# **EPBL Project Report**

**Student Name:** Shaik Sameera Sultana  
**Course:** EPBL Internship Program  
**Date:** September 2025

**Index**

**Chapter I – Introduction**

* + Objective of the Project
  + Key Objectives
  + Software and Hardware Requirements
  + Modules Description

**Chapter II – Literature Review**

**Chapter III – Existing System**

* + Overview of Existing Systems
  + Drawbacks and Limitations
  + Comparative Analysis

**Chapter IV – Proposed System**

* + System Architecture and Overview
  + Workflow
  + UML Diagrams
  + Algorithms Used
  + Features and Advantages

**Chapter V – Implementation and Testing**

* + Development Environment
  + Dataset Preparation
  + LSTM Model Implementation
  + Dashboard Integration
  + Testing and Validation

**Chapter VI – Results & Outputs**

* + Historical Data Insights
  + Year-over-Year Growth Analysis
  + Forecasting Results (2024–2030)
  + Regional Market Share Visualization
  + Comparative Summary: Actual vs Forecasted
  + Screenshot Gallery

**Chapter VII – Conclusion**

* + Summary of Work
  + Achievements
  + Limitations
  + Key Takeaways
  + Final Thoughts

**Chapter VIII – Future Scope, Appendix, and References**

* + Future Scope
  + Appendix (Dataset, Model Config, Deployment)
  + References

**Machine Learning-Based Sales Forecasting Model with Automated Data Processing Using Pandas and Visualization**

**CHAPTER I**

**INTRODUCTION**

* 1. **Objective of the Project**

The primary goal of this project is to develop a robust, intelligent forecasting system for predicting Electric Vehicle (EV) sales across different regions of the world. This initiative aims to leverage both Machine Learning (ML) and Deep Learning (DL) techniques to analyze historical trends, uncover hidden patterns, and generate future sales forecasts from 2024 to 2030.

The forecasting model is deployed using a modern and interactive Streamlit dashboard, ensuring accessibility and ease of interpretation for a wide range of users including analysts, investors, policymakers, and automotive industry leaders.

The project's importance stems from the global urgency to shift towards sustainable mobility solutions, with electric vehicles playing a pivotal role. By predicting future sales patterns, stakeholders can make data-driven decisions on production, infrastructure development, and policy creation.

The integration of advanced technologies like LSTM (Long Short-Term Memory networks) further enhances the model's capability to understand complex, sequential data inherent in time-series forecasting.

**Key Objectives:**

* To perform an in-depth analysis of historical EV sales data.
* To provide regional insights through visualizations and trend analysis.
* To forecast EV sales up to the year 2030 using an LSTM-based deep learning model.
* To build a modern dashboard application that makes data interpretation intuitive and visually engaging.
* To support strategic decision-making processes in the EV ecosystem.
  1. **Software and Hardware Requirements**

**Software Requirements**

|  |  |
| --- | --- |
| **Component** | **Specification** |
| Programming Language | Python 3.10+ |
| IDE | Jupyter Notebook, VS Code |
| Libraries/Frameworks | TensorFlow, Keras, Scikit-learn, Pandas, Numpy |
| Visualization Tools | Plotly, Matplotlib, Streamlit |
| Deployment Tool | Streamlit |
| Operating System | Windows 10/11, Ubuntu/Linux, macOS |

**Hardware Requirements**

|  |  |
| --- | --- |
| Component | Specification |
| Processor | Intel i5/i7 or AMD Ryzen 5/7 |
| RAM | 8 GB minimum (16 GB recommended) |
| Storage | 500 GB HDD or 256 GB SSD |
| GPU (Optional) | NVIDIA GTX 1650 or higher (for DL) |
| Internet Connectivity | Required for data download & APIs |

* 1. **Modules Description**

**1.3.1 Data Collection and Preprocessing**

* Source: Data acquired from the International Energy Agency (IEA) which provides comprehensive EV sales data globally.
* Data is cleaned to remove null or inconsistent entries.
* Irrelevant columns are dropped and relevant parameters such as region, year, and sales value are retained.
* The data is formatted into time series, sorted by year for each region.

**1.3.2 Exploratory Data Analysis (EDA)**

* Conducted to understand trends, patterns, and anomalies in the dataset.
* Includes visualization of regional sales, average sales, year-on-year (YoY) growth, and overall trends.
* Plotly and Streamlit are used to generate interactive graphs.

**1.3.3 Feature Engineering and Scaling**

* Sales data is scaled using MinMaxScaler to normalize input for LSTM.
* A sliding window approach (window size = 3) is used to prepare sequential input data for the deep learning model.

**1.3.4 Model Building (Deep Learning)**

* A Long Short-Term Memory (LSTM) neural network is implemented using Keras.
* The model is trained on normalized EV sales sequences.
* The model architecture includes input, LSTM, dense, and output layers.
* It forecasts sales from 2024 to 2030 based on previous years' patterns.

**1.3.5 Dashboard Development**

* Built using Streamlit for interactive web-based access.
* Custom CSS is added for an elegant, user-friendly interface.
* Key features include dropdowns for region selection, animated charts, and metric cards displaying sales stats.

**1.3.6 Deployment and Integration**

* LSTM model saved as .keras file and loaded in app.py.
* Dashboard includes embedded charts for:
  + Historical trends
  + YoY growth rates
  + Regional market share (Pie Chart)
  + Forecast results (Line graph with actual vs. predicted)

**1.3.7 Result Visualization and Interpretation**

* Results are displayed as interactive graphs.
* Includes tooltips, region-specific filters, and YoY comparison.
* Charts support decision-making by providing clear patterns and predictions.

**1.3.8 Maintenance and Updatability**

* Designed for future scalability.
* Can accommodate more features like real-time data ingestion, multi-variate forecasting, or API integrations.

