information. Text pre-processing is the initial step in preparing textual data for analysis. It involves:

- **Cleaning:** Removing any irrelevant characters, symbols, or formatting from the text. This can include removing special characters, punctuation, HTML tags, and other non-essential elements.
- **Tokenizing:** Breaking down the text into smaller units called "tokens." Tokens can be words, phrases, or sentences. Tokenization is crucial for further analysis as it helps to organize and understand the text's structure.

For example, consider the sentence: "Text pre-processing is essential for NLP analysis." After pre-processing, it might be tokenized into: ["Text", "pre-processing", "is", "essential", "for", "NLP", "analysis"].

2. Sentiment Analysis: Analysing Text Sentiment Using Machine Learning

Sentiment analysis, also known as opinion mining, is the process of determining the sentiment or emotional tone expressed in a piece of text. It is commonly used to understand whether the sentiment in the text is positive, negative, or neutral. Sentiment analysis typically involves:

- **Text Classification:** Using machine learning techniques to classify text as positive, negative, or neutral based on the sentiment expressed. This can be valuable for understanding customer feedback, social media sentiment, and product reviews.

For instance, in a customer review, sentiment analysis might classify "I love this product" as positive sentiment and "This product is terrible" as negative sentiment.

3. Named Entity Recognition (NER): Identifying Entities in Text

Named Entity Recognition (NER) is a subtask of information extraction that focuses on identifying and classifying entities in text, such as names of people, places, organizations, dates, and more. NER involves:

- **Entity Identification:** Detecting and classifying specific entities within the text. For example, identifying "Apple" as a company name, "John Smith" as a person's name, or "New York" as a location.

NER is useful in various applications, including information retrieval, language translation, and extracting structured data from unstructured text.

4. Text Classification: Classifying Text Documents into Predefined Categories

Text classification, also known as text categorization, involves categorizing text documents into predefined categories or classes based on their content. It is commonly used in various applications, including:

- **Document Classification:** Sorting news articles into categories like sports, politics, or entertainment.
- **Spam Detection:** Identifying and filtering out spam emails from legitimate ones.
- **Sentiment Analysis:** Categorizing text as positive, negative, or neutral.

Text classification relies on machine learning algorithms that learn from labelled training data, allowing them to make predictions about the category of unseen text documents.

These training modules equip participants with a strong foundation in Python, statistics, data analysis, and machine learning, empowering them to embark on AI and ML projects, make data-driven decisions, and contribute to the rapidly evolving field of artificial intelligence. The knowledge gained from these modules forms the basis for advanced AI and ML .

Chapter 4

Snapshots of the Project or Screen Layout

In this chapter, we will provide snapshots and screen layouts of key components of the **Sales Forcasting for E-Commerce** project. These images offer a visual representation of the project's user interface and features. Please note that these snapshots are illustrative and may not cover all aspects of the project.

