

Product Matching for Enhanced Competitive Analysis

Animesh Choubey
Sr. Data Scientist, Competitive Intelligence
Lowe's Services India Pvt Ltd
Bangalore, India
animesh.choubey@lowes.com

Nidhi Talwar
Data Scientist, Competitive Intelligence
Lowe's Services India Pvt Ltd
Bangalore, India
nidhi.s.talwar@lowes.com

Abstract—Effective product catalogue matching across competitive companies has emerged as a critical imperative in today's dynamic business landscape, further fueled by the ubiquitous "endless aisle" paradigm of online retail. This paper addresses the intensifying significance of this challenge by proposing practical solutions derived from real-world projects. The complexity of the task arises from diverse factors, including the multitude of products, disparate data sets, incomplete information, and varying data quality. Moreover, the perpetual introduction of new products complicates efforts to achieve precise categorization and alignment. With companies now grappling with the intricate task of maintaining assortment value parity across millions of items to meet market demands and customer preferences, robust product matching solutions are indispensable.

In response, this paper concentrates on two pivotal questions central to successful product matching. Firstly, it delves into the categorization of product relationships, involving the intricate process of determining whether two products are identical, similar, or distinct. This nuanced process is quantified through a strategic amalgamation of natural language processing (NLP) techniques and neural network-based classification models. Secondly, the paper explores the strategic alignment of products from distinct company catalogues, a prerequisite for comprehensive comparative analysis. This alignment leverages retrieval techniques like BM25+ to efficiently identify initial product pairs for further investigation.

Key methodologies encompass supervised learning for precise product similarity quantification, deep learning techniques for cross-company catalogue matching, and the application of retrieval strategies for meticulous pair selection. Through these methodologies, the proposed approach enhances the accuracy of product matching, providing businesses with valuable insights to refine their product positioning and strategies, particularly in relation to aspects like assortment optimization. The practical implementation of these techniques further emphasizes their potential and effectiveness. Ultimately, this paper serves as a valuable guide for researchers, practitioners, and industry experts keen on enhancing catalogue matching across competitive enterprises, grounded in tangible real-world experiences.

Keywords— *NLP Techniques, Retrieval Strategies, Neural Networks, Product Similarity, Real-World Implementation*

I. INTRODUCTION

As retailers navigate the digital landscape, the strategic importance of comprehending competitors' product positioning has surged. This transformation has empowered both sellers and buyers to conduct product comparisons with ease. However, it has also ushered in a formidable challenge: the alignment of products across diverse data sources, each representing a different retailer. Within this dynamic context, where hundreds of millions of items are listed daily, this challenge looms even larger. Termed the 'endless aisle,' the

online retail environment has intensified the need for precise product comparisons and matching across competing companies. This paper is dedicated to addressing this imperative challenge by presenting practical solutions drawn from real-world projects.

A. The Complexity of the Challenge

The complexity inherent in product catalogue matching stems from a multitude of factors [6]. These include the vast number of products available in the market, the presence of disparate and sometimes incomplete datasets, and the inconsistent quality of data. Moreover, the perpetual introduction of new products further compounds the challenge, making it increasingly difficult to achieve precise categorization and alignment.

Today, companies are confronted with the intricate task of maintaining assortment value parity across an extensive array of items to satisfy market demands and customer preferences. In this context, robust product matching solutions become indispensable.

B. The Focus of This Paper

The paper focuses on addressing these challenges by zeroing in on two pivotal questions that are central to successful product matching. Firstly, it delves into the intricate process of product relationship categorization [6], involving the determination of whether two products are identical, similar, or distinct. This nuanced process is quantified through a strategic fusion of natural language processing (NLP) techniques and neural network-based classification models.

Secondly, the paper explores the strategic alignment of products sourced from distinct company catalogues—an essential step for comprehensive comparative analysis [2]. This alignment becomes particularly crucial from an engineering standpoint as we scale, given the impracticality of comparing every item from one catalogue to every item from another catalogue (a cross-product approach). To address this challenge, our approach leverages retrieval techniques, notably BM25+, to efficiently identify initial product pairs that warrant further investigation.

