Product Matching using Machine Learning for E-Commerce Products: A Comprehensive Study

1st Paritosh Tripathi
Machine Learning Engineer
Greater Noida, India
tripathiparitosh935@gmail.com

2nd Manas Tripathi
Software Engineer
Greater Noida, India
manastripathi1027@gmail.com

***Abstract*—Automatically determining pairs of products that are same or similar is known as product matching. Manual matching can be time-consuming and error-prone due to the enormous number of products available on e-commerce platforms. Because of this, using machine learning approaches for product matching has attracted a lot of attention recently.**

In this method, numerous features are extracted and used to train a machine learning model, including product titles, descriptions, photos, and pricing. Even when there are variances in wording, phrasing, or other qualities, the model can accurately match products by learning to recognise similarities between them.

The e-commerce sector will be significantly impacted by the application of machine learning for product matching because it will allow for more precise recommendations, cut down on redundant listings, and enhance customer experience overall. Nevertheless, difficulties still exist, like maintaining data accuracy and addressing product differences and inconsistencies.

Overall, machine learning-based product matching has the potential to dramatically raise the efficacy and efficiency of e-commerce platforms. This field of study and development is now active.

***Index Terms*—product matching, machine learning, supervised learning, unsupervised learning, personalized recommendation, cross-selling.**

I. Introduction

A. Overview of Product Matching in E-Commerce

With the increasing number of products being sold on e-commerce websites, the problem of product matching has become a significant challenge for online retailers. Product matching refers to the process of identifying similar products across different online stores or marketplaces. The challenge lies in the fact that different sellers may describe the same product using different titles, descriptions, and attributes. As a result, online shoppers may find it difficult to compare prices and features of different products, leading to confusion and frustration.

B. Importance of Product Matching using Machine Learning

To address this problem, many e-commerce companies are turning to machine learning techniques to automate the process of product matching. Machine learning algorithms can learn patterns in the product data and identify similarities between products, even when they have different descriptions and attributes. Product matching using machine learning can help online retailers to provide a better shopping experience for their customers, by reducing the time and effort required to find the right product, increasing the accuracy of search results, and reducing the risk of errors in product recommendations.

C. Objectives of the Study

This study's main goal is to assess how well machine learning methods work for matching products in an e-commerce dataset. We specifically want to assess how well various machine learning models match products and look into how different feature extraction methods affect the models' accuracy.

D. Research Questions

To achieve our objectives, we will address the following research questions:

What are the most effective machine learning models for product matching in an e-commerce dataset? What feature extraction techniques are most suitable for product matching using machine learning? What is the impact of different feature extraction techniques on the accuracy of the machine learning models for product matching?

E. Scope and Limitations

The scope of this study is limited to product matching in e-commerce datasets. We will focus on comparing the performance of different machine learning models and feature extraction techniques on a single dataset of product information. The limitations of this study include the availability and quality of the dataset, the computational resources required for training and testing the machine learning models, and the inherent limitations of machine learning algorithms.

F. Structure of the Paper

The remainder of this paper is organized as follows. In the next section, we will describe the methodology used for this study, including the research design, data collection and preprocessing, machine learning models used for product matching, feature extraction techniques, and evaluation criteria. In the following section, we will present the results of the experiments and analyze the performance of the machine learning models. Finally, we will conclude the paper with a summary of the findings, a discussion of the implications and limitations of the study, and suggestions for future research.

II. Literature Survey

In e-commerce systems, matching products is a crucial activity because it aids in locating related or comparable products. In the past, manual matching was employed, but as the number of products grew, it became laborious and prone to error. In order to better match products, researchers have looked into using machine learning approaches.

Products have been matched using supervised learning, unsupervised learning, and deep learning models, among others. In the supervised learning approach, each pair of products is classified as either matching or mismatching, and a model is trained using this labelled data. The unsupervised learning approach includes classifying related products based on their traits without being aware of their compatibility beforehand. Convolutional neural networks (CNN) and recurrent neural networks (RNN) are examples of deep learning approaches that have been used to extract complex information from product photos and descriptions.

