

Optimizing Retail Forecasting Through Advanced AI Models: The Role of Prompt Engineering



Bio



- Data Scientist with 7 years of experience
- Specialist in Data Science with a focus on Forecasting at Scale
- Researcher, Author, Mentor
- Master of Science from University of Texas at Austin, Bachelor of Engineering, Mechanical

Sijo Valayakkad Manikandan

Table of Contents

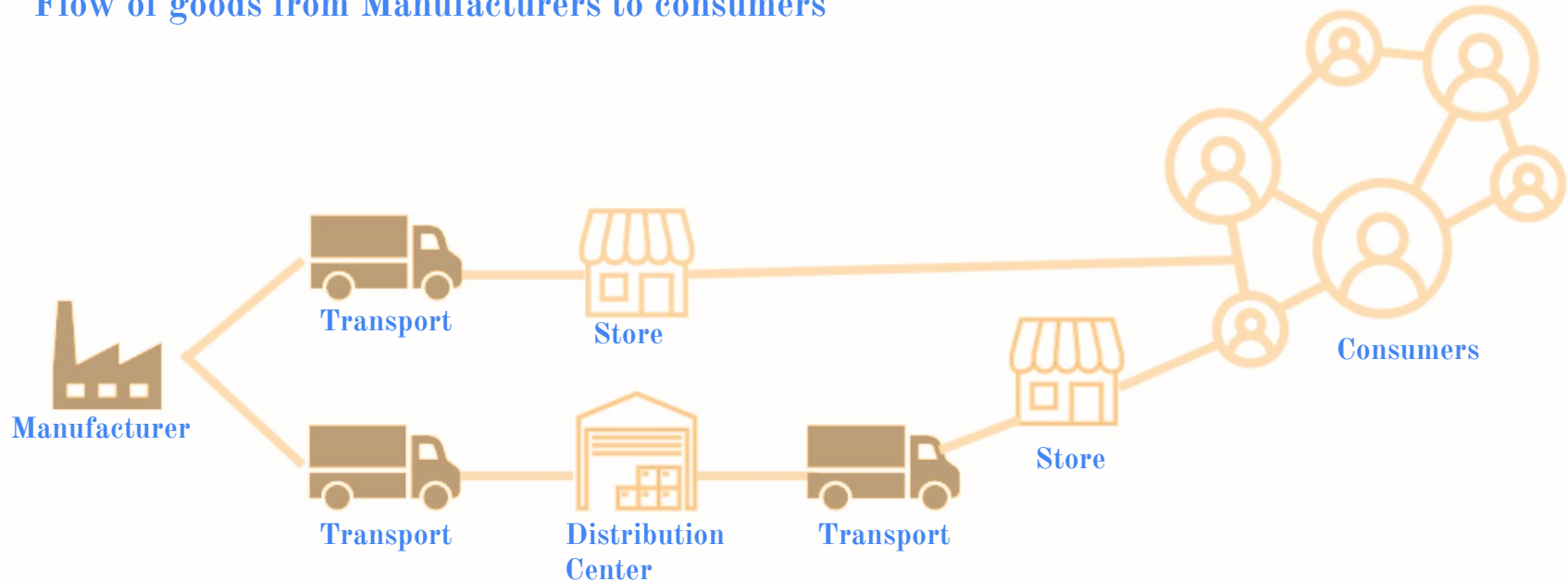
1. Why do we forecast?
2. Forecasting Challenges
3. Prompt Engineering
4. Prompt Engineering in Forecasting
5. Future of AI in Retail Forecasting



**1. Why do we
forecast?**

Retail Value Chain

Flow of goods from Manufacturers to consumers



Retail Landscape and Forecasting Necessity

- Long manufacturing lead times
- Increasing complexity in **customer preferences and behaviors**
- **Omnichannel retailing**: blending online, in-store, and hybrid experiences
- Retailers' need for **real-time inventory optimization**
- **Competition driving** the need for better decision-making tools
- **Forecasting as a critical component** for cost control and maximizing sales



2. Challenges



1 - 3 Weeks	3 - 12 weeks	3 - 24 Months	> 24 Months
<ul style="list-style-type: none">• Store Replenishment• Sales & Operations Execution• Smooth Delivery Flows• Distribution Center Replenishment• Inventory allocation (end-of-season clearance, seasonal promotion)	<ul style="list-style-type: none">• Workforce Optimization• Capacity Management• Sales & Operations Execution• Smooth Delivery Flows• Distribution Center Replenishment• Inventory allocation (end-of-season clearance, seasonal promotion)	<ul style="list-style-type: none">• Assortment Planning• Space Optimization• Long-lead time purchasing• Sales & Operation Planning (S&OP)	<ul style="list-style-type: none">• Strategic Planning• Product Design & Development• Contractual Obligations

Forecasting Challenges

- **Granularity:** Forecasting at SKU, store, region, and channel levels.
- **Short-term vs. long-term:** Immediate vs. strategic decision-making.
- **Data fragmentation:** Integrating disparate sources for cohesive forecasts.
- **Demand volatility:** Handling fluctuations in customer behavior.
- **Trade-offs** between accuracy and operational efficiency.



