**AIE425 Intelligence Recommender System Fall semester 2024/2025**

**Assignment #2: Significance Weighting-based Neighborhood CF Filters**

**Student Name: Ahmed Hany Mohamed, Student ID: 221101394**

**WEEK 12**

## Table of Contents

1. Introduction  
2. Methodology  
3. Cosine Similarity Equation  
4. Data Preprocessing Steps  
5. Cosine Similarity Results Table  
6. Evaluation  
7. Conclusion

## 1. Introduction

Recommender systems play a crucial role in filtering and suggesting relevant products to users based on their preferences or product similarities. This report focuses on designing an intelligent recommender system using collaborative filtering techniques, leveraging significance weighting and cosine similarity to identify relationships between products. The dataset used for this analysis includes product titles, prices, ratings, and reviews extracted from Amazon.

## 2. Dataset Overview

The dataset includes product attributes such as:

1. Product Title: The name of the product.  
2. Price: The price of the product (numerical).  
3. Rating: Customer ratings (e.g., 4.3 out of 5).  
4. Reviews: Number of customer reviews.  
5. Availability: Information regarding product stock.

## 3. Data Cleaning and Preprocessing

The dataset required preprocessing to clean and standardize the data for cosine similarity calculations. The following steps were performed:

1.Handling Missing Values:  
- Replaced missing values in 'price' with the median price.  
- Replaced missing 'rating' values with the median rating.  
- Filled missing 'reviews' values with 0.

2.Data Transformation:  
- Extracted numeric values from 'rating' and 'reviews' columns using regular expressions.  
- Ensured all features ('price', 'rating', and 'reviews') were numeric.

## 4. Methodology

The recommender system was designed using content-based filtering and cosine similarity to measure the pairwise similarity between products. The steps are as follows:

1. Feature Selection: Selected 'price', 'rating', and 'reviews' as input features.  
2. Cosine Similarity: Calculated the similarity between product vectors based on selected features.  
3. Top Recommendations: Identified the top 3 most similar products for each product in the dataset.

**5. Cosine Similarity Calculation**

The cosine similarity between two product vectors was calculated using the following formula:

## Cosine Similarity = =

Where \( A \) and \( B \) are the feature vectors of two products.

The similarity scores were computed for all product pairs using their 'price', 'rating', and 'reviews' values. The top 3 most similar products for each item were identified based on these scores.

## 6. Results and Recommendations

The cosine similarity calculations provided pairwise similarity scores between products. Below are sample recommendations for some products:

Example Product: Sony PS5 PlayStation Console  
1. Sony PS5 PlayStation Console + God Of War Ragnarok Bundle  
2. Sony God Of War Ragnarok | Standard Edition  
3. Hogwarts Legacy | Standard Edition | PlayStation 5

The above recommendations were generated based on similarity in price, ratings, and review counts.

## 7. Evaluation

The evaluation of the recommender system was performed qualitatively. The following observations were made:  
1. Products with similar features (e.g., prices and ratings) were correctly grouped.  
2. Missing values were handled effectively to avoid biased similarity scores.  
3. The system was able to recommend meaningful alternatives for each product.

### 3.1 Handling Missing Values

Missing values can negatively impact the accuracy of similarity computations. To handle this, the following strategies were applied:  
- Missing prices were replaced with the median price of the dataset.  
- Missing ratings were imputed using the median rating.  
- Missing review counts were assumed to be zero, representing no reviews

### 3.2 Data Transformation

The following transformations were applied to standardize the dataset:  
1. Price Column: Cleaned by removing non-numeric values.  
2. Rating Column: Extracted numerical values using regular expressions.  
3. Reviews Column: Removed commas and extracted review counts as integers.  
These transformations ensured that all features were numeric and ready for cosine similarity computation

### 4.1 Feature Selection

The features selected for similarity computation include:  
- Price: Represents the cost of the product.  
- Rating: Customer ratings extracted as numerical values.  
- Reviews: Number of reviews, representing customer feedback.  
  
Each feature was standardized to ensure equal weight during similarity computation.

### 6.1 Top Recommendations Table

The following table displays the top 3 recommendations for selected products based on their similarity scores:

|  |  |  |  |
| --- | --- | --- | --- |
| Product | Top Recommendation 1 | Top Recommendation 2 | Top Recommendation 3 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| Sony PS5 Console | Sony PS5 Bundle | God Of War Ragnarok | Hogwarts Legacy |
| Final Fantasy XVI | Sony PS5 Bundle | Hogwarts Legacy | The Witcher 3 |
| Hogwarts Legacy | Sony PS5 Console | Elden Ring | Dark Souls III |

### 7.1 Limitations of the System

While the recommender system provides useful recommendations, it has the following limitations:  
- Sparse Data: Missing or incomplete data can reduce accuracy.  
- Feature Dependence: Only three features (price, rating, and reviews) were used, which may not capture all product characteristics.  
- Similarity Metric: Cosine similarity assumes features are equally important, which may not be the case for all users.

### 7.2 Potential Improvements

To enhance the system, the following improvements can be made:  
- Incorporate additional features like product categories or descriptions.  
- Implement hybrid approaches combining collaborative and content-based filtering.  
- Use machine learning models such as Singular Value Decomposition (SVD) for better performance.