Machine learning approaches can considerably increase the accuracy of product matching, according to numerous research. For instance, Li et al. achieved 92.5 percent accuracy while using a supervised learning algorithm for product matching on an e-commerce platform. A CNN-based model was employed in a different study by Kim et al. to match products based on these photos, and it was successful in achieving an accuracy of 88.6 percentage

Machine learning can enhance cross-selling and product matching accuracy in addition to increasing the accuracy of product matches. For instance, Liao et al. (2019) created personalised suggestions for consumers based on their purchasing history using a deep learning-based model. Machine learning-based product matching still faces difficulties, including assuring the accuracy of the data and dealing with product variations and inconsistencies. Numerous research have suggested approaches to address these issues, including enhancing the calibre of labelled data through active learning and using product embedding to accommodate product variances.

In conclusion, e-commerce is significantly impacted by the application of machine learning techniques in product matching. This can increase the precision of product matching, provide personalised recommendations, and lessen the number of duplicate listings. To address new issues in this area, however, more study is required.

III. Preliminary Knowledge

A. Explanation of Product Matching using Machine Learning Techniques

Product matching is the process of locating and assembling related goods from various sources. The capacity of machine learning techniques to handle vast and complicated datasets has led to an increase in their application for product matching. Product matching may be automated with machine learning, saving time and resources. We outline the benefits of using machine learning techniques for product matching.

B. Explanation of Different Evaluation Metrics Used in Previous Studies

In earlier studies, a variety of evaluation measures were employed to assess the effectiveness of machine learning-based product matching. These measurements include accuracy, F1-score, recall, precision, and others. We give a summary of the many evaluation measures that have been used in earlier research to assess the effectiveness of machine learning-based product matching.

C. Machine Learning Models and Datasets

Various machine learning models, including Recurrent Neural Networks (RNN), Random Forest, Naive Bayes and Convolutional Neural Networks (CNN), have been employed for product matching. The complexity, size, and needs of the assignment determine which machine learning model should be used. We give a succinct description of several machine learning models and how well they function.

We collect a large dataset of product information from multiple e-commerce websites, including Flipkart and Amazon. The dataset includes product attributes such as product title, brand, category, description, and price. We preprocess the data to handle missing values, remove irrelevant attributes, and standardize the format of the product information. We also perform data exploration and visualization to gain insights into the characteristics of the dataset.

The project structure includes several directories, such as data, notebooks, src, tests, config.yml, requirements.txt, and README.md. The data directory contains raw, processed, and interim data. The src directory contains code for data processing, feature engineering, modeling, and utility functions. The notebooks directory contains Jupyter notebooks for experimentation and analysis. The tests directory contains test scripts. The config.yml file contains configuration information, and the requirements.txt file lists the required dependencies. The README.md file contains information about the project, such as project description, installation instructions, usage, and contribution guidelines.

IV. Methodology

A. Research Design

This study's research design uses a quantitative methodology. To find similarities and matches between various items in an e-commerce dataset, we intend to apply machine learning approaches to product matching. To compare how well various machine learning models match products, we will utilise a comparative analysis approach.

B. Data Collection and Preprocessing

To collect product information, we scrape data from two popular e-commerce websites, Flipkart and Amazon. The data points we collect include the product's unique identifier (SKU), name, description, image path, category, timestamp, URL, and price.

We preprocess the data after it is collected to make sure it is reliable and consistent. We handle missing data, eliminate pointless attributes, and standardise the product information's format. For the success of our machine learning models, this phase is critical in ensuring that the data is precise and clean.

In order to learn more about the features of the dataset, we additionally explore and visualise the data. Through this method, we are able to comprehend the data's distribution and spot any patterns or outliers that might have an impact on our models. We can improve our feature selection processes and preprocessing strategies by investigating and visually representing the data.]

For the sample size, we currently scraped 20,000 products from Flipkart and Amazon websites. This sample size was chosen to ensure that the dataset is large enough to train machine learning models effectively while also being manageable in terms of computational resources and time constraints. However, it should be noted that the sample size may not be representative of the entire population of products on these e-commerce platforms.