4.1 Dashboard

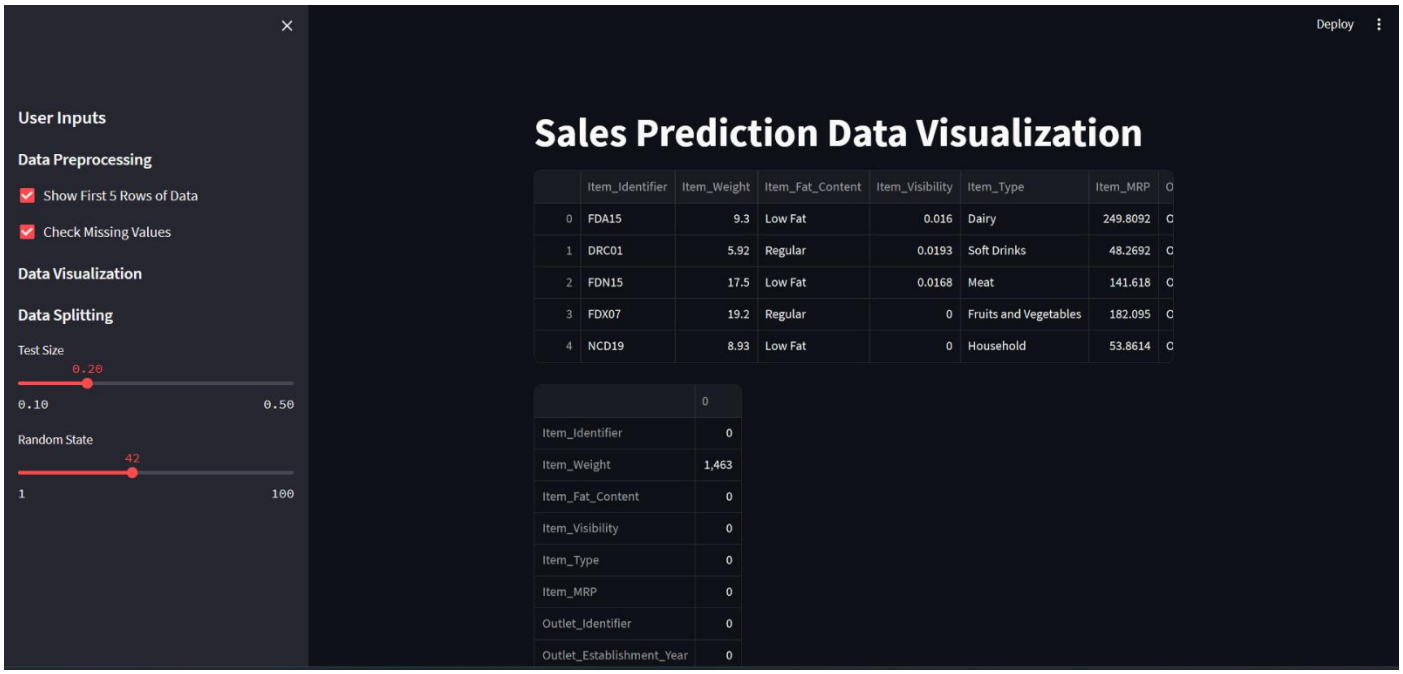

Figure 4.1 Dashboard

The dashboard in the Real-time Sales Forecasting project plays a pivotal role in providing insights of the datasets and display required predictions. It serves as the central hub for monitoring and managing sales information, offering several key functionalities:

1. **Real-time Monitoring:** The dashboard continuously streams and updates sale data,

ensuring that sales professionals have access to the most current information. This real-time monitoring is essential for tracking changes in sale's vital information and conditions as they occur.

2. **Intuitive Interface:** The dashboard is designed with user-friendliness in mind. It offers an intuitive and easy-to-navigate interface, making it accessible to sales professionals with varying levels of technical expertise. This ensures that users can quickly and efficiently access the data they need.
3. **Access to Vital Information:** Within the dashboard, sales professionals can access vital sale information such as est. year, MRP, price and outlet sales. This information is presented in a clear and organized manner, allowing for quick assessment and analysis.
4. **Analytics and Insights:** In addition to displaying raw data, the dashboard provides advanced analytics and visualization tools. Sales professionals can visualize trends, patterns, and anomalies in sale data, enabling them to make informed decisions and interventions.
5. **Alert System:** The dashboard is equipped with an alert system that immediately notifies sales professionals of critical sale conditions or abnormalities. This rapid alerting mechanism ensures that sales providers can respond promptly to emergencies.
6. **User Interaction:** The interactive features of the dashboard enable sales professionals to interact with the data. They can zoom in on specific time frames, select individual sales, and customize their view to focus on specific vital signs or sale groups.
7. **Data Integration:** The dashboard seamlessly integrates with the underlying sales database, allowing for efficient data storage and retrieval. This integration ensures that historical sale data is readily available for reference and analysis.

4.2 Main Code

In [20]:
```
Item_type_Weight=X_train_N.pivot_table(values='Item_Weight',index='Item_Type',aggfunc='median').reset_index()

Item_type_Weight_map=dict(zip(Item_type_Weight['Item_Type'],Item_type_Weight['Item_Weight']))
Item_type_Weight_map.items()
```

Out[20]:
```
dict_items([('Baking Goods', 11.925), ('Breads', 10.6), ('Breakfast', 10.695), ('Canned', 12.35), ('Dairy', 13.35), ('Frozen Foods', 12.85), ('Fruits and Vegetables', 12.85), ('Hard Drinks', 9.695), ('Health and Hygiene', 12.35), ('Household', 13.15), ('Meat', 12.65), ('Others', 14.5), ('Seafood', 11.65), ('Snack Foods', 12.925), ('Soft Drinks', 11.6), ('Starchy Foods', 13.5)])
```

