PREDICTING RESALE PRICES IN SECONDHAND FASHION USING MACHINE LEARNING AND GRAPH NEURAL NETWORKS

Piangpim CHANCHARUNEE, Chiao-Kai CHIANG, Ruxi HE, I-Hsun LU
Machine Learning in Network Science - Final Project Report

ABSTRACT

The resale fashion market has grown rapidly in recent years, with platforms like Vestiaire Collective enabling users to buy and sell secondhand luxury and designer items. However, pricing in this market lacks transparency, and sellers often have limited guidance when setting prices. Accurately estimating resale value can help sellers price items more competitively, help buyers make informed decisions, and enable platforms to improve search, recommendations, and overall marketplace efficiency. In this project, we aim to predict the resale price of fashion items using structured product and seller information.

We treat this as a supervised regression problem and compare three modeling approaches: Random Forest, a traditional machine learning algorithm; Graph Convolutional Networks (GCNs), which are designed to learn from graph-structured data; and GraphSAGE, a more flexible GNN architecture that samples and aggregates information from neighboring nodes. To construct our graph, we connect products based on shared features such as brand, category, and material, and also add edges using similarity in product engagement.

While the GCN model struggled to perform well due to sparsity and loss of structure in the downsampled graph ($R^2 = 0.0546$), the GraphSAGE model demonstrated much stronger results ($R^2 = 0.33$), outperforming both GCN and Random Forest ($R^2 = 0.13$). This suggests that GraphSAGE is better suited to handle limited connectivity and sparsity in our dataset, making it a promising direction for future graph-based modeling of resale prices. Overall, our findings highlight the potential of relational learning for fashion resale prediction, while also pointing to the importance of choosing the right GNN architecture and maintaining strong graph structure during preprocessing.

1 INTRODUCTION & MOTIVATION

The rise of online resale platforms has significantly reshaped the second-hand fashion market, making it easier for customers to buy and sell pre-owned luxury and everyday fashion items. However, pricing in the resale market is often inconsistent, leading to listings that may be undervalued or overpriced. Understanding what drives the resale value of fashion products is beneficial not only for sellers and buyers but also for the platform itself. A more accurate prediction model could help sellers set competitive prices, resulting in better sales outcomes and a smoother shopping experience for buyers.

1.1 STAKEHOLDERS

For sellers, knowing the product features (such as brand, color, season, or material) that lead to higher resale value can help them price items more competitively and understand how to sell the product. For example, if material is a key feature in driving resale value, sellers can emphasize the material in the product description. It can also influence what items they choose to buy or resell in the future.

For buyers, insights into resale value of a product can induce them to make a purchase. This is especially prevalent in luxury items or designer fashion, where customers may be willing to pay

more for the product initially if it means they are able to resell the item later. For example, a buyer might spend €2,000 on a Chanel handbag, knowing that it typically resells for €1,500 or more on the secondhand market. This perceived value retention can make the purchase feel less like a cost and more like an investment.

For the platform, resale value prediction improves overall marketplace efficiency. It can be used to optimize search rankings, personalize recommendations, and even set dynamic pricing or suggested prices. It also builds trust in the platform, as users are more likely to return if they believe they are getting their money's worth when buying an item from the platform.

By accurately predicting resale value of items, sustainability goals in the fashion industry can also be supported. By helping sellers better understand which items hold their value, the model can encourage the resale of higher-quality, longer-lasting products rather than fast fashion items that lose value quickly and contribute to waste. Pricing items appropriately also helps sellers sell their items faster, reducing the number of unused listings and increasing the circulation of existing goods. For buyers, access to resale value predictions can promote more mindful purchasing decisions, such as favoring items that retain value and can be resold later, rather than impulsively buying disposable fashion. At the platform level, resale value predictions can be used to highlight listings that are more sustainable, helping to shift demand toward durable, high-quality products. In the long run, this kind of intelligent pricing infrastructure can reduce overproduction, extend product lifecycles, and support a circular economy in fashion, where fewer items end up in landfills and more are reused or repurposed.