1) Project structure: The project follows a specific directory structure to maintain organization and modularity:

- **data/** - contains the raw, processed, and interim data directories.
  - **Amazon/** - contains the data scraped from Amazon website.
    - **raw/** - contains the raw data downloaded directly from the Amazon website.
    - **processed/** - contains the cleaned and preprocessed data.
    - **models/** - contains the saved machine learning models.
    - **scripts/** - contains the Python scripts for scraping and database connection.
    - **logs/** - contains the logs generated during the data scraping stage.
  - **Flipkart/** - contains the data scraped from Flipkart website.
    - **raw/** - contains the raw data downloaded directly from the Flipkart website.
    - **processed/** - contains the cleaned and preprocessed data.
    - **scripts/** - contains the Python scripts for scraping and database connection.
    - **logs/** - contains the logs generated during the data scraping stage.
  - **database.db** - contains the SQLite database for storing the scraped data.
- **lib/** - contains library code for reuse across the project.
  - **genricHtmlib.py** - contains generic functions for HTTP requests and HTML parsing.
- **notebooks/** - contains Jupyter notebooks for data exploration, model training and evaluation.
- **src/** - contains the source code for the project.
  - **models/** - contains the code for training and evaluating the machine learning models.
  - **preprocessing/** - contains the code for preprocessing the raw data.
  - **utils/** - contains utility functions used throughout the project.
- **tests/** - contains unit tests for the source code.
- **requirements.txt** - contains the necessary Python packages required for the project.
- **README.md** - contains information about the project, its objectives, and instructions for running the code.
- **StuctureMaker.py** - contains the Python script to generate the directory structure in LaTeX format.
- **push.sh** - contains the bash script to push changes to both Amazon and Flipkart directories.
- **push-amazon.sh** - contains the bash script to push changes to the Amazon directory.
- **push-flipkart.sh** - contains the bash script to push changes to the Flipkart directory.

C. Machine Learning Models Used for Product Matching

We experiment with various machine learning models for product matching, including but not limited to:

- Random Forest
- Naive Bayes
- Deep Learning models such as Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN)

The algorithms discussed above are all capable of handling textual input and recognising patterns and similarities between products, making them all appropriate for the task of product matching.

Both the well-liked machine learning algorithms Random Forest and Naive Bayes are adaptable to many data types and reasonably simple to use. These algorithms can handle high-dimensional data and can recognise crucial attributes for matching, making them appropriate for product matching. Additionally, they can deal with the noise and missing data that are prevalent in real-world datasets.

Text categorization and sentiment analysis are two examples of natural language processing (NLP) applications where deep learning models like CNN and RNN have demonstrated considerable promise. These models are able to recognise links between words and phrases and can learn intricate features from text input. Due to their ability to manage unstructured data and spot patterns that are not immediately apparent to humans, they are especially well suited for product matching.

After applying these methods, we can get a model that can accurately match products based on their name, description, and other relevant attributes. By comparing the performance of different models and optimizing their hyperparameters, we can identify the most effective algorithm for our specific problem statement. This will allow us to build a robust and accurate product matching system that can be used for a wide range of applications, including e-commerce and inventory management.

These models are chosen based on their suitability for the task of product matching and their performance in previous studies. We will also explore different configurations and hyperparameter.

D. *Feature Extraction Techniques*

We employ various feature extraction techniques to represent the product information in a meaningful way for machine learning algorithms. These techniques may include:

- Bag of Words (BoW)
- Word Embeddings (such as Word2Vec or GloVe)
- Other relevant feature extraction techniques such as TF-IDF, character-level embeddings, and product metadata extraction

These techniques are used to convert the raw product information into numerical representations that can be fed into machine learning models.

E. *Evaluation Criteria*

To evaluate the performance of the machine learning models for product matching, we use various evaluation criteria, such as:

- Precision, Recall, and F1-score
- Accuracy
- Other relevant evaluation metrics

These criteria are used to measure the effectiveness and efficiency of the product matching models and compare their performance against each other. We will also perform cross-validation and statistical tests to ensure the robustness of the results.

F. *Ethical Considerations*

In this study, we will adhere to ethical guidelines for conducting research. We will ensure that the data used for the study is obtained legally and with proper permissions. Any personal or sensitive information in the dataset will be anonymized and protected to maintain data privacy. We will also consider potential biases in the data, model outputs, and interpretations of the results. The study will be conducted with integrity, transparency, and accountability to ensure the validity and reliability of the findings.

V. IV. Results and Discussion

A. *A. Performance of different machine learning models for product matching*

We evaluated the performance of several machine learning models for product matching. Among these models, the random forest algorithm achieved the highest accuracy of 77. The results are summarized in Table.