In [21]:
```
def impute_item_weight(data_frame):
    data_frame.loc[:,'Item_Weight']= data_frame.loc[:,'Item_Weight'].fillna(data_frame.loc[:,'Item_Identifier'].map(Item_ID_
    data_frame.loc[:,'Item_Weight']= data_frame.loc[:,'Item_Weight'].fillna(data_frame.loc[:,'Item_Type'].map(Item_ID_Weight
    return data_frame
```

In [22]:
```
X_train_N=impute_item_weight(X_train_N)
```

In [23]:
```
X_train_N.isnull().sum()
```

Out[23]:
```
Item_Identifier               0
Item_Weight                  16
Item_Fat_Content              0
Item_Visibility               0
Item_Type                     0
Item_MRP                      0
Outlet_Identifier             0
Outlet_Establishment_Year     0
Outlet_Size                1935
Outlet_Location_Type          0
Outlet_Type                   0
dtype: int64
```

In [24]:
```
X_train_N.groupby(by=['Outlet_Type','Outlet_Size']).size()
```

Out[24]:
```
Outlet_Type        Outlet_Size
Grocery Store      Small           413
Supermarket Type1  High            744
                   Medium          720
                   Small          1493
Supermarket Type2  Medium          752
Supermarket Type3  Medium          761
```

Figure 4.2 Main Code

The sensor reading map is a critical component of the Real-time Sales Analytics project that provides sales professionals with a visual representation of sale locations and real- time sensor data. Here's a brief explanation of its functionality:

1. **Sale Locations:** The sensor reading map displays the locations of sales within a sales facility or any monitored environment. Each sale is typically represented by a marker or icon on the map, allowing sales professionals to identify where each sale is situated.
2. **Tracking Sale Movements:** Sales professionals can track sale movements in real-time as sales move within the facility. This is particularly valuable in situations where sale mobility needs to be monitored closely, such as in a hospital or a senior care facility.
3. **Sensor Data Visualization:** The map provides a graphical representation of sensor readings associated with each sale's location. For example, it might display color-coded data points or graphs that show variations in vital signs. This allows sales professionals to quickly assess the health status of multiple sales at a glance.

4.3 EDA

```
Outlet_Type                0
dtype: int64
```

In [18]:
```
X_train_N[['Item_Identifier','Item_Weight']].drop_duplicates().sort_values(by=['Item_Identifier'])
```

Out[18]:

	Item_Identifier	Item_Weight
1245	DRA12	11.60
4416	DRA24	NaN
4900	DRA24	19.35
8371	DRA59	8.27
6057	DRA59	NaN
...	...	...
6056	NCZ42	10.50
6879	NCZ53	9.60
2084	NCZ53	NaN
1896	NCZ54	NaN
214	NCZ54	14.65

2515 rows × 2 columns

In [19]:
```
Item_ID_Weight=X_train.pivot_table(values='Item_Weight',index='Item_Identifier',aggfunc='median').reset_index()
Item_ID_Weight_map=dict(zip(Item_ID_Weight['Item_Identifier'],Item_ID_Weight['Item_Weight']))
Item_ID_Weight_map.items()
```