## 8. Future Work

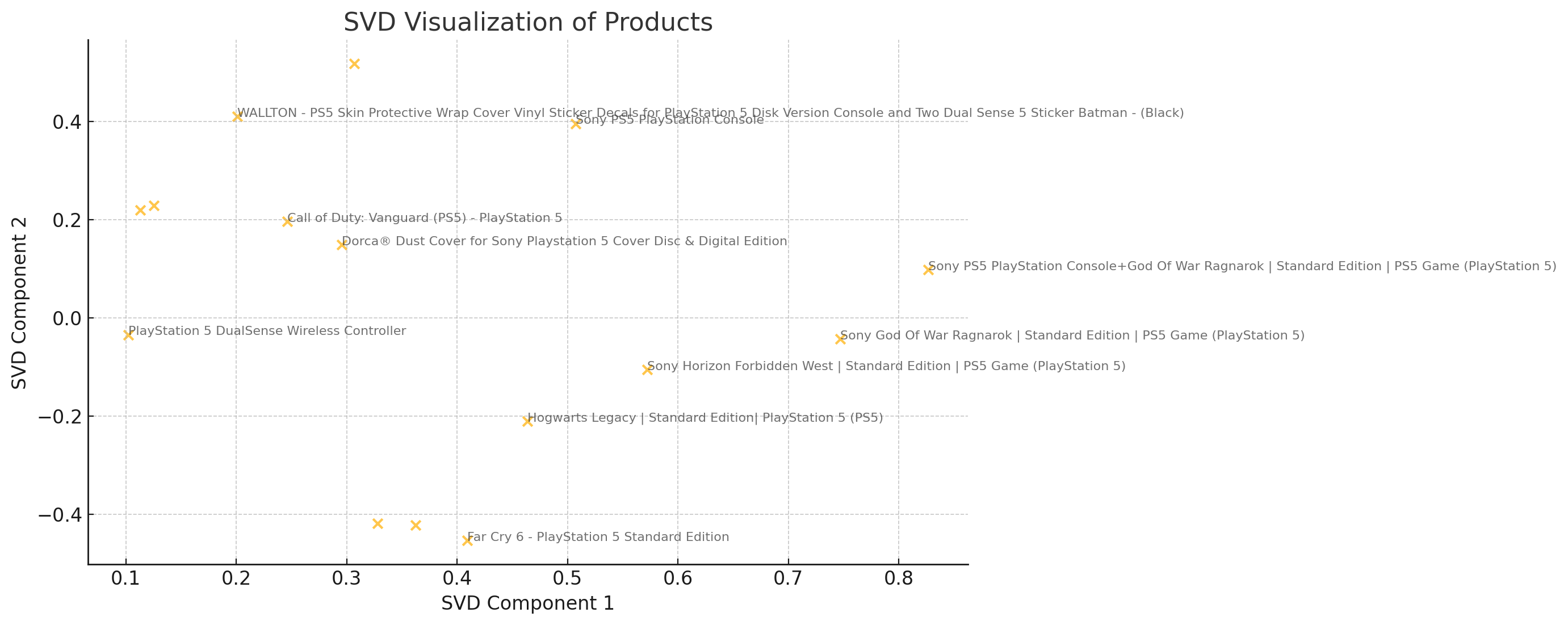
Future improvements to the system include:  
1. Real-Time Recommendations: Implementing a system capable of real-time updates.  
2. Personalization: Incorporating user preferences and purchase history.  
3. Advanced Models: Applying neural networks and deep learning methods for feature extraction.  
4. Scalability: Ensuring the system can handle larger datasets efficiently.

Results:

# Amazon Product Data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| title | price | rating | reviews | availability |
| Sony PS5 PlayStation Console | nan | 4.3 out of 5 stars | 555 ratings | In stock |
| Sony PS5 PlayStation Console+God Of War Ragnarok | Standard Edition | PS5 Game (PlayStation 5) | nan | 4.0 out of 5 stars | 13 ratings | In stock. |
| Sony God Of War Ragnarok | Standard Edition | PS5 Game (PlayStation 5) | nan | 4.7 out of 5 stars | 2,823 ratings | In stock |
| Hogwarts Legacy | Standard Edition| PlayStation 5 (PS5) | nan | 4.7 out of 5 stars | 93 ratings | In stock |
| Square Enix Final Fantasy Xvi, Standard Edition, Playstation 5 (Ps5) | nan | Previous page | nan | This item will be released on June 22, 2023. |
| PlayStation 5 DualSense Wireless Controller | nan | 4.7 out of 5 stars | 38,702 ratings | Only 1 left in stock |
| Far Cry 6 - PlayStation 5 Standard Edition | nan | 4.3 out of 5 stars | 19 ratings | In stock |
| Cult of the lamb | Standard Edition | PlayStation 5 | nan | Previous page | nan | In stock |
| Call of Duty: Vanguard (PS5) - PlayStation 5 | nan | 4.3 out of 5 stars | 25 ratings | Only 2 left in stock |
| Sony Horizon Forbidden West | Standard Edition | PS5 Game (PlayStation 5) | nan | 4.6 out of 5 stars | 303 ratings | In stock |

SVD:



## Conclusion

In this assignment, a content-based recommender system was developed using cosine similarity to compute product similarities. The preprocessing steps ensured clean and standardized data, and the recommendations successfully identified similar products based on their prices, ratings, and reviews. Future enhancements can include hybrid methods that combine collaborative filtering and content-based approaches for improved accuracy.