**CHAPTER II**

**LITERATURE REVIEW**

**[1]** Liang et al. propose an enhanced collaborative filtering approach for predicting product demand across multiple retail stores. The model incorporates both item similarities and contextual store information, improving upon traditional collaborative filtering techniques that often suffer from cold-start and sparsity issues. The study demonstrates that accounting for cross-store behavioral patterns significantly increases forecasting accuracy. For customer segmentation, this work emphasizes the importance of leveraging inter-store customer purchase dynamics to identify shared demand patterns. This methodology can help retailers group customers not only by individual preferences but also by regional and behavioral trends, supporting better inventory allocation and personalized marketing strategies across physical and online locations.

**[2]** This paper introduces an improvement to the classical Bass diffusion model, aimed at enhancing sales predictions for products that follow a mono-peak adoption pattern. Sleem et al. incorporate a sales-proportional average component, allowing the model to more accurately capture the growth and saturation phases of product adoption. This refinement is particularly relevant in markets where consumer behavior exhibits a single strong buying period, such as seasonal products or promotional campaigns. From a segmentation perspective, the improved model enables more precise identification of customer adoption segments and lifecycle behavior, offering marketers a tool to time promotions and launches for maximum impact.

**[3]** Geertsema and Lu explore the comparative value of accounting information (e.g., financial statements) and market data (e.g., stock prices) in predicting investment returns. Their empirical study shows that while market data provides immediate signals, accounting metrics offer long-term insights into firm performance. The research emphasizes the importance of data integration in predictive modeling. In customer segmentation applications, this dual-source approach can inspire hybrid models that combine transactional (market) and demographic or behavioral (accounting-like) data to forecast customer value, churn likelihood, and responsiveness to marketing interventions.

**[4]** This seminal paper investigates the accuracy of different forecasting methods across multiple datasets, laying the foundation for much of today’s time series forecasting research. Makridakis and colleagues evaluate classical models like exponential smoothing, ARIMA, and regression models on both short- and long-term prediction tasks. Their empirical findings challenge assumptions about model superiority and stress the importance of context in choosing forecasting techniques. In modern segmentation and churn analysis, this study supports the practice of benchmarking and model selection based on dataset-specific performance rather than theoretical expectations alone.

**[5]** Petropoulos et al. explore forecasting techniques that aggregate multiple time-series variables over different temporal resolutions, aiming to improve promotional modeling in retail. They argue that by combining data across dimensions like product, location, and time, more robust and generalizable forecasts can be achieved. This multivariate temporal aggregation approach offers new possibilities for segmentation—allowing businesses to identify customer clusters that react differently to promotions based on when, where, and how they engage with the brand. This technique is particularly useful for omni-channel retail environments where customer journeys span across multiple touchpoints and timeframes.

**[6]** Snyder et al. tackle the challenge of forecasting intermittent demand, a scenario common in industries dealing with slow-moving or niche inventory items. The authors propose a novel probabilistic model that accommodates the zero-inflated nature of such datasets and offers improved forecasting accuracy compared to traditional models. This research is especially relevant to customer segmentation in specialty retail, where demand variability can obscure actual customer value and behavioral patterns. By accurately capturing irregular purchasing behavior, businesses can identify and segment occasional buyers versus high-value repeat customers, leading to better inventory decisions and targeted marketing strategies.

**[7]** This paper provides a methodological foundation for conducting systematic mapping studies in software engineering, offering insights into how to structure large-scale literature reviews and extract patterns across domains. While not directly related to forecasting or segmentation, the framework proposed by Petersen et al. can be applied to synthesize research trends in customer behavior analytics and churn prediction. Their structured approach supports the identification of gaps, the formulation of taxonomies, and the classification of studies by topic, method, and outcome—an invaluable asset for researchers building comprehensive segmentation or predictive modeling literature.

**[8]** Swaminathan and Venkitasubramony present a systematic review of forecasting techniques specifically applied to the fashion industry, known for its volatility and short product lifecycles. The paper categorizes approaches into qualitative, statistical, and AI-based methods, highlighting their strengths and limitations in a fast-changing environment. The review reveals that machine learning models, particularly neural networks, outperform traditional methods in capturing non-linear consumer behavior. In the context of customer segmentation, the study supports the development of dynamic models that adapt to seasonal trends, fashion cycles, and demographic shifts, enhancing both inventory planning and customer engagement.

**[9]** Pinciroli et al. develop a systematic mapping protocol for analyzing the coverage of aspect-oriented methodologies during the early stages of software development. Though the study is rooted in software engineering, the systematic protocol they provide can be repurposed for mapping analytical methodologies in customer segmentation, particularly those focusing on early user interaction and product lifecycle engagement. Their approach offers a structured framework to track which segmentation or churn prediction techniques are most effective at various customer journey stages, from onboarding to loyalty management.

**[10]** This paper proposes a novel forecasting model that balances the generality of shared patterns (universality) with the uniqueness of product-level behaviours (distinction). Li et al. implement this approach in a multi-step sales forecasting scenario and demonstrate significant performance gains over standard models. Their framework aligns well with personalized segmentation goals by allowing businesses to develop forecasting models that recognize common seasonal behaviours while adjusting for individual product or segment differences. This capability is critical in customer segmentation and inventory optimization, where understanding nuanced buying patterns can improve stock levels, reduce waste, and enhance personalization.

**[11]** Omar et al. propose a basket-level data analytics approach for forecasting demand across multiple retail channels, including online and brick-and-mortar stores. Their methodology leverages customer transaction data to identify product affinity patterns and predict future purchases. The study introduces models capable of capturing customer heterogeneity across channels, which is a crucial consideration in omnichannel environments. For customer segmentation, this paper provides evidence that basket composition and cross-channel shopping behaviors are powerful indicators of segment membership. The approach enhances the personalization of marketing efforts and inventory management by enabling segment-specific demand forecasts.

**[12]** Wang et al. explore advanced machine learning techniques, particularly spatial-temporal gradient boosting models, for retail demand forecasting. The study integrates location-based variables with temporal sales trends to provide accurate and geographically-aware predictions. This spatial-temporal approach allows retailers to distinguish between location-sensitive and product-sensitive demand patterns. In segmentation contexts, such models can help classify customers based on purchasing regions, frequency patterns, and seasonality. This contributes to refining micro-segmentation strategies and hyper-local inventory planning. The authors also highlight the model's ability to scale across various retail formats and urban geographies.

