

Semantic-Retrieval-Reasoning Pipeline: A Hybrid Large Language Model Framework for Intent-Aware and Explainable Bundle Recommendation

Andy Maulana Yusuf

Telkom University

Adiwijaya Adiwijaya

adiwijaya@telkomuniversity.ac.id

Telkom University

Agung Toto Wibowo

Telkom University

Z. K. A. Baizal

Telkom University

Research Article

Keywords: Bundle Recommender System, Large Language Models, Multi-intent Detection, Human-in-the-Loop, Semantic-Retrieval-Reasoning

Posted Date: August 19th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-10565106/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Semantic-Retrieval-Reasoning Pipeline: A Hybrid Large Language Model Framework for Intent-Aware and Explainable Bundle Recommendation

Andy Maulana Yusuf^{1†}, Adiwijaya Adiwijaya^{1*},
Agung Toto Wibowo^{1†}, Z. K. A. Baizal^{1†}

^{1*}Computer Science, Telkom University, Jl. Telekomunikasi No.01,
Bandung, 40257, West Java, Indonesia.

*Corresponding author(s). E-mail(s): adiwijaya@telkomuniversity.ac.id;

Contributing authors: andymaulanayusuf@telkomuniversity.ac.id;
agungtoto@telkomuniversity.ac.id; baizal@telkomuniversity.ac.id;

[†]These authors contributed equally to this work.

Abstract

Increasing revenue for grocery sellers is a challenge today and utilize the Bundle Recommendation System for able to sell several products in one package based on consumer needs. Previous research using the user’s basket as a single intent results in monotonous recommendations. The current methods such as GNN or Contrastive based also does not help even though it succeeds in increasing relevance, but users are not informed why they get the recommendation, and the size of the bundle is very large and solved by trimmed to 2 to 3. Starting from this, we added an approach that utilizes LLM in a hybrid based, namely Prompting and Embedding, which are arranged within the Semantic-Retrieval-Reasoning Pipeline (SRRP) framework. Firstly, we are detecting the user’s basket to be multi-intent, then utilizing each intention to get Top-N baskets with similar intentions, arranged it by LLM into dynamic bundles, have selling names, and reasonable explanations, and have been validated by 9 annotators through the Human-in-the-Loop (HITL) method. The results on the Instacart dataset show that SRRP is able to produce diverse bundle recommendations (ILD >0.53) far above the threshold of 0.4 and relevant (Recall 0.1502, MRR 0.0683) for top-20, more importantly, SRRP successfully provides bundles with a size of 3 to 5 items for around 79% of users who frequently interact, and Long-tail distribution of up

to 45 items for users who buy massively. Moreover, the linguistically generated bundle naming and explanation (G-Eval 4.79 out of 5.0) are robust and in line with human (Cosine Similarity 0.59).

Keywords: Bundle Recommender System, Large Language Models, Multi-intent Detection, Human-in-the-Loop, Semantic-Retrieval-Reasoning

1 Introduction

Product bundling has highly become an effective marketing strategy, where designed to enhance user experience and maximize transaction revenue [1]. Subsequently, the number of items and user preferences have grown, automated Bundled Recommender Systems (BRS) have been extensively developed to simplify their management [2, 3]. However, for making a balance between recommendation relevance, diversity, and explainability remains a significant challenge.

BRS in its early stages relied on constraint-based feature engineering and data mining like Apriori or FP-Growth [4–6]. Study by Fang et al. [7] attempted to establish item correlations using constraints and a frequently occurring item set. While simple, this approach resulted in monotonous recommendations with very low diversity [8]. To address these limitations, recent literature has shifted to latent feature approaches [8–10]. Researchers such as Chang et al. [11] and Tao et al. [12] integrated bundling strategies by leveraging item bundle graphs and deep learning (BGCN and BundleNet). While these Graph Neural Network (GNN) models successfully captured hidden interaction patterns to improve prediction accuracy [13], Lubos highlighted their fundamental weakness [14], that the complex nonlinear transformations within these systems make the models act as black boxes. They fail to explain the cause and effect behind why certain items are grouped together, creating a significant trust gap for users who receive recommendations without any logical justification [15].

