| **Project Title** | **Neural Networks for the Publishing Industry: Enhancing Customer Experience and Sales** |
| --- | --- |
| **Skills take away From This Project** | Data Cleaning and Preprocessing  Exploratory Data Analysis (EDA)  Feature Engineering for Neural Networks  Building and Training Artificial Neural Networks (ANNs)  Model Evaluation and Optimization  MYSQL usage and integreation  Deploying Deep Learning Models using AWS and Streamlit  Documentation and Reporting |
| **Domain** | **Publishing Industry (E-commerce/Bookstore Analytics)** |

**Problem Statement:**

Leverage neural networks to analyze and predict customer behavior, recommend books, and forecast demand in the publishing industry, improving both customer experience and sales.

**Business Use Cases:**

1. **Customer Churn Prediction**: Identify customers likely to stop purchasing and implement retention strategies.
2. **Personalized Book Recommendations**: Provide targeted recommendations to enhance customer satisfaction.
3. **Demand Forecasting**: Predict future book sales for inventory optimization.

**Approach:**

**1. Data Understanding & Preprocessing**

* Import SQL data into PostgreSQL.
* Normalize tables and handle missing data.
* Convert raw data into a format suitable for machine learning.

**2. Exploratory Data Analysis (EDA)**

* Generate statistics and identify trends.
* Visualize relationships between customers, orders, and books (optional).

**3. Feature Engineering**

* Transform and encode categorical features (e.g., genres, locations).
* Aggregate customer behavior metrics (e.g., purchase frequency).
* Create temporal features for forecasting.

**4. Model Development**

* Build ANN models for:
  + **Churn Prediction**: Predict whether a customer will churn.
  + **Genre Prediction**: Classify books into genres using metadata.
  + **Demand Forecasting**: Use time-series data to predict future book sales.

**5. Model Evaluation**

* Evaluate models using the following metrics:
  + **Churn Prediction**: Accuracy, Precision, Recall, F1-Score.
  + **Genre Prediction**: Accuracy, Confusion Matrix, Precision, Recall, F1-Score.
  + **Demand Forecasting**: MAE, RMSE.

**6. Deployment**

* Deploy trained models to AWS using EC2.
* Create an interactive frontend using Streamlit to:
  + Display recommendations.
  + Provide churn predictions.
  + Visualize demand forecasts.

**Results:**

 A functional system for predictions, recommendations, and demand forecasting.

 Insights into customer behavior and sales patterns.

 Improved operational efficiency in inventory management.

**Project Evaluation metrics:**

** For Churn Prediction:**

* **Accuracy, Precision, Recall, F1-Score**

** For Genre Prediction:**

* **Accuracy, Confusion Matrix, Precision, Recall, F1-Score**

** For Demand Forecasting:**

* **MAE (Mean Absolute Error), RMSE (Root Mean Squared Error)**

**Technical Tags:**

 Data Cleaning

 Feature Engineering

 Artificial Neural Networks (ANN)

 Postgresql

 Churn Prediction

 Time-Series Forecasting

 AWS Deployment

 Streamlit for Frontend

**Data Set:**

The dataset will be structured using the SQL scripts provided.

**Data Set Explanation:**

**Tables and Data Description**

**1. author Table**

* **Purpose**: Contains information about authors.
* **Columns**:
  + author\_id: Unique identifier for each author (e.g., 1, 2).
  + author\_name: Name of the author (e.g., "J.K. Rowling", "George Orwell").
* **Role in Dataset**: Provides metadata about authors who contribute to books.
* **Volume**: Depends on the number of distinct authors in the system.

**2. publisher Table**

* **Purpose**: Contains information about publishers.
* **Columns**:
  + publisher\_id: Unique identifier for each publisher (e.g., 1, 2).
  + publisher\_name: Name of the publisher (e.g., "Penguin Random House", "HarperCollins").
* **Role in Dataset**: Provides metadata about the organizations responsible for publishing books.
* **Volume**: Depends on the number of publishers in the system.