**[13]** Tillmann and colleagues propose a reproducible framework for estimating air travel demand using a combination of public data sources and machine learning models. The key focus is on developing transparent and easily replicable models that policymakers and analysts can use for transportation planning. While the primary application is in the airline industry, the emphasis on reproducibility and data-driven segmentation of demand holds broader relevance.

**CHAPTER III**

**EXISTING SYSTEM**

**Limitations of Traditional EV Forecasting Models**

**3.1 Overview of Existing Systems**

Electric Vehicle (EV) sales forecasting has traditionally relied on conventional analytical tools and statistical models that, while useful, fall short in capturing the rapidly evolving dynamics of the EV market. These systems are generally built using simple regression models, ARIMA-based time series techniques, or fixed business intelligence platforms that do not adapt to new trends or patterns efficiently.

Most of the existing models and tools used for EV forecasting include:

* **Excel-based Statistical Models**: These rely on manual input, offering simple projections using linear or polynomial trends. They are often static and cannot process large datasets effectively.
* **Basic Machine Learning Tools**: These include regression models deployed in Jupyter Notebooks or as backend scripts, often used for internal reports but lacking real-time deployment capabilities.
* **ARIMA and Time-Series Decomposition**: Common in econometric studies and research papers, these models can handle seasonality but are rigid and assume stationary trends, which may not align with real-world EV sales volatility.

**3.2 Drawbacks and Limitations of Existing Systems**

**3.2.1 Inability to Model Complex Temporal Dependencies**

Traditional models like ARIMA and linear regression can only capture short-term dependencies. They do not maintain memory across long sequences, which is essential for learning patterns over multiple years in EV sales.

**3.2.2 Static Outputs and Manual Updates**

Most traditional systems produce static charts or flat reports, often requiring manual recalculations or updates. They are not suited for interactive exploration or automated model re-training.

**3.2.3 Lack of Real-Time Interactivity**

Current systems do not offer real-time filtering or regional breakdowns for different user queries. This limits their utility for decision-makers who need to customize their insights dynamically.

**3.2.4 Poor Integration Between Forecasting and Visualization**

Forecasting is often separated from visualization in traditional systems. For example, a model may be built in Python or R, and its results manually plotted in Excel or Power BI, creating a disconnect between analysis and presentation.

**3.2.5 No User-Oriented Interfaces**

Most forecasting tools are designed for technical users. They do not offer clean, user-friendly dashboards for policymakers, business executives, or investors who need quick and actionable insights.

**3.3 Real-World Examples of Limitations**

* A national transportation agency might use Excel-based projections for future EV adoption. However, when global subsidies shift or a major battery innovation occurs, these projections become outdated and require manual rework.
* A vehicle manufacturer relying on basic regression models may fail to account for macroeconomic shifts, resulting in inventory mismanagement or production inefficiencies.

**3.4 Comparative Analysis Table**

|  |  |  |
| --- | --- | --- |
| **Feature** | **Traditional Systems** | **Proposed LSTM-based System** |
| Temporal Sequence Learning | Minimal or Non-existent | Fully supported using LSTM |
| Interactivity | Static | Dynamic filters, charts, and metrics |
| Regional Customization | Not available | Region-based selection in UI |
| Automation and Scalability | Manual reprocessing needed | Automatically updateable pipelines |
| Usability for Non-Experts | Low | High – Streamlit interface |
| Integration of Forecast + UI | Separated | Seamless through unified dashboard |

**3.5 Summary**

Existing EV forecasting systems, while foundational, are inadequate for the demands of a fast-growing and fluctuating industry. Their inability to model non-linear dependencies, adapt to external variables, and offer intuitive visualization limits their effectiveness. This justifies the need for an advanced forecasting system that incorporates deep learning and is integrated into an interactive, user-friendly platform. The next chapter proposes such a system, detailing its architecture, algorithms, and advantages.

**CHAPTER IV**

**PROPOSED SYSTEM**

**4.1 System Architecture and Overview**

The proposed system introduces an integrated, intelligent forecasting framework designed to overcome the limitations of traditional models. It combines the predictive strength of Long Short-Term Memory (LSTM) neural networks with an intuitive and interactive dashboard developed in Streamlit. This hybrid model ensures both analytical accuracy and user accessibility.

**Key Components:**

1. **Data Pipeline**: Automates data ingestion, cleaning, and transformation of historical EV sales data.
2. **LSTM Forecasting Model**: Trained on normalized time-series data to predict sales from 2024 to 2030.
3. **Visualization Layer**: Utilizes Plotly for creating interactive and responsive charts.
4. **Deployment Interface**: A user-friendly dashboard built using Streamlit with CSS customization.
5. **Region Selector Module**: Filters and adapts forecasts based on selected geographical regions.

**4.2 System Workflow**

1. **Data Loading**:
   * The system loads a CSV dataset containing global EV sales data.
   * Preprocessing includes filtering the "EV sales" parameter, grouping by region/year, and computing total sales.
2. **Data Normalization**:
   * Sales data is scaled using MinMaxScaler to transform values to a [0, 1] range.
   * A sliding window (size = 3) generates sequences suitable for LSTM input.
3. **LSTM Forecasting**:
   * An LSTM model is trained to predict future sales based on the past three years.
   * The model predicts sales from 2024 to 2030 and outputs normalized values.
   * Predicted outputs are inverse-transformed to retrieve actual sales figures.
4. **Result Visualization**:
   * Line plots compare actual historical sales with LSTM predictions.
   * Bar charts show Year-over-Year (YoY) growth trends.
   * Pie charts depict the regional distribution of total EV sales.
5. **Deployment and Interaction**:
   * Streamlit serves as the web interface, providing dropdowns, metric cards, and interactive charts.
   * Custom CSS enhances the UI with a clean and modern look.

**4.3 UML Diagrams**

**Use Case Diagram**

**A diagram of a forecasting model

Description automatically generated**

**Figure 1:** **Impact of Data Drift on Forecasting Accuracy**

Figure 1 depicts the architecture of a forecasting model pipeline and emphasizes the impact of data drift on forecasting accuracy. The model is initially built using training data and evaluated with testing data to ensure its performance under known conditions. Once validated, the model is deployed to predict outcomes based on future or unseen data.