C. Methodologies Employed

To attain these objectives, the paper employs a diverse set of methodologies. These encompass the application of retrieval strategies to carefully select pairs across different companies, the utilization of a BERT-based model in conjunction with other similarity techniques to quantify product similarities, and the incorporation of deep learning-based supervised learning techniques. These methods collectively enable the paper to address the intricate patterns

inherent in distinguishing between identical, similar, and different products.

By applying these methodologies, the proposed approach significantly enhances the accuracy of product matching, thereby providing businesses with valuable insights to refine their product positioning and strategies. The practical implementation of these techniques underscores their potential and effectiveness.

D. Paper structure overview

The paper unfolds systematically, beginning with an introduction that introduces the topic of attribute similarity learning. The subsequent sections explore prior research, present the methodology, explain data preparation and processing, and outline the experimental setup, following which, results are analyzed. In conclusion, findings are summarized, and their importance highlighted. The paper concludes with a list of references for further reading.

II. RELATED WORK

In the context of effective product catalogue matching across competitive companies, we need to establish a system for extracting product attributes, use of blocking for limiting the search space, and fitting a suitable machine learning model.

The choice of search space is of paramount significance. Numerous studies [1,2,3] have proposed methodologies relying on techniques such as K- Nearest Neighbours (KNN) or Locality-Sensitive Hashing (LSH) to address this challenge effectively. However, it is necessary to acknowledge that the applicability of these approaches may be limited in certain scenarios, particularly when dealing with data characterized by a high number of attributes. In our specific dataset, the utilization of KNN or LSH techniques demonstrates sub-optimal performance, necessitating the exploration of alternative strategies to enhance the accuracy and efficiency of product matching.

[4] introduces a system for extracting product attributes from short, structure-limited text in eCommerce product titles. It combines sequence labelling algorithms like Conditional Random Fields and Structured Perceptron with normalization techniques, demonstrating effectiveness, particularly for attributes like brand. The challenges that remain unaddressed by this approach are:

- Inability of product titles to encapsulate all the product attributes
- Lack of uniformity in product titles among similar items
- High similarity in product titles of similar items.

E.g.: The titles “Style Selections Chantilly 2-Piece Patio Conversation Set with Off-white Cushions” and “Style Selections Chantilly 3-Piece Patio Conversation Set with Off-white Cushions” are highly similar, but they are different products.

These unresolved issues underscore the need for novel methodologies and strategies.

III. PROPOSED METHODOLOGY

Our methodology for addressing the problem at hand encompasses three pivotal phases: feature learning, pair identification, and similarity learning. This section delves into the intricacies of these phases, emphasizing their academic rigor and technical depth.

A. Feature Learning

The initial phase focuses on standardizing features and their values across diverse data sources. This standardization expands the scope of comparisons beyond generic attributes like title, brand, and category, enabling the inclusion of product-specific attributes such as wattage for appliances or blade count for fans. The primary challenge here is the non-standard nomenclature of attributes[4].

To address this challenge, we propose a solution leveraging labelled pairs from multiple sources. Through the establishment of a learning algorithm, we create regex patterns and feature dictionaries. For instance, consider a pair (p1) consisting of data from source1 and source2, where feature values (V1, V2) are linked to attribute pairs (a1, a2). These attribute pairs (a1, a2) are subject to specific conditions:

If the attribute type is,

$\text{Numeric} \rightarrow \left\{ 1 - \frac{\text{abs}(V1-V2)}{\text{Max}(V1,V2)} \right\} \geq 0.99 \rightarrow A1=A2$
$\text{Text} \rightarrow \text{Similarity}(V1,V2) \geq 0.8 \rightarrow A1=A2$

Figure 1. Attribute pairing based on the type of the value

Since the comparison leverages established pairs, employing primitive similarity metrics such as Jaccard yield precise results.

By repeating this process across thousands of pairs, our methodology systematically formulates a set of rules defining each attribute. For instance, an attribute like 'Height' may manifest in over 50 variations, including 'product height,' 'H * W * L,' or 'Height.' Through iterative matching and labelling, we derive versatile regex patterns such 'height\|(. * x h. *)\|(h x . *)'. These patterns serve as the cornerstone for creating attribute mappings across datasets.