Out[19]:
```
dict_items([('DRA12', 11.6), ('DRA24', 19.35), ('DRA59', 8.27), ('DRB01', 7.39), ('DRB13', 6.115), ('DRB24', 8.785), ('DRB2
5', 12.3), ('DRB48', 16.75), ('DRC01', 5.92), ('DRC12', 17.85), ('DRC13', 8.26), ('DRC24', 17.85), ('DRC25', 5.73), ('DRC2
7', 13.8), ('DRC36', 13.0), ('DRC49', 8.67), ('DRD01', 12.1), ('DRD12', 6.96), ('DRD13', 15.0), ('DRD15', 10.6), ('DRD24', 1
3.85), ('DRD25', 6.135), ('DRD27', 18.75), ('DRD37', 9.8), ('DRD49', 9.895), ('DRD60', 15.7), ('DRE01', 10.1), ('DRE03', 19.
6), ('DRE12', 4.59), ('DRE13', 6.28), ('DRE15', 13.35), ('DRE25', 15.35), ('DRE27', 11.85), ('DRE37', 13.5), ('DRE48', 8.4
3), ('DRE49', 20.75), ('DRE60', 9.395), ('DRF01', 5.655), ('DRF03', 19.1), ('DRF13', 12.1), ('DRF15', 18.35), ('DRF23', 4.6
1), ('DRF25', 9.0), ('DRF27', 8.93), ('DRF36', 16.1), ('DRF37', 17.25), ('DRF48', 5.73), ('DRF49', 7.27), ('DRF51', 15.75),
('DRF60', 10.8), ('DRG01', 14.8), ('DRG03', 14.5), ('DRG11', 6.385), ('DRG13', 17.25), ('DRG15', 6.13), ('DRG23', 8.88), ('D
RG25', 10.5), ('DRG27', 8.895), ('DRG36', 14.15), ('DRG37', 16.2), ('DRG39', 14.15), ('DRG48', 5.78), ('DRG49', 7.81), ('DRG
```

Figure 4.3 EDA

The weighted item graph within the Real-time Sales Analytics project is a crucial feature that offers a real-time display of weighted item readings for sales. This graph serves as a graphical representation of vital sign data, specifically focusing on weighted item measurements. Its primary purpose is to aid sales professionals in the early detection of anomalies and irregularities in sale's weighted item.

1. **Real-Time Monitoring:** The weighted item graph provides sales professionals with continuously updated weighted item readings in real-time. This means that as soon as a new measurement is recorded, it is reflected on the graph immediately.
2. **Graphical Representation:** Visualizing data through graphs and charts is often more informative than raw numbers. The graphical representation of weighted item readings

allows sales providers to quickly assess trends, spikes, or dips in a sale's weighted item.

3. **Anomaly Detection:** By closely monitoring the weighted item graph, sales professionals can identify any unusual patterns or sudden changes in a sale's weighted item.
4. **Informed Decision-Making:** Armed with real-time data and visual insights from the weighted item graph, sales professionals can make informed decisions regarding sale care.
5. **Enhanced Sale Care:** Ultimately, the weighted item graph contributes to enhanced sale care by facilitating early intervention and ensuring that sales providers have access to the most up-to-date information about a sale's vital signs.