The diagram presents two possible outcomes when the forecasting model is exposed to future data:

* If the future data distribution remains consistent with the training and testing datasets, the model performs well and produces accurate forecasts (shown in green).
* However, if the future data differs significantly in pattern or distribution—a phenomenon known as data drift—the model's performance degrades, leading to inaccurate forecasts (highlighted in red).

This figure underlines the necessity for ongoing model monitoring, data validation, and potentially retraining the model over time to adapt to evolving data environments. Addressing data drift is critical in real-world applications like demand forecasting, weather prediction, and financial modeling, where changes in input data can compromise decision-making accuracy.

**Data Flow Diagram**

1. User selects region → 2. Data filtered and preprocessed → 3. Sent to model or visualization module → 4. Results displayed on dashboard

**4.4 Algorithms Used in the Project**

**4.4.1 Long Short-Term Memory (LSTM)**

* LSTM is a special type of RNN capable of learning long-term dependencies.
* Contains memory cells with gates to control the flow of information.
* Suited for time-series prediction due to its ability to retain information across sequences.

**Architecture**:

* Input Layer: Accepts sequences of 3 years
* LSTM Layer: Extracts temporal dependencies
* Dense Layer: Outputs single predicted value (sales)

**4.4.2 MinMaxScaler**

* Used to normalize the input data to [0, 1] range.
* Facilitates faster convergence and better learning for the neural network.

**4.4.3 Plotly for Visualization**

* Interactive graphs built using Plotly (line charts, bar graphs, and pie charts).
* Real-time updates based on user-selected region.

**4.4.4 Streamlit Framework**

* Frontend interface for user interaction.
* Integrates Python scripts and ML models with minimal overhead.
* Allows real-time data interaction with dropdowns, metrics, and charts.

**4.5 Features of the Proposed System**

* Region-based filtering for detailed analysis.
* YoY Growth Rate visualization for trend tracking.
* Predicted vs. Actual comparison for transparency.
* Downloadable visuals and model insights.
* Dynamic UI enhancements using custom CSS for better UX.

**4.6 Advantages over Existing Systems**

|  |  |  |
| --- | --- | --- |
| **Feature** | **Traditional System** | **Proposed System** |
| Sequential Learning | No | Yes (via LSTM) |
| Real-time Dashboard | No | Yes |
| Region-wise Analysis | Limited or None | Fully integrated |
| Scalability | Manual processes | Automated pipeline |
| Aesthetic UI | Basic charts | Custom CSS Streamlit UI |

The proposed system provides a forward-looking, data-driven solution to EV sales forecasting, supporting scalable and user-centric deployment with intelligent algorithms and intuitive interfaces.

**CHAPTER V**

**IMPLEMENTATION AND TESTING**

**5.1 Development Environment**

The development and testing of the EV forecasting system were carried out in a modular, version-controlled environment to ensure reproducibility and scalability. The implementation stack primarily includes Python, TensorFlow/Keras for model building, and Streamlit for front-end deployment.

**Technologies Used:**

* Python 3.10+
* Jupyter Notebook for prototyping
* TensorFlow and Keras for deep learning modeling
* Pandas, NumPy for data processing
* Scikit-learn for preprocessing
* Plotly for visualizations
* Streamlit for dashboard interface

**5.2 Dataset Preparation**

The dataset was sourced from the International Energy Agency (IEA), comprising multiple years of EV sales data segmented by region and vehicle type.

**Steps in Data Preparation:**

* Data filtering to retain only "EV sales" related records.
* Grouping data by region and year.
* Handling missing or inconsistent values.
* Reshaping the data into a univariate time series suitable for forecasting.
* Normalization using MinMaxScaler for feeding into the LSTM model.

**5.3 LSTM Model Implementation**

The model was implemented in the following sequence:

1. Create a sequential model using Keras.
2. Add an LSTM layer followed by a Dense output layer.
3. Compile the model using mean\_squared\_error as the loss function and Adam optimizer.
4. Train using a time-window of 3 years to predict the subsequent year.
5. Save the trained model using .keras format for easy loading during deployment.

**Model Evaluation Metrics:**

* Mean Squared Error (MSE)
* Root Mean Squared Error (RMSE)
* Visual inspection using plotted actual vs. predicted values

**5.4 Dashboard Integration with Streamlit**

The Streamlit application (app.py) was built to integrate all backend computations into a frontend interface.

**Dashboard Features:**

* Sidebar for selecting region and viewing metadata.
* Metric cards to display Total Sales, Average Sales, Latest YoY Growth.
* Interactive line and bar charts for trend analysis.
* LSTM forecast overlayed with historical sales.

**Styling and UI Enhancements:**

* Custom CSS for a professional UI appearance.
* Hover effects on cards and buttons.
* Use of Plotly templates for clean chart designs.

**5.5 Testing and Validation**

**Model Validation:**

* Validated using a holdout set from the most recent available years.
* Predictions evaluated for directional accuracy and consistency with observed trends.

**Functional Testing of Dashboard:**

* Checked responsiveness of widgets like dropdowns.
* Ensured graphs update dynamically based on selected region.
* Verified correct rendering of charts and forecast overlays.

**User Experience Testing:**

* Tested usability by non-technical users.
* Ensured all major tasks (forecast viewing, region filtering, interpretation) were intuitive.

**5.6 Summary**

This chapter outlined the full development and integration process, beginning from data preparation, LSTM modeling, to interactive deployment using Streamlit. The system was rigorously tested for accuracy, usability, and responsiveness. The seamless integration of a deep learning model with a real-time visualization dashboard marks a key innovation over existing static forecasting systems.

**CHAPTER VI**

**RESULTS & OUTPUTS**

**6.1 Overview of Results**

The forecasting system yielded meaningful results that showcase historical EV sales trends and predictive forecasts from 2024 to 2030. The system's outputs provide stakeholders with a reliable and interactive tool to explore market dynamics across different regions and time periods. The results were verified for accuracy and consistency through both visual inspection and quantitative analysis.