Impact of COVID-19 on Forecasting

- **Drastic shifts in consumer behavior** due to lockdowns and economic uncertainty.
- Massive **supply chain disruptions** affecting inventory management
- Increased importance of **real-time forecasting** and **adaptability**
- Accelerated **adoption of AI/ML models** to handle unpredictability
- Lessons learned: Need for **flexible, resilient** forecasting systems.



3. Prompt Engineering

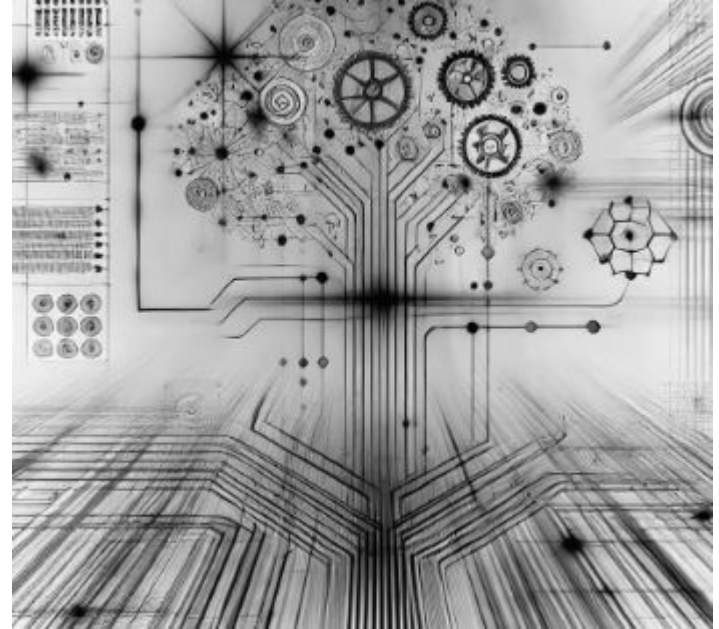
Introduction to Prompt Engineering

What is Prompt Engineering?

- The process of guiding generative AI solutions to produce high-quality, desired outputs.
- It involves selecting appropriate formats, phrases, and structures.

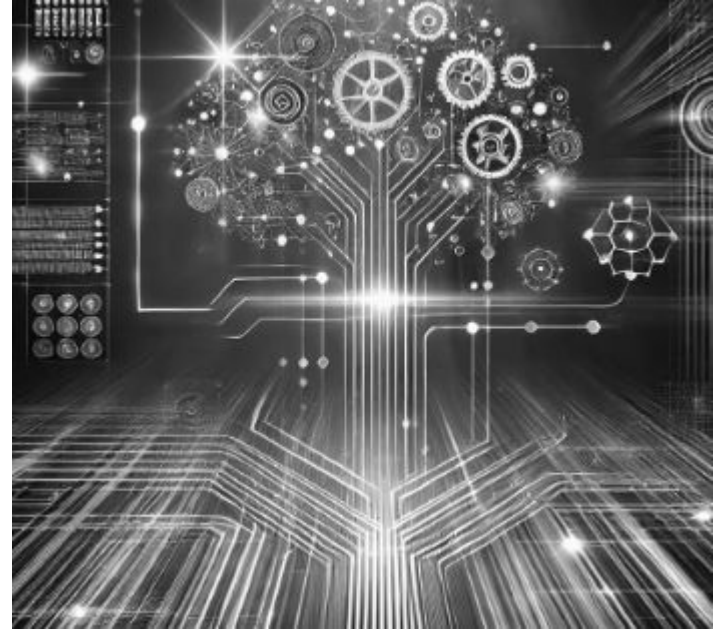
Purpose of Prompt Engineering

- Ensures meaningful interaction between AI models and users.
- Aids in creating a prompt library for scalable and reusable AI solutions.



Key Techniques in Prompt Engineering

- 1. Chain-of-Thought Prompting:** Breaks complex questions into logical steps for improved reasoning.
- 2. Tree-of-Thought Prompting:** Expands on multiple ideas before reaching a conclusion.
- 3. Self-Refine Prompting:** Involves solving, critiquing, and refining answers iteratively.



1. Chain-of-Thought Prompting

Purpose: Break down the forecasting task into logical, step-by-step components.