1.2 Project Introduction

Therefore, we have decided to focus our project on predicting the resale value of fashion items listed on Vestiaire Collective. Vestiaire Collective is a global online marketplace for buying and selling pre-owned luxury and designer fashion items. By using data from Vestiaire Collective to analyze product attributes, seller characteristics, and pricing details, we plan to identify the key drivers that influence an item's resale potential. We will assume that the resale value is accurately represented by the selling price of the product, as price is a direct and quantifiable indicator. The price reflects how sellers believe the item is worth in the secondhand market, and is usually informed by seller expectations, market trends, and perceived value, making it a strong proxy for what an item is actually worth in resale.

1.3 Assumptions

Certain assumptions also must be made when using the price as an approximation of resale value. First, we assume that sellers set the listing price to maximize either profit or likelihood of sale, meaning that the price reflects what they believe the item is worth in the current resale market. Second, we assume sellers have some awareness of the market and base their pricing on similar items, trends, or prior experience, not completely at random. Third, we assume that pricing patterns in the dataset are representative and not skewed by interventions from the platform e.g., automatic price recommendations or hidden fees not shown in listed price.

1.4 Potential Applications

We have identified several practical use cases for our project. First and foremost, the model can assist sellers in setting competitive, data-driven prices for their items. By understanding which features most influence resale value (such as brand, material, or condition) sellers can price their products more accurately, increasing the likelihood of a successful sale. This helps reduce pricing guesswork and minimizes the number of overpriced or undervalued listings on the platform. For sellers who frequently resell items, this model could even inform future purchasing behavior by highlighting which types of products tend to retain value over time.

In addition to helping sellers, the model can benefit buyers and the platform itself. Buyers can use predicted resale prices to make more informed decisions, especially when purchasing higher-end or designer items. Knowing that an item is likely to hold its value may encourage a buyer to proceed with a purchase, viewing it as more of an investment than a one-time expense. On the platform side, predicted resale value can improve the quality of search rankings, recommendations,

and pricing suggestions. It can also support the development of advanced e-commerce tools that combine machine learning and network science to forecast demand, optimize pricing strategies, and identify trends in secondhand fashion. These capabilities not only enhance user experience but also contribute to a more efficient and sustainable resale ecosystem.

2 Problem Definition

We aim to predict the resale price of second-hand fashion items using relational and attribute-based information. We assume that sellers price their items reasonably, such that resale price reflects perceived value influenced by features like brand, material, condition, and engagement metrics. We model the dataset as a graph to capture inter-item similarity, leveraging Graph Neural Networks (GNNs) for improved prediction. We frame this task as a regression problem over a graph-structured dataset, where each product is a node with associated features, and edges represent similarities between items. The goal is to predict the resale price of each item by learning from both individual product attributes and their relationships in the graph.

2.1 Notation

Let:

- $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph, where:
  - $\mathcal{V}$ is the set of nodes, each representing a fashion item.
  - $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges representing similarity between items.
- Each node $v_i \in \mathcal{V}$ has a feature vector $\mathbf{x}_i \in \mathbb{R}^d$, composed of:

  $\mathbf{x}_i = \text{encode}(\text{product\_type, broad\_type, gender, category, color, brand, condition, material, like\_count})$
- Each node v_i has a resale price target $y_i \in \mathbb{R}_+$.
- $\hat{y}_i = f(\mathbf{x}_i, \mathcal{G})$ is the predicted resale price via a graph-based model f.

2.2 Objective Function

We minimize the mean squared error between the predicted and actual prices:

$$\min_f \quad \mathcal{L}(f) = \frac{1}{|\mathcal{V}|} \sum_{v_i \in \mathcal{V}} (y_i - \hat{y}_i)^2$$

2.3 Graph Construction

We define edges $(v_i, v_j) \in \mathcal{E}$ based on:

- Material similarity (same material label)
- Category similarity (same product category)
- Engagement similarity (similar like count ranges)

2.4 Constraints and Assumptions

1. **Non-negativity:** $\forall i,\ y_i \geq 0$
2. **Missing feature values** are encoded with a special "unknown" token or learned embedding.
3. **Reasonable pricing assumption:** We assume y_i reflects resale value reliably.
4. **Smoothness constraint (optional):** Neighboring items should have similar prices:

$$\sum_{(v_i, v_j) \in \mathcal{E}} w_{ij} \cdot (\hat{y}_i - \hat{y}_j)^2 \text{ is minimized}$$

2.5 Problem Hardness

The task is a nonlinear regression problem on a high-dimensional graph with categorical and numerical features. While predicting exact resale value is inherently noisy and potentially non-convex, GNN-based learning provides a scalable and tractable approach via backpropagation and message passing.