4.4 Visualization Graphs

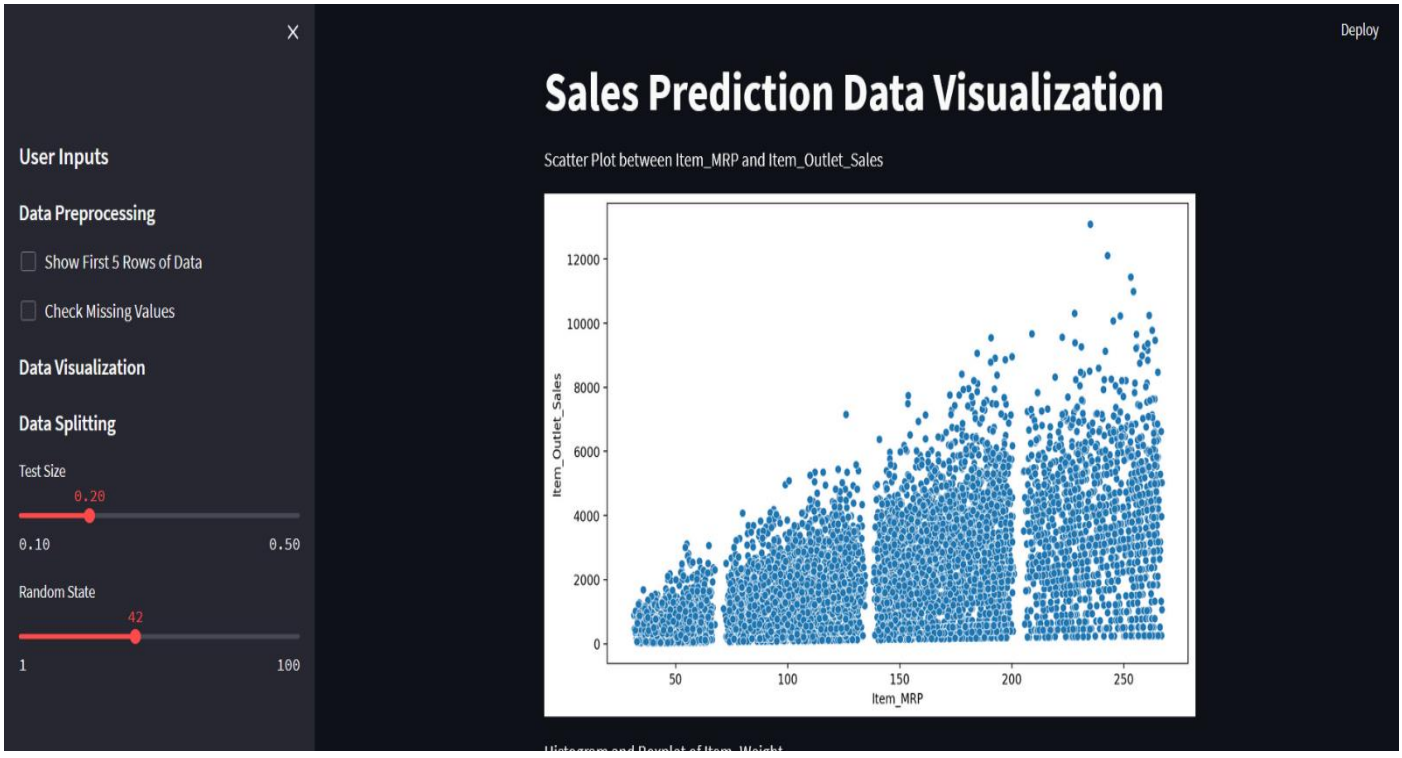

Figure 4.4 Visualization Graphs(a)