Prompt:

"Develop a retail sales forecasting model by breaking down the process into key steps. Start by preparing the data, then explore and evaluate different forecasting models, and conclude with an evaluation on test data. Outline each step and include key considerations."

Expected AI Response:

1. **Data Preparation:**
 - Convert dates to datetime format and set as index.
 - Check for missing values and fill them using interpolation.
 - Aggregate data by month to create a monthly sales series.
2. **Exploring Models:**
 - Begin with a simple ARIMA model as a baseline.
 - Test a Seasonal ARIMA (SARIMA) model to capture seasonality.
 - Experiment with machine learning models like Random Forest or XGBoost for comparison.
3. **Model Tuning:**
 - Optimize parameters for each model using grid search or cross-validation.
4. **Model Evaluation:**
 - Evaluate using metrics such as Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE).
 - Plot actual vs. forecasted values on test data to visually assess accuracy.

Benefit: This prompt guides the AI to systematically approach each component of the retail forecasting task, ensuring that all critical steps are covered logically.

2. Tree-of-Thought Prompting

Purpose: Explore multiple approaches to solving the forecasting problem before selecting a final model.

Prompt:

"Consider three different approaches for forecasting retail sales: a traditional time series model, a machine learning model, and a hybrid model. For each approach, provide a basic outline of the steps, advantages, and limitations. Finally, recommend the most suitable model for a large, seasonal retail dataset."

Expected AI Response:

1. **Traditional Time Series Model (ARIMA/SARIMA):**
 - **Steps:** Preprocess data, fit the ARIMA/SARIMA model, tune parameters, evaluate on test data.
 - **Advantages:** Simple, interpretable, captures trends and seasonality.
 - **Limitations:** May struggle with non-linear patterns and complex seasonality.
2. **Machine Learning Model (XGBoost/Random Forest):**
 - **Steps:** Preprocess data, create time-lagged features, split data, fit and tune model, evaluate.
 - **Advantages:** Handles large datasets, captures complex patterns.
 - **Limitations:** Requires feature engineering, can be less interpretable.
3. **Hybrid Model (e.g., ARIMA + XGBoost):**
 - **Steps:** Use ARIMA for trend-seasonality components, fit residuals with XGBoost, combine predictions.
 - **Advantages:** Captures both linear and non-linear patterns, flexible.
 - **Limitations:** More complex, requires expertise to tune.

Suggested Approach:

"For large, seasonal retail data, the hybrid model may offer the best performance by capturing complex patterns with flexibility."

Benefit: This prompt enables the AI to explore multiple solutions, compare them, and choose the best approach based on the problem context.

3. Self-Refine Prompting

Purpose: Guide the AI to iteratively refine the forecasting model by critiquing its performance and making adjustments.

Prompt:

"Create a retail sales forecasting model using a Seasonal ARIMA (SARIMA) model, then critique the model based on its test performance. Identify any issues, such as underestimating seasonal peaks, and refine the model to improve accuracy. Finally, evaluate the refined model and report the improvements."

Expected AI Response:

1. **Initial Model (SARIMA):**
 - Build a SARIMA model to capture monthly seasonality in retail sales.
 - Evaluate the model on test data with metrics like MAPE.
2. **Critique:**
 - "The SARIMA model accurately captures seasonality but underestimates peaks during holiday seasons, leading to higher error in these periods."
3. **Refinement:**
 - Adjust seasonal parameters to account for sharper peaks, or introduce exogenous variables (e.g., holiday indicator) to improve peak forecasts.
 - Test the refined model on the same test data.
4. **Final Evaluation:**
 - Recalculate MAPE and compare to the initial model, noting improvements in peak forecasting accuracy.

Benefit: This approach allows the AI to assess and refine its forecast iteratively, leading to a more accurate and tailored model for the retail data.

4. Prompt Engineering in Forecasting

Applications in Retail Forecasting

- Proof of Concept Development
- Generating Scripts
 - Exploratory Data Analysis
 - Complex scripts
 - Debugging
- Signals from textual Data

Example 1: Proof of Concept Development:

Comprehensive Time Series Forecasting Model

Prompt:

"Develop a Python script to create a comprehensive proof-of-concept time series forecasting model using SARIMA. The script should:

- Begin with data preprocessing steps, including time-based indexing, handling missing values, and outlier detection.
- Incorporate seasonality adjustments, selecting optimal model parameters using grid search or AIC minimization.
- Split the data into training (80%) and testing (20%) sets, evaluate model performance on the test set using metrics like Mean Squared Error (MSE) and Mean Absolute Percentage Error (MAPE), and provide visualizations of both actual vs. forecasted values and residual diagnostics.
- Conclude with a summary report, assessing model assumptions (stationarity, autocorrelation) and limitations, and suggest potential next steps. Use pandas, statsmodels, and matplotlib to implement this robust approach."