This meticulous standardization of attributes not only enhances accuracy but also facilitates seamless comparisons across diverse datasets, setting the stage for subsequent phases of our methodology.

B. Pair Identification

The task of pair identification shares similarities with the concept of 'blocking' in search engines or recommender systems, where the goal is to retrieve the most relevant documents or items for a given query. Traditionally, indexing algorithms like Approximate Nearest Neighbour (ANN) or Locality Sensitive Hashing (LSH) have proven effective in such tasks [3], typically requiring some fine-tuning to achieve satisfactory results. However, these methods encounter scalability issues when dealing with datasets that necessitate comparisons across multiple attributes simultaneously due to data availability and variance. The challenge is exacerbated by the sparsity of data across attributes. Furthermore, previous approaches within the context of this problem often rely on

domain expertise and extensive parameter tuning to attain optimal results, incurring significant costs.

In response to these challenges, we introduce a two-stage retrieval system, resembling a funnel. The first stage aims to identify the most relevant generic attributes across products, excluding the product titles. To accomplish this, we employ a combination of the most frequently occurring attributes (with a coverage threshold of $> x\%$) and the top 'n' most frequently used filters in Left-Hand Navigation (LHN), where 'x' and 'n' are parameters within the final equation. Subsequently, we determine the closest 'm' values for each attribute, creating the initial level of filtering. For the top 'm' values, we employ the BM25+ similarity metric, known for its scalability across attributes, allowing for tuning over a range of parameters.

The second stage focuses on identifying the top 'k' similar product titles from the filtered set, ultimately generating the final pairs for further examination. It's crucial to emphasize that the effectiveness of this step directly impacts the quality of matches discovered in subsequent stages. The parameters defined here, namely 'x,' 'n,' 'm,' and 'k,' must be finely tuned to achieve the desired coverage while efficiently managing computational resources. Our optimization aims for a coverage level of at least 90%.

It's worth noting that the 'title' of a product is generated based on a fixed set of rules specific to the data source, and similarity is applied to a subset of the dataset. Statistical techniques like BM25+ deliver satisfactory results, even with a moderate value of 'k' (approximately 500) when dealing with such cases. However, it's essential to acknowledge that this may not hold true for cross-industry datasets, where a more advanced semantic similarity approach may be necessary.

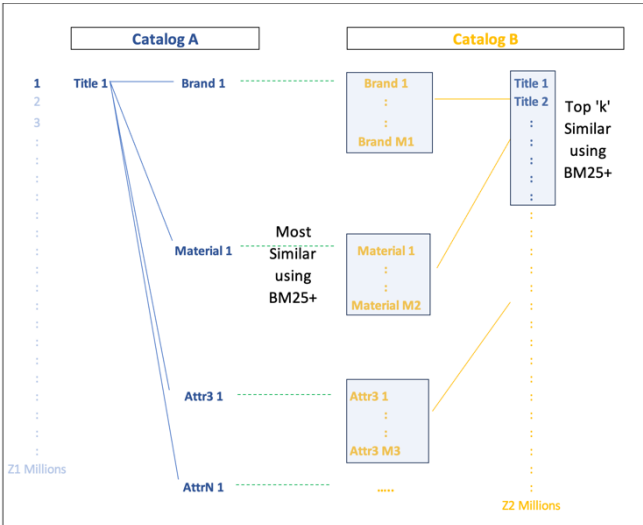


Figure 2. Process overview: Identification of pairs

C. Similarity Learning

In the realm of similarity learning, the task at hand becomes more straightforward once data is standardized, and initial pairs for comparison have been identified. When dealing with a set of attributes, we have devised a systematic approach to calculate similarity based on the data type of each attribute.

For numerical attributes, similarity is determined by assessing the difference in values concerning their absolute magnitudes. In contrast, text attributes are subjected to a two-fold analysis, considering both semantic and syntactic aspects.

The formulas governing these similarity measures are presented below.

Similarity	{	Numerical, $1 - \frac{\text{abs}(V1-V2)}{\text{Max}(V1,V2)}$
		Text, Semantic $1 - \text{cosine similarity}$
		Text, Syntactic $1 - \text{Lev similarity}$

Figure 3. Similarity Calculation

Semantic Embeddings: To capture the nuanced semantic meaning of attributes, we employ BERT-based sentence embeddings. This approach proves particularly valuable when semantic understanding is paramount.