**3. book\_language Table**

* **Purpose**: Defines the available languages for books.
* **Columns**:
  + language\_id: Unique identifier for each language (e.g., 1, 2).
  + language\_code: Short code for the language (e.g., EN for English, FR for French).
  + language\_name: Full name of the language (e.g., "English", "French").
* **Role in Dataset**: Maps books to the languages they are available in.
* **Volume**: Typically a limited number of languages.

**4. book Table**

* **Purpose**: Contains metadata about books.
* **Columns**:
  + book\_id: Unique identifier for each book (e.g., 1, 2).
  + title: Title of the book (e.g., "1984", "Harry Potter and the Philosopher's Stone").
  + isbn13: International Standard Book Number (e.g., 9781234567890).
  + language\_id: Links to book\_language (e.g., 1 for English).
  + num\_pages: Number of pages in the book (e.g., 300).
  + publication\_date: Date when the book was published (e.g., 2001-06-26).
  + publisher\_id: Links to publisher (e.g., 1 for Penguin Random House).
* **Role in Dataset**: Core table that captures detailed information about books.
* **Volume**: High, as it represents individual books in the system.

**5. book\_author Table**

* **Purpose**: Defines the many-to-many relationship between books and authors.
* **Columns**:
  + book\_id: References book.book\_id (e.g., 1, 2).
  + author\_id: References author.author\_id (e.g., 1, 2).
* **Role in Dataset**:
  + Supports cases where books are written by multiple authors.
  + Links books to their respective authors.
* **Volume**: High, as many books can have multiple authors and vice versa.

**6. address\_status Table**

* **Purpose**: Tracks the status of addresses.
* **Columns**:
  + status\_id: Unique identifier for the status (e.g., 1 for Active).
  + address\_status: Status name (e.g., "Active", "Inactive").
* **Role in Dataset**: Provides a status reference for addresses.
* **Volume**: Small, with a limited number of statuses.

**7. country Table**

* **Purpose**: Stores information about countries.
* **Columns**:
  + country\_id: Unique identifier for each country (e.g., 1, 2).
  + country\_name: Full name of the country (e.g., "United States").
* **Role in Dataset**: Reference data for addresses.
* **Volume**: Small, with one entry per country.

**8. address Table**

* **Purpose**: Stores customer and shipping addresses.
* **Columns**:
  + address\_id: Unique identifier for each address.
  + street\_number: Number of the street (e.g., "123").
  + street\_name: Name of the street (e.g., "Main Street").
  + city: City of the address (e.g., "New York").
  + country\_id: Links to country.
* **Role in Dataset**: Provides detailed address information for customers and orders.
* **Volume**: Moderate, depending on the number of customers and orders.

**9. customer Table**

* **Purpose**: Contains information about customers.
* **Columns**:
  + customer\_id: Unique identifier for each customer.
  + first\_name: First name of the customer (e.g., "John").
  + last\_name: Last name of the customer (e.g., "Doe").
  + email: Email address of the customer.
* **Role in Dataset**: Provides metadata about customers who place orders.
* **Volume**: Moderate, depending on the number of customers.

**10. customer\_address Table**

* **Purpose**: Tracks the relationship between customers and their addresses.
* **Columns**:
  + customer\_id: References customer.
  + address\_id: References address.
  + status\_id: References address\_status.
* **Role in Dataset**: Links customers to multiple addresses with status information.
* **Volume**: High, as customers can have multiple addresses.

**11. shipping\_method Table**

* **Purpose**: Contains available shipping methods.
* **Columns**:
  + method\_id: Unique identifier for each shipping method (e.g., 1).
  + method\_name: Name of the shipping method (e.g., "Standard Shipping").
  + cost: Cost of the shipping method (e.g., 5.99).
* **Role in Dataset**: Provides options for shipping books to customers.
* **Volume**: Small, with limited shipping methods.