TABLE I
Random Forest Model Results

Metric	Value
Accuracy	0.77
Precision	0.75
Recall	0.78
F1 Score	0.76

B. *B. Comparison of supervised and unsupervised learning models for feature extraction*

We also compared the performance of supervised and unsupervised learning models for feature extraction. Our results showed that the supervised approach outperformed the unsupervised approach in terms of accuracy.

C. *C. Personal observations and findings*

During the course of our study, we observed that product matching is a challenging task, especially when dealing with large and complex datasets. We also found that the performance of machine learning models heavily depends on the quality of the data and the choice of features.

D. *D. Potential applications of product matching using machine learning*

Product matching using machine learning has numerous potential applications, including e-commerce, inventory management, and supply chain optimization. By accurately matching products across different platforms and vendors, businesses can streamline their operations and improve efficiency.

VI. V. Challenges and Future Trends

A. *A. Challenges in product matching using machine learning*

Despite the significant progress in machine learning, product matching is still a challenging problem. Some of the challenges include:

- **Data quality:** The quality of data is critical for accurate product matching. Poor data quality, such as incorrect

product information, can result in inaccurate product matching.
- **Feature selection:** Feature selection is crucial in machine learning, and selecting the right features can significantly impact the accuracy of product matching models.
- **Large-scale matching:** Matching a large number of products can be computationally intensive and can require significant computational resources.
- **Cross-domain matching:** Matching products from different domains or categories can be challenging due to the differences in product attributes and characteristics.

B. B. Future trends in product matching using machine learning

Despite the challenges, product matching using machine learning has enormous potential. Some of the future trends in this field include:

- **Deep learning-based methods:** Convolutional neural networks (CNNs) and recurrent neural networks (RNNs), both of which are based on deep learning, have demonstrated promising results in a number of applications, including computer vision and natural language processing. These techniques can be used for product matching jobs and may increase the precision of product matching models.
- **Transfer learning:** Transfer learning can enable the reuse of pre-trained models on similar tasks, resulting in faster and more accurate model development.
- **Integration of structured and unstructured data:** Integrating structured and unstructured data can provide more comprehensive and accurate representations of products, which can improve the accuracy of product matching models.

C. C. Ethical considerations

As with any machine learning application, product matching using machine learning raises ethical considerations, such as:

- **Data privacy:** Product matching requires access to product data, and it is essential to ensure that the data used for product matching is collected and used ethically and with proper consent.
- **Bias:** Machine learning models can amplify existing biases in the data, resulting in biased product matching. It is critical to ensure that the models used for product matching are fair and unbiased.
- **Transparency:** Machine learning models can be complex and difficult to understand. It is essential to ensure that product matching models are transparent, and their decision-making processes are explainable.

VII. VI. Conclusion

A. A. Summary of the study

Using information from e-commerce websites, we investigated the use of machine learning for product matching in this study. After testing a number of machine learning methods, we discovered that the random forest approach had a 77 percent accuracy rate for our matching assignment. Also, we explored the relative merits and shortcomings of supervised and unsupervised learning techniques for feature extraction.

B. B. Implications of the study

The results of this study have implications for the use of machine learning in e-commerce, particularly in product matching. By improving the accuracy of product matching, e-commerce websites can enhance the customer experience, increase customer satisfaction, and ultimately drive more sales.

C. C. Contributions to the field

This study contributes to the field of machine learning by demonstrating the effectiveness of random forest algorithm in product matching. It also highlights the importance of feature engineering in machine learning, and provides insights into the strengths and weaknesses of supervised and unsupervised learning models for this task.

D. D. Recommendations for future research

Future research in this area could explore the use of deep learning algorithms such as neural networks for product matching. Another avenue for future research could be to investigate the use of machine learning in other areas of e-commerce, such as personalized recommendations and fraud detection. Additionally, further research could be conducted on the ethical considerations surrounding the use of machine learning in e-commerce, particularly in regards to data privacy and fairness.

References

1) https://medium.com/gobeyond-ai/product-matching-via-machine-learning-abstract-8b5de637114b
2) https://arxiv.org/abs/1608.04670
3) https://medium.com/walmartglobaltech/product-matching-in-ecommerce-4f19b6aebaca
4) http://www.semantic-web-journal.net/system/files/swj1470.pdf
5) https://towardsdatascience.com/unravelling-product-matching-with-ai-1a6ef7bd8614