Applied Deep Learning - Assignment 1

194.077 Applied Deep Learning 2025WS, TU Wien
Lena Schill, 12404231

`Graph Neural Network-Based Demand Forecasting`

1 Introduction

In this project, I aim to develop a demand forecasting model for products by leveraging Graph Neural Networks (GNNs) combined with temporal modeling. Traditional forecasting methods often treat products independently, ignoring inter-product dependencies such as co-purchases. In Austria, for example, customers frequently buy *Brötchen* with *Aufschnitt* or *Bratwürste* with *Senf.* Capturing these dependencies can improve demand predictions significantly. Recent research has demonstrated the potential of combining graph-based approaches with time series forecasting to capture both relational and temporal patterns in demand data. Kozodoi et al. 2024 showed that GNNs can effectively model product relationships through co-purchase patterns, leading to more accurate probabilistic demand forecasts compared to traditional methods. Their work highlights how graph structures can encode complex dependencies between products that are missed by conventional univariate or multivariate time series models. The survey by Jin et al. 2024 provides a comprehensive overview of graph neural networks for time series analysis, demonstrating their effectiveness across various domains including retail forecasting. They identify key advantages of GNN-based approaches, such as the ability to model dynamic relationships between entities, incorporate heterogeneous data sources, and capture both local and global patterns in the data. Particularly relevant to demand forecasting, they show how GNNs can propagate information across product networks, allowing the model to learn from the demand patterns of related products when making predictions for a target product. This is especially valuable for new products with limited historical data or for capturing seasonal patterns that affect groups of related products simultaneously.

2 Project Idea and Methodology

In this project, I will develop a neural network architecture that integrates a Graph Neural Network (GNN) with a temporal forecasting model, such as an LSTM, to predict future demand for products. The fundamental idea is to leverage both inter-product relationships and historical sales patterns simultaneously, as these two sources of information can improve forecast accuracy. Products will be modeled as nodes in a graph, where edges capture meaningful relationships between products. These relationships can be derived from co-purchase patterns, indicating products that are frequently bought together, or from semantic similarities based on metadata such as aisle or department, reflecting products that are naturally related within the store context. Incorporating these dependencies allows the model to account for cross-product influences in forecasting demand.

The GNN will operate on this product graph to generate embeddings, which are vector representations of each product that summarize both its own characteristics and the aggregated influence of its neighboring products in the graph. These embeddings effectively encode relational information, such as which products are likely to be co-purchased, and can be used to enhance the temporal forecasting process. The temporal component, implemented using an LSTM or a temporal GNN variant, will model sequential patterns in historical sales data, such as seasonality, trends, and Iekly purchase cycles. By feeding the GNN-generated embeddings into the temporal model, the architecture can produce demand predictions that are informed by both temporal trends and relational dependencies between products.

This project fits the “Bring Your Own Method” category because I will re-implement and adapt a state-of-the-art GNN-based forecasting architecture, rather than using a purely off-the-shelf solution. My approach will include experimenting with different strategies for graph construction, such as adjusting edge definitions or Iighting co-purchase frequency differently, exploring alternative GNN layers like GraphSAGE or GAT, and testing various methods to fuse graph embeddings with temporal inputs. The goal is not only to reproduce existing results from the literature but also to explore modifications that may improve forecasting accuracy. This process will provide hands-on experience in both graph representation learning and temporal forecasting, and allow me to evaluate how relational information can enhance demand prediction in a realistic retail setting.

3 Dataset

The project uses the publicly available **Instacart Online Grocery Shopping Dataset**, which contains detailed purchase records and product metadata from over 200,000 users and millions of individual orders. The dataset includes the following CSV files:

File	**Description**
orders.csv	Metadata for each order
order_products__prior.csv	Products in prior orders (historical purchase data)
order_products__train.csv	Products in training orders
products.csv	Product metadata
aisles.csv	Maps aisle IDs to aisle names
departments.csv	Maps department IDs to department names

Table 1: Overview of CSV files in the Instacart dataset

This dataset is well-suited for this project for several reasons. First, the large number of users and orders provides rich temporal information, allowing me to construct reliable demand time series for each product, which is perfect for a deep learning model. Second, the detailed product-level data enables the creation of a product graph, where nodes represent products and edges can be defined based on co-purchase frequency or shared categories. This structure is ideal for a Graph Neural Network, which can exploit relational dependencies between products to improve forecasting accuracy. Third, the aisle and department metadata provide additional semantic information, allowing the model to infer connections between products that are often purchased together even if they do not frequently co-occur in the same basket. Finally, the combination of historical purchase sequences, product attributes, and structured relationships offers a realistic, large-scale dataset that reflects true consumer behavior, making it an excellent testbed for developing and evaluating advanced GNN-based demand forecasting models.