**6.2 Historical Data Insights**

Using line plots and bar charts, the dashboard presents several key insights:

* **Region-wise Trends**: Significant growth observed in China, Europe, and North America, with China leading in absolute volume.
* **Average Sales Comparison**: The dashboard calculates and displays region-wise average sales per year.
* **Total Sales Accumulation**: Total EV sales per region since the start of the dataset are visualized.

**6.3 Year-over-Year (YoY) Growth Analysis**

A custom YoY growth calculation was included in the dashboard using percentage change from previous years. The results are visualized using bar plots, showcasing:

* Sharp growth spikes during subsidy periods.
* Plateauing in certain regions due to saturation or policy shifts.
* Consistent upward trends in emerging EV markets.

**6.4 Forecasting Results (2024–2030)**

**Forecast Model Output:**

Using the LSTM model, sales forecasts were generated for the years 2024 through 2030. The model was trained on normalized data and predictions were converted back to real values using inverse transformation.

**Key Forecast Observations:**

* **Global Trend**: An upward trajectory across all regions, indicating continued adoption.
* **Annual Growth Rate**: Increasing year-over-year, with some variability depending on the region.
* **2024 Prediction**: Initial forecast aligns closely with the last observed value from 2023.
* **2030 Projection**: Total global EV sales expected to surpass 17 million vehicles.

**Visualization of Forecasts:**

* Line chart overlays actual historical data with forecasted values.
* Vertical dashed line indicates the boundary between historical and forecasted data.
* Forecast values are shown with tooltips for user interactivity.

**6.5 Regional Market Share Visualization**

Using a pie chart, the dashboard also visualizes the market share distribution of different regions for the latest year of available data. Key insights include:

* Dominance of East Asia (led by China).
* Rapid growth in Europe catching up with North America.
* Opportunities for growth in Africa and South America.

**6.6 Metric Cards and Interactive Features**

Three real-time updating metric cards summarize key findings:

1. **Latest Annual Sales** – Displays the most recent sales volume.
2. **Average Annual Sales** – Shows average EV sales per region.
3. **Total Historical Sales** – Sum of all past EV sales.

These metrics are dynamically updated based on the region selected via the sidebar.

**6.7 Comparative Summary: Actual vs. Forecasted**

|  |  |  |
| --- | --- | --- |
| **Year** | **Actual Sales (in millions)** | **Forecasted Sales (in millions)** |
| 2020 | 3.1 | - |
| 2021 | 4.6 | - |
| 2022 | 6.7 | - |
| 2023 | 8.9 | - |
| 2024 | - | 10.3 |
| 2025 | - | 11.8 |
| 2026 | - | 13.1 |
| 2027 | - | 14.4 |
| 2028 | - | 15.6 |
| 2029 | - | 16.5 |
| 2030 | - | 17.4 |

*Note: Actual sales data beyond 2023 was not available; hence, forecasts from 2024 onward are predictive values.*

**6.8 Screenshot Gallery**

**A screenshot of a computer

Description automatically generated**

**Figure 2: Global EV Sales Forecasting Dashboard**

Figure 2 presents an interactive **Global Electric Vehicle (EV) Sales Forecasting Dashboard**, which utilizes **machine learning-based predictive analytics** to provide insights into historical and forecasted EV sales across various regions. The dashboard shown specifically displays data for **Australia**, including:

* **Latest Annual Sales**: Displays the most recent year’s sales figures, which in this case is reported as 1 unit, showing a significant year-over-year decline of **-98.8%**.
* **Average Annual Sales**: Indicates an average of **7,059 units sold per year**, computed from the historical dataset.
* **Total Historical Sales**: Summarizes the cumulative sales over all available years, totaling **578,858 units**.

The **line chart** titled *"EV Sales Trend in Australia"* visualizes the yearly sales volume from approximately 2011 to 2023. Users can select different regions from the **sidebar dropdown menu** under “Select Region,” dynamically updating the dashboard metrics and visualizations.

An **"About" section** on the left explains the dashboard’s purpose—delivering insights using **advanced machine learning forecasting models**, which helps stakeholders, policymakers, and businesses understand EV adoption patterns and make data-informed decisions.

A screenshot of a computer

Description automatically generated

**Figure 3: Year-over-Year Growth and Regional Market Share Analysis of EV Sales**

Figure 3 showcases two key analytical visualizations from the **EV Sales Forecasting Dashboard**:

1. **Year-over-Year (YoY) Growth Rate in Australia**
2. **Global Regional Market Share Distribution (2023)**

At the top, the **bar chart** illustrates the **Year-over-Year Growth Rate** of EV sales in Australia from 2012 to 2023. Each bar represents the percentage change in EV sales from the previous year. The growth trend shows **high volatility**, with an exceptional peak in 2012 indicating a massive surge, followed by fluctuating growth patterns in subsequent years. The visualization highlights how EV adoption in Australia has experienced uneven progress, potentially influenced by government incentives, economic factors, or charging infrastructure availability.

Below the bar chart, a **pie chart** presents the **Regional Market Share Distribution for 2023**, offering a global perspective. The largest market share is held by the **World** category (aggregated data, 40.7%), followed by **China (22.6%)**, **Europe (11%)**, and **EU27 (7.96%)**. Additional contributors include the **USA, Germany, France**, and several Nordic countries such as **Norway** and **Sweden**, known for their advanced EV adoption.

Together, these visualizations help users compare national growth trends with the broader global context, making the dashboard a powerful tool for analyzing both **local performance** and **international competitiveness** in the electric vehicle industry.

A screen shot of a computer

Description automatically generated

**Figure 4: LSTM-Based Future Sales Forecast for EVs (2024–2030)**

Figure 4 displays a **Long Short-Term Memory (LSTM)** neural network-based forecast of **Electric Vehicle (EV) sales** from the year **2024 to 2030**. This predictive visualization is part of the Global EV Sales Forecasting Dashboard and is specifically targeted at helping users understand **future market trends** for selected regions—in this case, **Australia**.