2.6 Goal

Given a set of second-hand fashion items and a graph encoding inter-item relationships, learn a graph-based regression model to predict resale price. The model optimizes mean squared error while leveraging both local (item-level) and global (graph-structure) information.

3 Related Work

3.1 Dif4FF

A research topic related to our project is Dif4FF: Leveraging Multimodal Diffusion Models and Graph Neural Networks for Accurate New Fashion Product Performance Forecasting by Avogaro et al. (2024). In this paper, the topic of New Fashion Product Performance Forecasting (NFPPF), or predicting the performance of new fashion products, is addressed. Traditional sales forecasting models often struggle with new products because they rely on historical data, which is not available for items that are newly introduced. This challenge is similar to what we face in our project, as both models are trying to predict how well a product will perform in the market without prior data, but rather by using product features.

To address the challenge of predicting future performance for new fashion products, previous research has emphasized the importance of using available information such as product specifications (color, type, material), release timing, and public interest signals (Avogaro et al., 2024, p. 2). These features serve as useful inputs for advanced deep learning models to make accurate forecasts, even when historical data is unavailable. Similarly, in our project, we use structured product features (such as brand, color, material, and season) to predict how well a product will perform in the resale market.

Avogaro et al. propose a two-stage prediction pipeline called Dif4FF that combines a multimodal score-based diffusion model with a Graph Convolutional Network (GCN) to forecast the sales performance of new fashion products (p. 3). The GCN is specifically used to model relationships between predictions across time and prediction space (p. 8), one capturing temporal relationships between weekly predictions, and the other capturing similarities among diffusion outputs.

This approach is highly relevant to our project, which also employs a GCN to improve prediction accuracy, though our target is resale value instead of sales. Similar to their challenge, we face variability and uncertainty in how products perform on the platform, and a GCN allows us to leverage structural relationships across products (e.g., brand, category, or material) to generate better predictions. However, unlike Dif4FF, we do not use a diffusion model, and our target variable is resale price, which we assume can be used to estimate the resale value. This makes our project different in terms of both modeling structure and output target. At the same time, we are replicating the core idea that modeling relational structure among items can improve prediction accuracy in fashion forecasting tasks.

3.2 ASOS GraphReturns

Another paper relevant to our project is *Predicting Product Returns in E-Commerce with Graph Neural Networks* by Kawas et al. (2023). This paper explores the use of GNNs to predict whether an online purchase will be returned. Similar to our project, the authors modeled the e-commerce environment as a graph to capture relationships between products and customers. The goal of their project was to improve the performance of return prediction systems by leveraging the structure of interactions in a marketplace, which goes beyond treating each transaction independently. The authors concluded that using graph-based models improves performance over classical methods such

as XGBoost and multilayer perceptrons (p. 1). This reinforces the idea that structured relationships in e-commerce data can provide valuable signals for downstream prediction tasks.

The model they propose uses both customer- and product-level features. For customers, features include average return rate, ratios of different return reasons, and counts of historical purchases and returns. For products, the model uses features such as price, return rate, and most frequent return reasons (p. 5). In addition, the authors add virtual nodes to the graph to enrich its structure. These include nodes for shipping countries, brands, product types, and return reasons, with each node storing the average features of connected entities (p. 5). These virtual nodes help to group related items and reduce sparsity. The results of their work demonstrate how a graph structure can effectively connect different parts of a dataset and lead to better performance. In our project, we plan to apply a similar strategy by using product similarities (e.g., same brand, material, category) to define edges, which allows us to explore the potential of GNNs in predicting resale value rather than returns.