4 Project Plan

The project begins in October 2025 with data setup and exploration, as illustrated in Figure 1. I’ll start by loading and exploring the historical demand data, which should take me through the end of October. Once I understand the data structure, I’ll move into preprocessing and cleaning tasks in early November, folloId by feature engineering work that will wrap up by mid-November. During this same period, I'll construct the graph representation of the product network, building the connections betIen products based on co-purchase patterns and category relationships.

Development work kicks off in late November and runs through early December. I'll implement the GNN architecture first, then build out the temporal model components that will handle the time-series aspects of demand forecasting. These two pieces will be fused together to create our complete forecasting system. Throughout early December, I'll train and test the model to ensure it's working as expected.

Once I have a working model, I'll spend most of December optimizing its performance. This includes fine-tuning hyperparameters and experimenting with different graph structures and model variations to see if I can improve fortheecasting accuracy. I'll also create visualizations to better understand how the model learns product relationships and generates its predictions.

The final phase involves making the model accessible to users. While I'll start some frontend development work in late December, the main push for both the frontend dashboard and API development happens in January 2025. The dashboard will let users view demand forecasts both at an aggregate level and for individual products. The API will handle serving these predictions to the frontend. I expect to have everything fully integrated and deployment-ready by early 2025 and then write the report and prepare the presentation.

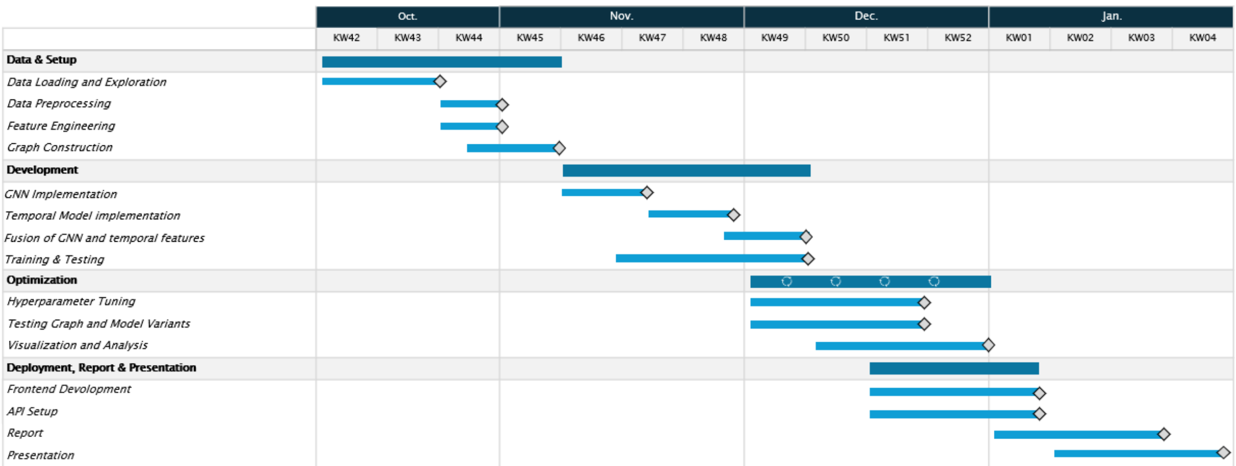

Figure 1: Project timeline showing the major phases and milestones from October 2025 to January 2026

References

Jin, Ming et al. (2024). "A Survey on Graph Neural Networks for Time Series: Forecasting, Classification, Imputation, and Anomaly Detection". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Kozodoi, Nikita et al. (2024). "Probabilistic Demand Forecasting with Graph Neural Networks". In: *arXiv preprint arXiv:2401.13096*.