**12. cust\_order Table**

* **Purpose**: Tracks orders placed by customers.
* **Columns**:
  + order\_id: Unique identifier for each order (Auto-Increment).
  + order\_date: Date and time the order was placed.
  + customer\_id: Links to customer.
  + shipping\_method\_id: Links to shipping\_method.
  + dest\_address\_id: Links to address.
* **Role in Dataset**: Core table for order management.
* **Volume**: High, representing all orders.

**13. order\_status Table**

* **Purpose**: Tracks the status of orders.
* **Columns**:
  + status\_id: Unique identifier for the status (e.g., 1 for Pending).
  + status\_value: Status name (e.g., "Pending", "Completed").
* **Role in Dataset**: Reference data for the current status of orders.
* **Volume**: Small, with limited statuses.

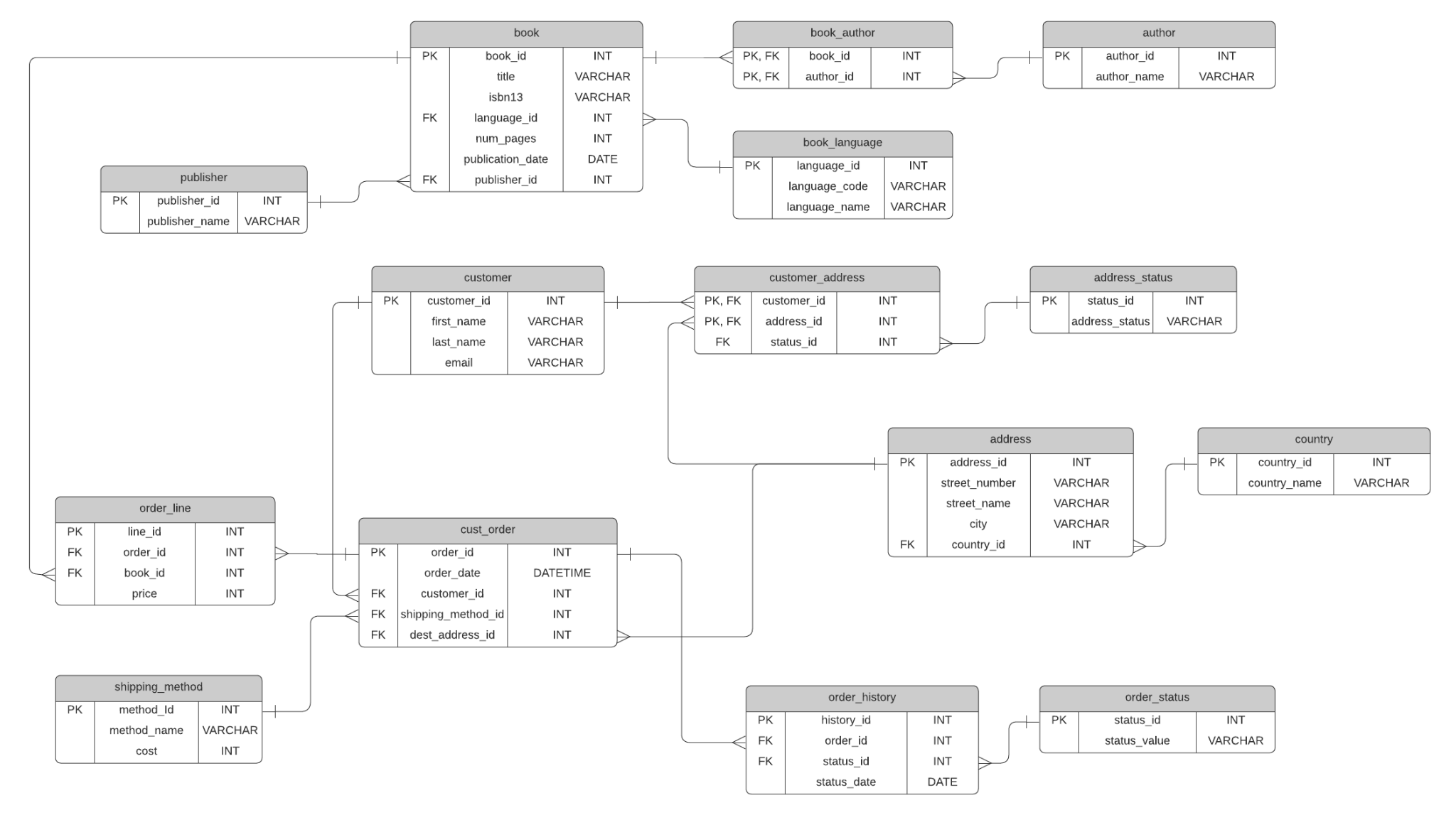
**14. order\_line Table**

* **Purpose**: Tracks details of items in an order.
* **Columns**:
  + line\_id: Unique identifier for each line item (Auto-Increment).
  + order\_id: References cust\_order.
  + book\_id: References book.
  + price: Price of the book.
* **Role in Dataset**: Tracks which books are included in each order.
* **Volume**: High, as each order can have multiple books.

**15. order\_history Table**

* **Purpose**: Tracks the history of status changes for orders.
* **Columns**:
  + history\_id: Unique identifier for each history record (Auto-Increment).
  + order\_id: References cust\_order.
  + status\_id: References order\_status.
  + status\_date: Date and time of the status update.
* **Role in Dataset**: Provides a timeline of order status changes.
* **Volume**: High, depending on the frequency of status updates.

**ER Diagram**



**Project Deliverables:**

- Cleaned and preprocessed dataset

- EDA report with visualizations (optional)

- Feature engineering code and descriptions

- Predictive models with code and explanations

- Model evaluation report

- Insights and recommendations report

- AWS Deployment with nohup

- Source code and documentation

**Project Guidelines:**

- Follow coding standards and best practices (PEP 8 for Python).