4 Methodology

4.1 Data Exploration and Cleaning

Our data was obtained from a Kaggle dataset, which can be found here: `https://www.kaggle.com/datasets/justinpakzad/vestiaire-fashion-dataset/data`. A sample of the first two rows of data can be found below:

Table 1: Sample of Selected Columns Used for Modeling

product_type_cleaned	broad_type	product_gender_target	product_category	product_color	brand_cleaned	product_condition	material_cleaned	product_like_count	price_usd
Jacket	Outerwear	Women	Women Clothing	Navy	Barbara Bui	Very good condition	Cotton	1	127.80
Skirt	Bottoms	Women	Women Clothing	Grey	Miu Miu	Never worn	Wool	34	272.92

The dataset contains product listings from Vestiaire Collective, containing approximately 900,000 rows. It includes 36 columns describing the product listing, for example: `product_name`, `product_keywords`, `product_gender_target`, `sold`, `brand_name`, `product_material`, `product_color`, `price_usd`, `seller_badge`, `has_cross_border_fees`, `seller_products_sold`.

First, we performed exploratory data analysis. To support meaningful graph construction and enable efficient model training, we first selected product-level features such as `product_type`, `product_gender_target`, `product_category`, `product_material`, `product_color`, `price_usd`, `brand_name`, and `product_condition`. Since some features like `product_type` and `brand_name` had extremely high cardinality, we applied text normalization and fuzzy matching to group similar values. For `product_type`, we mapped raw values to a cleaned version and further into high-level categories (e.g., top, outerwear) as `broad_type`. For `brand_name`, we identified the top 70 well-known high-fashion brands and grouped the rest as `others`. This process reduced sparsity while preserving semantic meaning for both edge creation and downstream machine learning.

4.2 Feature Selection

For feature selection, we used domain knowledge as secondhand fashion shoppers to choose attributes that are most likely to influence the resale price of a fashion item. Our goal was to include features that provide meaningful information about the value of a product while avoiding redundancy or overlap between features. For example, we excluded both `product_type` and `product_category` from being used together, since they often describe similar aspects of an item in different levels of detail.

We carefully cleaned and simplified certain columns to make them more useful for modeling. The final set of selected features includes: `product_type_cleaned`, `broad_type`, `product_gender_target`, `product_category`, `product_color`, `brand_cleaned`, `product_condition`, `material_cleaned`, and `product_like_count`. The target variable we aim to predict is `price_usd`.

4.3 Model Selection

To evaluate the effectiveness of different modeling approaches for resale price prediction, we experimented with both traditional machine learning models and graph-based deep learning models. This allows us to compare the performance of models that rely solely on item-level features with those that can incorporate relational structure among products. Specifically, we selected three models for this project: Random Forest (RF), Graph Convolutional Networks (GCN), and GraphSAGE (SAGE).

Random Forest is a classic machine learning algorithm that serves as a strong baseline for tabular data. It performs well with mixed data types, is robust to noise, and requires relatively little hyperparameter tuning. As a model that treats each product independently, it helps establish how much predictive power exists in the product features alone without any explicit modeling of inter-item relationships.

In contrast, Graph Convolutional Networks (GCNs) are designed to learn from graph-structured data by propagating information across connected nodes. In our case, nodes represent fashion items, and edges encode shared characteristics such as brand, material, or category. GCNs are particularly suited to scenarios where local neighborhood information may help improve prediction accuracy.

We also implemented GraphSAGE, a graph neural network architecture that samples and aggregates information from a node's neighborhood rather than processing the entire graph at once. This makes SAGE more scalable to larger graphs and better suited for datasets where full connectivity is difficult to maintain. By including SAGE, we aim to evaluate whether a more flexible and efficient message-passing strategy can overcome some of the challenges we faced with GCNs, especially under computational constraints.

By comparing RF, GCN, and SAGE, we are able to explore the trade-offs between modeling complexity, graph connectivity, and predictive performance. This helps us understand when and how network-based learning adds value in secondhand fashion marketplaces.