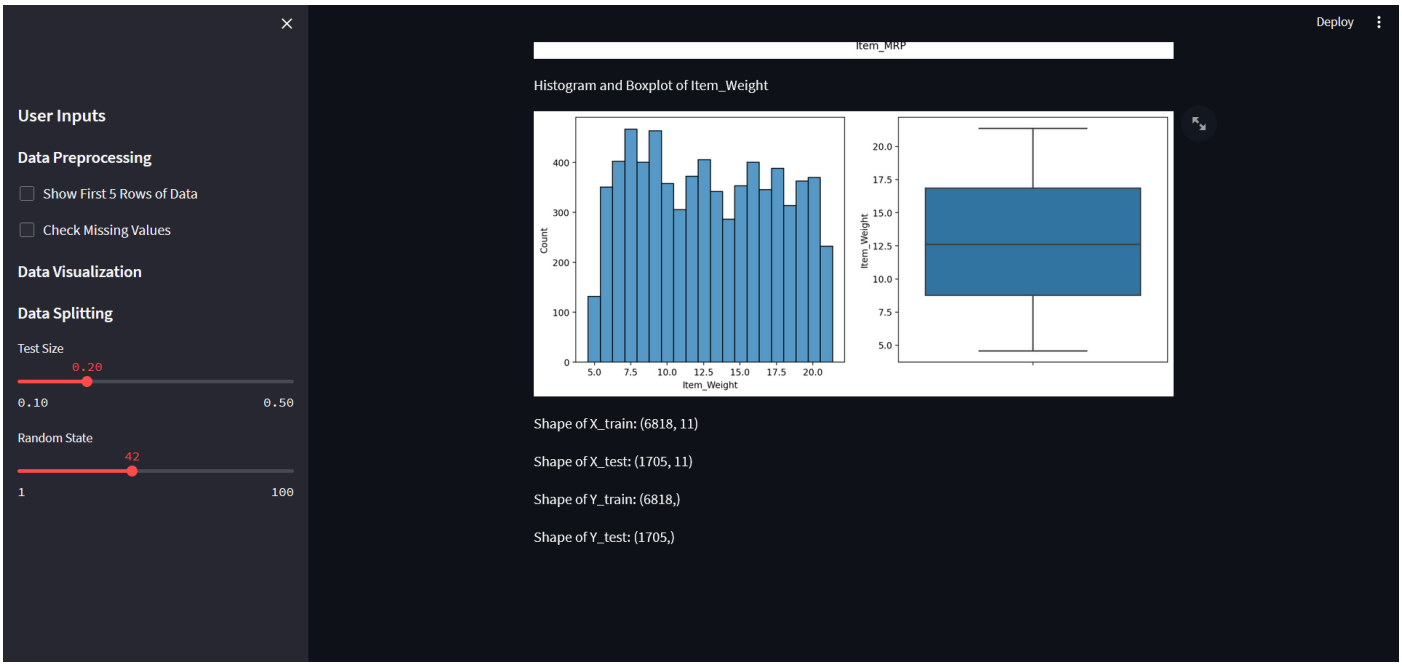

Figure 4.4 Visualization Graphs(b)

Item_Iden	Item_Weig	Item_Fat_	Item_Visib	Item_Type	Item_MRP	Outlet_Ide	Outlet_Est	Outlet_Siz	Outlet_Lo	Outlet_Typ	Item_Outlet_Sales
FDA15	9.3	Low Fat	0.016047	Dairy	249.8092	OUT049	1999	Medium	Tier 1	Supermark	3735.138
DRC01	5.92	Regular	0.019278	Soft Drinks	48.2692	OUT018	2009	Medium	Tier 3	Supermark	443.4228
FDN15	17.5	Low Fat	0.01676	Meat	141.618	OUT049	1999	Medium	Tier 1	Supermark	2097.27
FDX07	19.2	Regular	0	Fruits and	182.095	OUT010	1998		Tier 3	Grocery St	732.38
NCD19	8.93	Low Fat	0	Household	53.8614	OUT013	1987	High	Tier 3	Supermark	994.7052
FDP36	10.395	Regular	0	Baking Goo	51.4008	OUT018	2009	Medium	Tier 3	Supermark	556.6088
FDO10	13.65	Regular	0.012741	Snack Foo	57.6588	OUT013	1987	High	Tier 3	Supermark	343.5528
FDP10		Low Fat	0.12747	Snack Foo	107.7622	OUT027	1985	Medium	Tier 3	Supermark	4022.764
FDH17	16.2	Regular	0.016687	Frozen Foo	96.9726	OUT045	2002		Tier 2	Supermark	1076.599
FDU28	19.2	Regular	0.09445	Frozen Foo	187.8214	OUT017	2007		Tier 2	Supermark	4710.535
FDY07	11.8	Low Fat	0	Fruits and	45.5402	OUT049	1999	Medium	Tier 1	Supermark	1516.027
FDA03	18.5	Regular	0.045464	Dairy	144.1102	OUT046	1997	Small	Tier 1	Supermark	2187.153
FDX32	15.1	Regular	0.100014	Fruits and	145.4786	OUT049	1999	Medium	Tier 1	Supermark	1589.265
FDS46	17.6	Regular	0.047257	Snack Foo	119.6782	OUT046	1997	Small	Tier 1	Supermark	2145.208
FDF32	16.35	Low Fat	0.068024	Fruits and	196.4426	OUT013	1987	High	Tier 3	Supermark	1977.426
FDP49	9	Regular	0.069089	Breakfast	56.3614	OUT046	1997	Small	Tier 1	Supermark	1547.319
NCB42	11.8	Low Fat	0.008596	Health and	115.3492	OUT018	2009	Medium	Tier 3	Supermark	1621.889
FDP49	9	Regular	0.069196	Breakfast	54.3614	OUT049	1999	Medium	Tier 1	Supermark	718.3982
DRI11		Low Fat	0.034238	Hard Drink	113.2834	OUT027	1985	Medium	Tier 3	Supermark	2303.668
FDU02	13.35	Low Fat	0.102492	Dairy	230.5352	OUT035	2004	Small	Tier 2	Supermark	2748.422
FDN22	18.85	Regular	0.13819	Snack Foo	250.8724	OUT013	1987	High	Tier 3	Supermark	3775.086
FDW12		Regular	0.0354	Baking Goo	144.5444	OUT027	1985	Medium	Tier 3	Supermark	4064.043
NCB30	14.6	Low Fat	0.025698	Household	196.5084	OUT035	2004	Small	Tier 2	Supermark	1587.267
FDC37		Low Fat	0.057557	Baking Goo	107.6938	OUT019	1985	Small	Tier 1	Grocery St	214.3876
FDR28	13.85	Regular	0.025896	Frozen Foo	165.021	OUT046	1997	Small	Tier 1	Supermark	4078.025
NCD06	13	Low Fat	0.099887	Household	45.906	OUT017	2007		Tier 2	Supermark	838.908