Example 2: Script Generation

Sophisticated Data Transformation and Aggregation Script

Prompt:

"Generate Python code that performs an advanced data transformation for time series analysis on a dataset `sales_data` containing `date`, `product_category`, and `transaction_value`. The code should:

- Convert `date` to a time-based index, handle any missing or inconsistent timestamps, and adjust for timezone differences if necessary.
- Aggregate the data by month and by `product_category`, calculating key metrics like total monthly sales, average transaction value, and count of transactions.
- Add calculated features such as month-over-month growth rate, moving averages (3-month, 6-month), and seasonal indicators (e.g., holiday season).
- Output a clean, structured dataframe ready for time series analysis with columns: `month`, `product_category`, `total_sales`, `avg_transaction_value`, and `transaction_count`. Implement this using pandas, ensuring high flexibility for different time zones and potential categorical feature expansion."

Example 3: Exploratory Data Analysis

In-Depth Visualization of Trends, Seasonality, and Anomalies

Prompt:

"Design Python code that performs a detailed exploratory data analysis (EDA) on a time series dataset `sales_ts` with columns `date` and `sales`. The script should:

- Generate an interactive line plot to visualize sales trends over time, with custom tooltips for daily, weekly, and monthly sales aggregates.
- Decompose the time series into trend, seasonal, and residual components using Seasonal Decomposition of Time Series (STL) and plot each component separately.
- Detect and annotate anomalies using the Z-score method or seasonal-trend decomposition, and flag these in the plot for visual clarity.
- Summarize insights from the EDA in a report section, highlighting seasonal patterns, cyclic trends, and any anomalies. Use libraries such as `plotly` for interactive visualization, `statsmodels` for decomposition, and `scipy` for anomaly detection."

Example 4: Signals from Textual Data

Advanced Signal Extraction from Text for Time Series Modeling

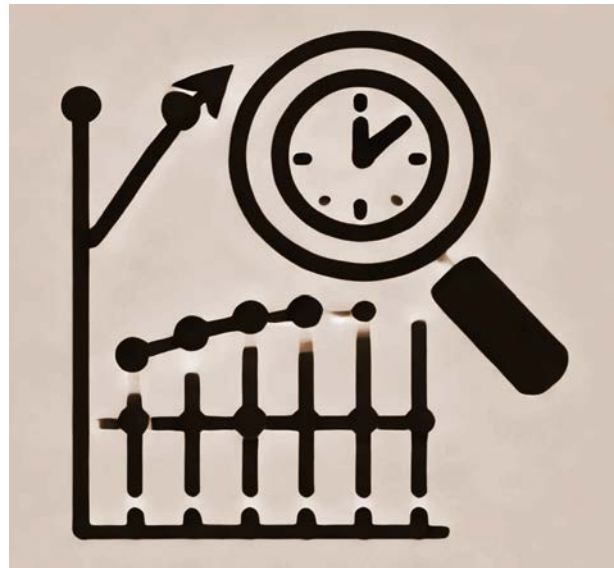
Prompt:

"Generate a Python script to extract meaningful signals from a dataset `news_headlines` containing date and headline, and integrate them into a time series `stock_prices` dataset with date and closing_price. The code should:

- Preprocess text data with advanced NLP methods, performing steps like tokenization, removal of stopwords, lemmatization, and extracting sentiment scores, using a fine-tuned model for finance or industry-specific sentiment.
- Create daily aggregate sentiment scores and keyword frequency metrics that reflect public sentiment and event frequency, and merge these with `stock_prices`.
- Use Granger causality tests to assess the predictive relationship between sentiment signals and stock price movements.
- Generate a visual correlation matrix and time-series plot overlaying sentiment score and stock price to illustrate alignment or divergence trends. Implement spaCy or Transformers for text processing, pandas for merging and transformation, and matplotlib or seaborn for visualization and correlation analysis."

Practical Implementation

- Dependence on Prompt Clarity:
- Limited Model Interpretability:
- Inadequate Handling of Unexpected Events:
- Reliance on High-Quality Data and Preprocessing:



5. Future of AI in Retail Forecasting

Future Trends in Retail Forecasting

- Incorporating **social media** and **external data** for demand sensing
- AI Agents for Supply Chain Planner for quick forecasting solutions
- Predictive analytics for **customer personalization** and **marketing optimization**
- **Ethical concerns:** balancing data collection with privacy regulations
- Fully **automated demand forecasting** systems driven by AI



Contact Me

Send your questions or feedback to:

- Email: sijo.vmanikandan@gmail.com



THANK YOU