5 Evaluation

The EDA showed that product price is highly right-skewed, with most items priced below $500 and a long tail of rare expensive products. This suggests that there may be a need for log-transformation or robust regression techniques in modeling. Key features that influence price include `broad_type`, `product_condition`, `brand_cleaned`, and `material_cleaned`. For example, outerwear, luxury brands like Chanel and Hermès, and materials such as fur and cashmere are consistently associated with higher prices. Additionally, products in better condition (e.g., "Never worn with tag") tend to fetch more. These insights guide feature importance and suggest that node-level attributes in a graph model should focus on product type, brand, material, and condition to learn meaningful embeddings or define graph edges effectively.

5.1 Random Forest Training

We first applied preprocessing by one-hot encoding all categorical features using `OneHotEncoder`, while leaving numerical features unchanged. This preprocessing step was implemented using a `ColumnTransformer` and combined with the `RandomForestRegressor` in a unified `Pipeline`. The dataset was split into training (70%), validation (15%), and test (15%) sets using `train_test_split` with fixed random seeds to ensure reproducibility. We trained the model using 100 trees (`n_estimators=100`) and default hyperparameters. The pipeline ensured consistent transformation of features during training and evaluation. One-hot encoding also allowed the model to handle high-cardinality categorical variables effectively. The Random Forest model trained successfully on the full dataset and achieved an R^2 score of 0.13, indicating that it was able to extract some predictive signal from the structured data.

5.2 GNN Model Training

To experiment with relational learning, we implemented two Graph Neural Network (GNN) models using the PyTorch Geometric library: Graph Convolutional Networks (GCN) and GraphSAGE

(SAGE). Each node in the graph represents a single product listing, and its feature vector includes both one-hot encoded categorical attributes (such as brand, material, product condition, and gender target) and standardized numerical values like `product_like_count`. The target variable is the original resale price in USD.

Edges were added between nodes based on shared attributes. For example, if two items had the same brand, material, or broad product category. We also introduced additional edges using k-nearest neighbors based on `product_like_count`, connecting listings that had similar engagement levels. These combined edges aimed to reflect both content-based and behavioral similarities between items. To reduce computational cost, we sampled 5% of the data across each brand before building the graph.

The models used a two-layer architecture: the GCN and SAGE variants both included two convolutional layers followed by a fully connected regressor that outputs a single predicted price. We trained both models using the Adam optimizer with a learning rate of 0.01 and a weight decay of 5×10^{-4} for 100 epochs. Performance was measured using Mean Squared Error (MSE), Mean Absolute Error (MAE), and the coefficient of determination (R^2) on an 80/20 node-level split.

After training, the GCN model reached an MSE of 644,539.81, MAE of 355.77, and R^2 of 0.0546, indicating very limited predictive power. In contrast, the GraphSAGE model performed much better, achieving an MSE of 455,572.88, MAE of 283.84, and R^2 of 0.3318.

We believe SAGE performed best because it is designed to better handle neighborhood aggregation in graphs with limited connectivity. Unlike GCN, which aggregates all neighbors equally, SAGE learns how to combine neighborhood information in a more flexible and robust way. This makes it more effective in real-world graphs where connections may be uneven or noisy, especially after heavy downsampling. Overall, the results suggest that while traditional models are dependable for standalone feature learning, graph-based models like GraphSAGE offer significant advantages when meaningful structural information can be preserved.

6 Conclusion and Future Work

In this project, we explored the task of predicting the resale price of secondhand fashion items using structured product and seller data from Vestiaire Collective. We compared several modeling approaches, including Random Forest, Graph Convolutional Networks (GCNs), and GraphSAGE, to evaluate their effectiveness. While GCNs theoretically offer a powerful way to model relationships between products, our initial experiments showed poor performance, likely due to the loss of graph connectivity after downsampling. Random Forest, a traditional machine learning model, served as a strong baseline, achieving an R^2 of 0.13 by leveraging item-level features. However, the most promising results came from GraphSAGE, which achieved an R^2 of 0.33. This improvement suggests that GraphSAGE is better suited for learning from sparse graphs by effectively aggregating information from local neighborhoods. These results demonstrate that when the graph structure is preserved well enough, network-based models like GraphSAGE can capture deeper patterns and outperform classical methods in predicting resale value.