Figure 4.5 Dataset

Chapter 5

Limitations and Future Scope

In this chapter, we will discuss the limitations encountered during the development and implementation of the Real-time Sales Analytics project and the valuable learning outcomes achieved through the project's execution.

5.1 Limitations

Every project, no matter how innovative, may face certain limitations. Here are some common limitations that may be associated with the Real-time Sales Analytics project:

1. **Just Estimates :**

   The future will be unpredictable at all times. Even if the best methods of forecasting are used and every factor possible is accounted for, a prediction is still just an estimation. With 100 percent effectiveness, one can never predict future events. So even the best-laid plans can be nothing at all. This will still be one of the forecasting's greatest constraints.

2. **Data Security:**

   Assumptions, approximations, natural conditions, etc are the basis of every forecasting system. This renders those predictions inaccurate. So, the inherent weaknesses of forecasting must always be kept in mind and everyone has to be careful about being over-reliant on them.

3. Factors Time and Cost:

There is usually a lot of data and knowledge needed to make structured forecasts. And, there is a lot of time and money involved in the processing and tabulation of such results. Another aspect is also the translation of qualitative data into quantitative data. One must be cautious that the forecasting time, resources, and effort expended must not overshadow the real benefits of such forecasts.

5.2 Future Scopes

Objectives of Demand Forecasting include Financial planning, Pricing policy, Manufacturing policy, Sales and Marketing planning, Capacity planning and expansion, Manpower Planning, and Capital expenditure.:

1. **Data Collection:** Gather historical sales data, promotional event information, seasonal trends, and economic indicators.

2. **Data Preprocessing**: Clean and preprocess collected data, handle missing values, and ensure data quality.

3. **Feature Engineering:** Engineer features that capture temporal trends, promotional impacts, and external influences.

4. **Model Development:** Utilize time series analysis techniques and regression modeling to build the demand forecasting model.

5. **Inventory Planning Integration**: Integrate the forecasting model with inventory planning strategies.

Chapter 6

References

In this chapter, we provide a list of references and sources that have been used or consulted during the development of the Real-time Sales Analytics project. These references include academic papers, documentation, books, and online resources that have contributed to the project's research and implementation. Here is a sample bibliography:

1. **"Real-Time Analytics: Techniques to Analyze and Visualize Streaming Data"** by Byron Ellis, Mukesh Kumar, et al. (Book)

2. **"Machine Learning: A Probabilistic Perspective"** by Kevin P. Murphy (Book)

3. **"Introduction to Natural Language Processing"** by Jacob Eisenstein (Online Course Material)

4. **"Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions"** by Jayavardhana Gubbi, Rajkumar Buyya, et al. (Research Paper)

5. **"Tele-sales: A Systematic Review of Economic Evaluations"** by Xiangjun Kong, Xiaoxu Zhang, et al. (Research Paper)

6. **"Blockchain Technology for Sales: Facilitating the Transition to Sale-Driven Interoperability" by Maksim Izmailov, Michael D. Abramoff, and Andre Souffrant (Research Paper)**

7. **"The Genomic Data Commons"** by Josh M. Stuart, Michael L. Steinmetz, et al. (Research Paper)

8. **"Big Data Analytics in Sales: Promise and Potential"** by Usha Rani Somala and P. Radha Krishna (Research Paper)

9. **"Ethics of Artificial Intelligence and Robotics"** by Vincent C. Müller (Book Chapter)

10. **"Global sales prediction Initiatives: Progress and Prospects"** by Peter Piot, Laurie Garrett, et al. (Research Paper)

These references encompass a wide range of topics, including real-time analytics, machine learning, IoT, telemedicine, blockchain technology, genomics, data analytics, ethical considerations, and global sales prediction initiatives.