Syntactic Similarity: For text attributes where employing embeddings may not yield substantial benefits and may introduce computational overhead, we opt for a more efficient strategy. In these cases, we calculate similarity using the Levenshtein distance, which provides valuable insights into textual resemblance.

Classification Model: Once we've computed similarity scores for attribute pairs, we deploy a neural network-based multi-classification model. This model has been trained on existing labelled data and classifies attribute pairs into three categories: identical, similar, and different. It's worth noting that the architecture of the model may require fine-tuning based on the specific characteristics of the input data. We chose a neural network for its ability to uncover intricate, non-linear patterns unique to the data's nuances.

IV. DATA PREPARATION AND PROCESSING

In this section, we outline the meticulous steps taken to prepare and process raw attribute data for our research on "Product Matching for Enhanced Competitive Analysis." The raw data contained a variety of formats, including text descriptions and numerical values for attributes like dimensions and pack sizes. To effectively leverage this data, we employed a blend of Natural Language Processing (NLP) techniques and regular expressions (regex) for extraction and standardization. Additionally, we discuss the creation of attribute embeddings to capture data richness.

A. Pre-processing attribute data

Text Extraction: We initiated the process by extracting pertinent information from textual attribute descriptions in the raw data. This involved identifying and extracting alphanumeric text, numeric values, dimensions, and brand names—essential elements for product matching and competitive analysis.

Numerical Data Extraction and Unit Standardization: For attributes such as product dimensions, pack sizes, and weight, we employed regex patterns to extract numerical values. To ensure uniformity, we standardized units, converting various units (e.g., centi-meters to inches, kilograms to pounds) to a common system for meaningful comparisons.

Attribute Embeddings: To represent pre-processed attributes suitable for machine learning and matching, we generated embeddings tailored to attribute types:

a) Numerical Attributes: For dimensions and pack sizes, numerical values served as embeddings, retaining quantitative information.

b) Colour Attributes: RGB (Red, Green, Blue) values were used to represent colour attributes, capturing their visual characteristics.

c) Text Attributes: Product descriptions underwent embedding using a BERT encoder to encapsulate semantic meaning for advanced textual data matching.

B. Train/Test data split

We divided standardized attribute data into two subsets: training – 80% and testing – 20%. The training set facilitated model development and fine-tuning, containing standardized attributes with known matches for supervised learning. The testing set assessed model accuracy by using attributes with unknown matches.

C. Handling new data

To adapt to new, unmatched attributes in practical applications, we employed feature extraction and matching techniques similar to those used in the training and testing phases. This allowed us to propose potential matches for previously unmatched product attributes, ensuring scalability and adaptability.

In summary, our data preparation and processing phase involved attribute extraction, standardization, and tailored embeddings. These embeddings formed the basis for our product matching models, enhancing competitive analysis in our research.

V. EXPERIMENTAL SETUP AND RESULTS

In this section, we present the outcomes of our experiments, with a particular emphasis on various elements of the training process. We begin by providing a concise overview of the datasets and baseline models. Subsequently, we delve into the architectural decisions made for our model.

A. Dataset and Architecture

In our experimental setup, we employed a multi-class neural network to identify product matches. Our training dataset consisted of approximately 500K well-balanced samples, encompassing identical, similar, and distinct product pairs. The base model was initially trained with a standard architecture, which was subsequently fine-tuned based on category-specific data. Below is the architecture of the initial model. The activation for all the layers here was ‘relu’ with cross entropy loss and Adam optimizer.

Model: "sequential"

Layer (type)	Output Shape	Param #
dense (Dense)	(None, 200)	2400
dense_1 (Dense)	(None, 100)	20100
dense_2 (Dense)	(None, 50)	5050
dense_3 (Dense)	(None, 40)	2040
dense_4 (Dense)	(None, 30)	1230
dense_5 (Dense)	(None, 2)	62
=====		
Total params: 30,882		
Trainable params: 30,882		
Non-trainable params: 0		

Figure 4. Model Architecture

To assess the model's performance, we implemented a two-step validation process. First, we evaluated the results against labelled data from the training and out-of-sample test datasets, monitoring precision and recall across the dataset. Precision here means the correctness of the recommendation made for a match. Recall accounts for the proportion of actual 'matches' present in the output.