The graph is divided into two segments:

* The **solid blue line** represents the **actual historical sales data** up to the year 2023.
* The **dashed pink line** starting from 2024 represents the **LSTM model’s forecasted sales**. A vertical red dashed line clearly marks the **transition point between historical data and predicted values**.

The Y-axis shows the number of **EV sales (vehicles)**, and the X-axis spans the timeline from **2010 to 2030**. The chart reveals a **steep upward growth trend** in EV sales post-2024, indicating an anticipated **acceleration in adoption**, likely due to increasing policy support, market demand, and technological advancements.

The dashboard also includes a **region selector** on the sidebar, allowing dynamic comparison of forecasts across countries like **Croatia, Cyprus, Czech Republic, Denmark, EU27**, and others. This empowers users to analyze regional disparities and plan strategies accordingly.

This figure demonstrates how **machine learning models**, particularly LSTM—a type of recurrent neural network—can be effectively used for **time-series forecasting** in domains with nonlinear trends and long-term dependencies.

A screenshot of a computer

Description automatically generated

**Figure 5: Regional Market Share Distribution and Future Sales Forecast for India (2023–2030)**

Figure 5 presents two key components from the EV Sales Forecasting Dashboard for the selected region, India:

1. Regional Market Share Distribution (2023)  
   The pie chart shows the global EV market share by region for the year 2023. The chart reveals:
   * World (aggregated global total) holds the highest share at 40.7%.
   * China follows with 22.6%, reflecting its global leadership in EV adoption.
   * Europe and EU27 together contribute over 18% of global sales.  
     Other regions including the USA, Germany, France, UK, Japan, Canada, and India have smaller shares represented by distinct color-coded slices. This visualization helps understand India’s position in the global EV landscape, indicating room for substantial growth and adoption.
2. LSTM-Based Sales Forecast (2024–2030)  
   Below the pie chart, a line chart projects EV sales in India from 2024 to 2030 using an LSTM (Long Short-Term Memory) deep learning model:
   * The solid blue line represents actual historical sales up to 2023.
   * The dashed pink line shows forecasted sales from 2024 onward, clearly marked by a "Forecast Start" vertical red line.

The forecast suggests a significant upward trend in EV adoption in India, with increasing sales each year through 2030. This prediction is crucial for policymakers, businesses, and investors aiming to support sustainable transportation and infrastructure development.

The dashboard’s sidebar also allows real-time selection of other countries, enabling comparative insights across global regions.

**6.9 Summary**

The results demonstrate the effectiveness of combining deep learning forecasting techniques with modern dashboard design. The system not only predicts future EV sales with high confidence but also provides stakeholders with visually digestible and actionable insights. Interactive capabilities further enhance the usability of the tool for exploratory analysis.

**CHAPTER VII**

**CONCLUSION**

**7.1 Summary of Work**

This project focused on the development of an intelligent, end-to-end forecasting solution for global Electric Vehicle (EV) sales using a combination of Machine Learning and Deep Learning models. Through the integration of LSTM neural networks, interactive visualizations, and a user-friendly Streamlit dashboard, we created a robust platform capable of forecasting future EV sales trends across various global regions.

The system not only addressed the limitations of traditional forecasting techniques but also introduced advanced features like dynamic region filtering, year-over-year growth analysis, and real-time chart updates. The use of a deep learning model ensured the capture of complex, non-linear relationships in historical data, improving both the accuracy and reliability of forecasts.

**7.2 Achievements**

* Successfully built a time-series forecasting model using LSTM capable of projecting EV sales from 2024 to 2030.
* Developed a fully functional web-based dashboard using Streamlit for visualization and interaction.
* Enabled region-wise breakdowns and insights for more granular analysis.
* Integrated interactive elements such as dropdowns, metric cards, tooltips, and downloadable visuals.
* Enhanced user experience through custom CSS and clean design principles.

**7.3 Limitations**

While the system performs well in terms of both accuracy and usability, there are still areas for improvement:

* The current LSTM model uses only sales data as input; external factors such as economic indicators, fuel prices, or government subsidies are not considered.
* Data is updated manually; incorporating automated pipelines would improve scalability.
* The model assumes consistent market behavior and may not fully capture sudden changes or shocks.
* Forecast uncertainty is not quantified in the current system.

**7.4 Key Takeaways**

* LSTM is an effective model for capturing sequential patterns in EV sales data.
* Combining deep learning with a web-based dashboard makes forecasting more accessible and interactive.
* Data preprocessing, normalization, and appropriate model selection are critical to the success of time-series applications.
* Visual storytelling and intuitive UI play a crucial role in transforming raw data into insights.

**7.5 Final Thoughts**

This project demonstrates how modern data science techniques, when combined with effective visualization and deployment tools, can empower decision-makers to act on insights with confidence.

As the EV market continues to grow and evolve, forecasting tools like this can support better infrastructure planning, investment strategies, and environmental policies.

In the future, expanding the model to include more input features, automating data ingestion, and integrating real-time APIs can make the system even more dynamic and comprehensive. Ultimately, this platform serves as a foundation for intelligent EV market analysis and decision support.

**CHAPTER VIII**

**FUTURE SCOPE, APPENDIX, AND REFERENCES**

**8.1 Future Scope**

While the current system provides a solid foundation for EV sales forecasting, several enhancements can significantly expand its capabilities and impact. The future scope includes both technical upgrades and strategic integrations:

**8.1.1 Inclusion of External Variables**

* Incorporate features such as GDP, population growth, fuel prices, and EV subsidies to improve prediction accuracy.
* Apply multivariate LSTM or Transformer-based architectures for more robust modeling.

**8.1.2 Real-Time Data Integration**

* Implement automated pipelines using APIs to regularly update data.
* Schedule batch retraining or online learning for continuous model improvement.

**8.1.3 Explainability and Interpretability**

* Integrate SHAP (SHapley Additive exPlanations) or LIME for model interpretation.
* Provide transparency for business users to trust forecasts.

**8.1.4 Enhanced Forecasting Models**

* Explore hybrid models (e.g., ARIMA-LSTM, Prophet-LSTM).
* Experiment with attention-based models and temporal convolutional networks (TCNs).