6.1 Findings

One of the key findings of our work is that while traditional models like Random Forest remain stable and reliable for structured tabular data, graph-based models can outperform them when graph structure is properly preserved. Our experiments showed that Graph Convolutional Networks (GCNs) were sensitive to graph sparsity and performed poorly after downsampling. However, GraphSAGE proved more robust in this setting, achieving significantly better performance by effectively aggregating information from neighboring nodes. This result highlights that, although Graph Neural Networks require careful graph construction, they can be very useful when utilized appropriately. It also demonstrates a trade-off between model complexity and data constraints, where the right graph design and model choice can unlock the potential of relational learning in real-world applications.

Additionally, we also performed feature importance for the Random Forest model. Understanding which features influence resale price can be valuable for both sellers and buyers. For sellers, knowing that attributes like brand, material, and condition play a major role allows them to set more

competitive and realistic prices when listing an item. Buyers, on the other hand, can use this information to judge whether an item is likely to retain value over time. For instance, investing in luxury brands such as Chanel or Hermès, or products made from materials like silk or real leather, may offer higher resale potential in the future.

Our model includes `product_like_count` as a predictive feature. During our feature importance evaluation `product_like_count` was found to be the most important feature with an importance of 0.12. However, we must note that likes are a result of the listing rather than an input available beforehand. As such, they cannot be used proactively when setting the initial price and are not helpful in real-world pricing decisions from a seller or buyer perspective. Because `product_like_count` is only available after a product has been listed, the model in its current form is primarily useful for the platform itself rather than for individual sellers or buyers. For example, the platform could use the model to dynamically assess listing quality, adjust search rankings, or recommend price adjustments after some user engagement has occurred. However, if the goal were to support sellers in setting an initial price, or to help buyers evaluate an item before it's posted, `product_like_count` would need to be removed from the model entirely. In that case, only features available at listing time (such as brand, material, condition, and category) could be used to ensure the model aligns with the needs of real users making pricing decisions.

6.2 Limitations

Although our models provide a useful baseline for resale price prediction, there are several limitations. First, the target variable `price_usd` reflects the listed price, not the final transaction amount, which may differ due to discounts, negotiations, or items remaining unsold. Second, our graph structure was relatively simple and based only on shared features, without dynamic or behavioral signals such as co-viewed or co-purchased products. Third, we excluded image and text data, which could contain insightful semantic cues about product quality and style. Finally, the GCN model's poor performance may reflect both structural weaknesses and insufficient graph connectivity in the sampled data.

6.3 Future Work

For future work, we plan to explore strategies that allow for more scalable and effective use of graph neural networks. One direction is to apply graph sampling techniques that preserve structural properties, enabling training on larger portions of the data without losing essential connectivity. We are also interested in experimenting with simpler and more lightweight models such as LightGCN, which may be better suited for sparse graphs. Finally, incorporating multimodal data, such as text from product descriptions or image embeddings, could help enrich the feature set and improve predictive performance.

REFERENCES

[1] Andrea Avogaro, Luigi Capogrosso, Franco Fummi, and Marco Cristani.
Dif4FF: Leveraging Multimodal Diffusion Models and Graph Neural Networks for Accurate New Fashion Product Performance Forecasting.
FashionAI Workshop at ECCV 2024. arXiv:2412.06840 [cs.LG], 2024.
https://doi.org/10.48550/arXiv.2412.06840

[2] Jamie McGowan, Elizabeth Guest, Ziyang Yan, Cong Zheng, Neha Patel, Mason Cusack, Charlie Donaldson, Sofie de Cnudde, Gabriel Facini, and Fabon Dzogang.
A Dataset for Learning Graph Representations to Predict Customer Returns in Fashion Retail.
Lecture Notes in Electrical Engineering, vol 981. Springer, Cham. 2023. arXiv:2302.14096 [cs.LG], 2023.
https://doi.org/10.1007/978-3-031-22192-7_6