- Use version control (e.g., Git) to manage code.

- Document all steps clearly, including data cleaning, feature engineering, modeling, and evaluation.

- Ensure reproducibility of results.

**Timeline:**

**Week 1**

* **Data Cleaning and Preparation**: Import and preprocess SQL data.
* **EDA** (optional): Analyze trends and distributions.
* **Feature Engineering**: Create meaningful features.

**Week 2**

* **Model Development and Evaluation**: Train, optimize, and evaluate ANN models.
* **Deployment**: Deploy models to AWS and create the Streamlit interface.
* **Final Report**: Summarize findings and insights.

**Milestone**

* End of Week 1:

Cleaned dataset ready.

EDA report completed.

Feature engineering completed.

* End of Week 2:

Models developed and evaluated.

Deployment completed.

Final submission with report and documentation.

**PROJECT DOUBT CLARIFICATION SESSION ( PROJECT AND CLASS DOUBTS)**

**About Session:** The Project Doubt Clarification Session is a helpful resource for resolving questions and concerns about projects and class topics. It provides support in understanding project requirements, addressing code issues, and clarifying class concepts. The session aims to enhance comprehension and provide guidance to overcome challenges effectively.

**Note: Book the slot at least before 12:00 Pm on the same day**

**Timing: Tuesday, Thursday, Saturday (5:00PM to 7:00PM)**

**Booking link :**[**https://forms.gle/XC553oSbMJ2Gcfug9**](https://forms.gle/XC553oSbMJ2Gcfug9)

**LIVE EVALUATION SESSION (CAPSTONE AND FINAL PROJECT)**

**About Session:** The Live Evaluation Session for Capstone and Final Projects allows participants to showcase their projects and receive real-time feedback for improvement. It assesses project quality and provides an opportunity for discussion and evaluation.

**Note: This form will Open on Saturday and Sunday Only on Every Week**

**Timing: Monday-Saturday (11:30PM to 12:30PM)**

**Booking link :** [**https://forms.gle/1m2Gsro41fLtZurRA**](https://forms.gle/1m2Gsro41fLtZurRA)