Second, we scored unlabelled data and conducted manual validation to assess the model's stability and performance with slightly different data. This step was crucial to mitigate bias from the existing dataset. Importantly, feedback from both validation steps informed our ongoing model training and refinement.

B. Results

Our dataset is proprietary and subject to confidentiality agreements, restricting the disclosure of specific details and data points in this presentation or publication. We aim to offer a high-level overview of our findings while honoring data privacy and confidentiality. The table below depicts the model's performance on the test split (OOS), and the unlabeled data (Diverse set) underwent manual validation.

While the results are satisfactory on both datasets, the variation in precision demands more diversity in the training data, exposing the model to new patterns for a match. This problem should subside as more feedback flows into the system.

Table 1. Model performance on test data

Source	Precision	Recall
Out of sample	85%	93%
Diverse set	78%	94%

VI. CONCLUSION

This paper has addressed the complex challenge of product matching in the dynamic online retail landscape. With the exponential growth in product listings and diverse data sources, achieving precise product comparisons is essential. We've focused on two key aspects: product relationship categorization and strategic alignment of products from different catalogues.

Our approach combines natural language processing and neural networks to categorize products accurately. Additionally, we've introduced an innovative strategy for aligning products across catalogues efficiently, which becomes critical for noisy data and scalability in e-commerce. We've emphasized the real-world integration of our approach, particularly in tasks like recommendation and search relevance.

While addressing computational resource challenges, our work equips businesses with valuable tools for enhancing product positioning and strategies. This research contributes to the advancement of product matching and catalogue alignment, benefiting both retailers and consumers in the competitive e-commerce landscape.

In future studies, we plan to explore the integration of image comparison techniques to enhance the precision of product matching [7,8]. This avenue holds promise for extending our research and addressing the evolving needs of the industry.

REFERENCES

- [1] Jingdong Wang, Heng Tao Shen, Jingkuan Song, and Jianqiu Ji. "Hashing for Similarity Search: A Survey" arXiv preprint arXiv:1408.2927v1 (2014).
- [2] J Omid Jafari, Khandker Mushfiqul Islam, Parth Nagarkar. "Drawbacks and Proposed Solutions for Real-time Processing on Existing State-of-the-art Locality Sensitive Hashing Techniques" arXiv preprint arXiv:1912.07091v1(2019)
- [3] Muhammad Ebraheem, Saravanan Thirumuruganathan, Shafiq Joty, Mourad Ouzzani, and Nan Tang. "Distributed Representations of Tuples for Entity Resolution" Proceedings of the VLDB Endowment Volume 11, Issue 11, pp 1454–1467, 10.14778/3236187.3269461 (2018)
- [4] Ajinkya More. "Attribute Extraction from Product Titles in eCommerce" arXiv preprint arXiv:1608.04670v1(2016)
- [5] Foxcroft, Jeremy & Chen, Tianle & Padmanabhan, Kanchana & Keng, Brian & Antonie, Luiza. "Product Matching Lessons and Recommendations from a Real World Application" Proceedings of the Canadian Conference on Artificial Intelligence, 10.21428/594757db.08c5079e (2021)
- [6] Janusz Tracz, Piotr Iwo Wójcik, Kalina Jasinska-Kobus, Riccardo Belluzzo, Robert Mroczkowski, and Ireneusz Gawlik. "BERT-based similarity learning for product matching" In Proceedings of Workshop on Natural Language Processing in E-Commerce, pages 66–75, Barcelona, Spain. Association for Computational Linguistics(2020)
- [7] Z. Liu, P. Luo, X. Wang, and X. Tang, "DeepFashion: Powering Robust Clothes Recognition and Retrieval with Rich Annotations," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
- [8] Sangeet Jaiswal, Dhruv Patel, Sreekanth Vempati, Konduru Saiswaroop. "Product Review Image Ranking for Fashion E-commerce" rXiv preprint arXiv:2308.05390v1 (2023).