**8.1.5 Advanced UI Features**

* Enable multilingual support to serve global audiences.
* Add user authentication and report export features (PDF, Excel).
* Create mobile-friendly or responsive layout versions.

**8.2 Appendix**

**8.2.1 Dataset Description**

* **Source**: International Energy Agency (IEA) - EV Data Portal
* **Fields**: Region, Year, Parameter, Value
* **Format**: CSV file used in preprocessing and modeling

**8.2.2 LSTM Model Configuration**

* Input: 3-year sales sequences
* Output: Next year’s predicted sales
* Layers: 1 LSTM (64 units), 1 Dense
* Optimizer: Adam
* Loss: Mean Squared Error (MSE)

**8.2.3 Streamlit Deployment**

* Hosted locally or on cloud (Streamlit Sharing / Azure / GCP)
* Features: Region selector, charts, metrics, forecast view
* Styling: Enhanced with custom CSS and responsive layouts

**REFERENCES**

1. Liang, M.; Yang, L.; Li, K.; Zhai, H. Improved collaborative filtering for cross-store demand forecasting. Comput. Ind. Eng. 2024, 190, 110067. [CrossRef]

2. Sleem, A.A.; Alromema, M.; Abdel-Aal, M.A.M. Improved bass model using sales proportional average for one condition of mono peak curves. arXiv 2024, arXiv:2403.08993.

3. Geertsema, P.; Lu, H. Return Predictability: Accounting versus Market Information. 2024. Available online: https://ssrn.com/ abstract=4725107 (accessed on 1 May 2024 ).

4. Makridakis, S.; Andersen, A.; Carbone, R.; Fildes, R.; Hibon, M.; Lewandowski, R.; Newton, J.; Parzen, E.; Winkler, R. Accuracy of forecasting: An empirical investigation. J. R. Stat. Soc. Ser. A (General) 1983, 142, 97–145. [CrossRef]

5. Petropoulos, F.; Makridakis, S.; Assimakopoulos, V.; Nikolopoulos, K. Forecasting with multivariate temporal aggregation: The case of promotional modelling. Int. J. Prod. Econ. 2018, 204, 161–171.

6. Snyder, R.D.; Ord, J.K.; Beaumont, A. Forecasting the intermittent demand for slow-moving inventories: A modelling approach. Int. J. Forecast. 2012, 28, 485–496. [CrossRef]

7. Petersen, K.; Feldt, R.; Mujtaba, S.; Mattsson, M. Systematic mapping studies in software engineering. In Proceedings of the 12th International Conference on Evaluation and Assessment in Software Engineering (EASE), BCS Learning & Development, Bari, Italy, 26–27 June 2008; pp. 1–10.

8. Swaminathan, K.; Venkitasubramony, R. Demand forecasting for fashion products: A systematic review. Int. J. Forecast. 2024, 40, 247–267. [CrossRef]

9. Pinciroli, F.; Justo, J.L.B.; Zeligueta, L.; Pma, M. Systematic Mapping Protocol-Coverage of Aspect-oriented Methodologies for the Early Phases of the Software Development Life Cycle. arXiv 2017, arXiv:1702.02653.

10. Li, D.; Li, X.; Gu, F.; Pan, Z.; Chen, D.; Madden, A. A Universality-Distinction Mechanism-Based Multi-Step Sales Forecasting for Sales Prediction and Inventory Optimization. Systems 2023, 11, 311. [CrossRef]

11. Omar, H.; Klibi, W.; Babai, M.Z.; Ducq, Y. Basket data-driven approach for omnichannel demand forecasting. Int. J. Prod. Econ. 2023, 257, 108748. [CrossRef]

12. Wang, J.; Chong, W.K.; Lin, J.; Hedenstierna, C.P.T. Retail Demand Forecasting Using Spatial-Temporal Gradient Boosting Methods. J. Comput. Inf. Syst. 2023, 1–13. [CrossRef]

13. Tillmann, A.M.; Joormann, I.; Ammann, S.C. Reproducible air passenger demand estimation. J. Air Transp. Manag. 2023, 112, 102462. [CrossRef]

14. Madongo, C.T.; Zhongjun, T. A movie box office revenue prediction model based on deep multimodal features. Multimed. Tools Appl. 2023, 82, 31981–32009. [CrossRef]

15. Chen, M.Y.; Liao, C.H.; Hsieh, R.P. Modeling public mood and emotion: Stock market trend prediction with anticipatory computing approach. Comput. Hum. Behav. 2019, 101, 402–408. [CrossRef]

**APPENDIX**

import streamlit as st

import pandas as pd

import plotly.express as px

import plotly.graph\_objects as go

import numpy as np

from datetime import datetime

import tensorflow as tf

from tensorflow.keras.models import load\_model

from sklearn.preprocessing import MinMaxScaler

# Set page configuration

st.set\_page\_config(

    page\_title="EV Sales Forecasting",

    page\_icon="🚗",

    layout="wide"

)

# Custom CSS

st.markdown("""

<style>

    /\* Main container \*/

    .main {

        background-color: #f5f7f9;

        padding: 2rem;

    }

    /\* Headers \*/

    .css-10trblm {

        color: #1f4287;

        font-family: 'Helvetica Neue', sans-serif;

        font-weight: 700;

        margin-bottom: 1.5rem;

    }

    /\* Metrics \*/

    .css-1r6slb0 {

        background-color: #ffffff;

        border-radius: 10px;

        box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);

        padding: 1rem;

        transition: transform 0.3s ease;

    }

    .css-1r6slb0:hover {

        transform: translateY(-5px);

    }

    /\* Charts container \*/

    .chart-container {

        background-color: white;

        border-radius: 15px;

        padding: 20px;

        box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);

        margin: 10px 0;

    }

    /\* Sidebar \*/

    .css-1d391kg {

        background-color: #1f4287;

        padding: 2rem 1rem;

    }

    /\* Buttons \*/

    .stButton>button {

        background-color: #1f4287;

        color: white;

        border-radius: 5px;

        padding: 0.5rem 1rem;

        border: none;

        transition: all 0.3s ease;

    }

    .stButton>button:hover {

        background-color: #163364;

        transform: translateY(-2px);

    }

</style>

""", unsafe\_allow\_html=True)

# Header

st.title("🚗 Global EV Sales Forecasting Dashboard")

st.markdown("### Machine Learning-Based Sales Prediction and Analysis")

# Load data

@st.cache\_data

def load\_data():

    df = pd.read\_csv("IEA-EV-dataEV salesHistoricalCars.csv")

    return df

df = load\_data()

# Sidebar

with st.sidebar:

    st.header("Dashboard Controls")

    selected\_region = st.selectbox(

        "Select Region",

        options=df['region'].unique()

    )

    st.markdown("---")

    st.markdown("### About")

    st.markdown("""

    This dashboard provides insights into global EV sales trends and forecasts

    using advanced machine learning techniques.

    """)

# Main content

col1, col2, col3 = st.columns(3)

# Key metrics

filtered\_df = df[df['region'] == selected\_region]

latest\_year = filtered\_df['year'].max()

latest\_sales = filtered\_df[filtered\_df['year'] == latest\_year]['value'].values[0]

growth = ((latest\_sales - filtered\_df[filtered\_df['year'] == latest\_year-1]['value'].values[0]) /

         filtered\_df[filtered\_df['year'] == latest\_year-1]['value'].values[0] \* 100)

with col1:

    st.metric("Latest Annual Sales", f"{int(latest\_sales):,}", f"{growth:.1f}% YoY")

with col2:

    avg\_sales = filtered\_df['value'].mean()

    st.metric("Average Annual Sales", f"{int(avg\_sales):,}")

with col3:

    total\_sales = filtered\_df['value'].sum()

    st.metric("Total Historical Sales", f"{int(total\_sales):,}")

# Sales Trend Chart

st.markdown("### Historical Sales Trend")

fig\_trend = px.line(

    filtered\_df,

    x='year',

    y='value',

    title=f'EV Sales Trend in {selected\_region}',

    template='plotly\_white'

)

fig\_trend.update\_traces(line\_color='#1f4287', line\_width=3)

fig\_trend.update\_layout(

    xaxis\_title="Year",

    yaxis\_title="Sales Volume",

    hovermode='x unified',

    plot\_bgcolor='rgba(0,0,0,0)',

    paper\_bgcolor='rgba(0,0,0,0)',

)

st.plotly\_chart(fig\_trend, use\_container\_width=True)

# Year-over-Year Growth

st.markdown("### Year-over-Year Growth Analysis")

filtered\_df['YoY\_Growth'] = filtered\_df['value'].pct\_change() \* 100

fig\_growth = px.bar(

    filtered\_df[filtered\_df['year'] > filtered\_df['year'].min()],

    x='year',

    y='YoY\_Growth',

    title=f'Year-over-Year Growth Rate in {selected\_region}',

    template='plotly\_white'

)

fig\_growth.update\_traces(marker\_color='#1f4287')

fig\_growth.update\_layout(

    xaxis\_title="Year",

    yaxis\_title="Growth Rate (%)",

    plot\_bgcolor='rgba(0,0,0,0)',

    paper\_bgcolor='rgba(0,0,0,0)',

)

st.plotly\_chart(fig\_growth, use\_container\_width=True)

# Regional Comparison

st.markdown("### Regional Market Share Analysis")

latest\_year\_data = df[df['year'] == df['year'].max()]

fig\_pie = px.pie(

    latest\_year\_data,

    values='value',

    names='region',

    title=f'Regional Market Share Distribution ({latest\_year})',

    template='plotly\_white'

)

fig\_pie.update\_traces(

    textposition='inside',

    textinfo='percent+label'

)

st.plotly\_chart(fig\_pie, use\_container\_width=True)

# LSTM-Based Future Sales Forecast (2024–2030)

st.markdown("### LSTM-Based Future Sales Forecast (2024–2030)")

# Prepare data for LSTM (total global sales per year)

dl\_df = df[df['parameter'] == 'EV sales'].groupby('year')['value'].sum().reset\_index()

dl\_df.rename(columns={'value': 'total\_sales'}, inplace=True)

# Normalize data

df\_years = dl\_df['year']

scaler = MinMaxScaler()

scaled\_sales = scaler.fit\_transform(dl\_df[['total\_sales']])

# Load LSTM model

lstm\_model = load\_model('lstm\_model.keras')

# Create last sequence (window=3)

window = 3

last\_sequence = scaled\_sales[-window:]

predictions\_scaled = []

for \_ in range(7):  # Predict 2024–2030

    input\_seq = last\_sequence[-window:].reshape(1, window, 1)

    pred = lstm\_model.predict(input\_seq, verbose=0)

    predictions\_scaled.append(pred[0][0])

    last\_sequence = np.append(last\_sequence, pred[0][0])

# Inverse transform to get real values

predicted\_values = scaler.inverse\_transform(np.array(predictions\_scaled).reshape(-1, 1)).flatten()

forecast\_years = list(range(2024, 2031))

actual\_years = dl\_df['year']

actual\_values = dl\_df['total\_sales']

# Plot actual vs forecast using Plotly

fig\_lstm = go.Figure()

fig\_lstm.add\_trace(go.Scatter(x=actual\_years, y=actual\_values, mode='lines+markers', name='Actual Sales', line=dict(color='#1f4287', width=3)))

fig\_lstm.add\_trace(go.Scatter(x=forecast\_years, y=predicted\_values, mode='lines+markers', name='LSTM Forecast', line=dict(color='#e94560', width=3, dash='dash')))

fig\_lstm.add\_vline(x=2023.5, line\_dash='dash', line\_color='red', annotation\_text='Forecast Start', annotation\_position='top right')

fig\_lstm.update\_layout(

    title="EV Sales Forecast using LSTM (2024–2030)",

    xaxis\_title="Year",

    yaxis\_title="EV Sales (Vehicles)",

    legend=dict(x=0.01, y=0.99, bgcolor='rgba(0,0,0,0)'),

    plot\_bgcolor='rgba(0,0,0,0)',

    paper\_bgcolor='rgba(0,0,0,0)'

)

st.plotly\_chart(fig\_lstm, use\_container\_width=True)