Furthermore, the complexity of semantic representation and candidate retrieval in current BRS remains a problem. Most BRS still operate under the "mono-intent" assumption [11, 16]. Sun et al. in their study, criticized that relying on frequently purchased products as the basis for bundle formation often results in noisy datasets, where many products are actually consumed without any underlying intent [17]. In other direction, Jiang et al. highlighted in their study, that relying solely on ID matching or historical co-occurrence methods suffers from data scarcity, where user interactions with bundles are much less frequent than with single items [18].

To address this issue, generative approaches using Large Language Models (LLMs) have emerged. Sun et al. proposed Adaptive Learning in Context (AICL) to infer user intent from previous sessions [19], while Hu et al. used generative LLMs for bundle generation, automatic naming, and annotation [20]. However, these generative methods still face serious computational challenges, such as combinatorial explosion [12, 21]. For solving this, they make bundle with limited number of items (e.g., to a maximum of 2 to 3 in one bundle) [22]. These things drastically reduce the flexibility

of the system, make recommendations uniform, and fail to accommodate the dynamic nature of real-world consumer needs.

To bridge the identified gap, we are creating the Semantic-Retrieval-Reasoning Pipeline (SRRP). SRRP is a novel hybrid framework that integrates LLM by combining Prompting and Embedding in a single pipeline. This framework does not separate technical accuracy from cognitive transparency but rather unifies them. Our overall key contributions are summarized as follows:

- We leverage LLM as a cognitive filter to dynamically decompose noisy user buckets into multi-intent clusters, eliminating reliance on rigid manual feature tuning and mono-intent assumptions.
- We replace traditional ID- or co-occurrence-based matching with dense vector database retrieval, ensuring that candidate retrieval maintains high relevance and intra-list diversity.
- We introduce a reasoning validation module to dynamically adjust bundle size, breaking the static constraints of previous generative models and generating persuasive natural language explanations and automatic naming. To ensure strict alignment with human cognitive consensus and prevent hallucinations, the system’s reasoning is rigorously validated against ground truth established by 9 independent human annotators through Human-in-the-Loop (HITL) methodology.

2 Related Work

The evolution of Bundle Recommender Systems (BRS) has undergone a significant transition. BRSs have evolved from constraint-based algorithms and latent feature modeling to the use of generative algorithms using Large Language Models (LLM). While relevance metrics continue to improve, a review of the State-of-the-Art (SOTA) literature reveals several fundamental gaps, such as limiting the system’s ability to present realistic and transparent item combinations.

2.1 Constraint and Latent-based Bundle Recommendation

Early approaches to BRS relied heavily on constraint features from a domain, such as price data or frequency of co-purchases [4–6]. Research by Yan Fang et al. [7] provides an example by establishing inter-item correlations using constraints and frequent itemset mining. Despite its simplicity, their approach produced monotonous recommendations with very low diversity [8].

To address this limitation, recent literature has shifted to latent feature approaches. Researchers such as Chang et al. [11] and Tao et al. [12] have integrated bundling strategies using item bundle graphs and deep learning (BGCN and BundleNet). While these models successfully capture hidden interaction patterns to improve prediction accuracy, Lubos highlight their fundamental weakness, the complex nonlinear transformations in these systems make the models act as black boxes [14]. They show that these models fail to explain the causality behind why certain items are grouped together, creating a significant trust gap for users who receive recommendations without logical justification [15].

2.2 Generative LLM in Bundle Recommendation

To overcome the black-box limitations, recent research has begun adopting generative approaches [11, 16]. Sun et al. proposed the In-Context Learning (ICL) paradigm to adaptively infer user intent based on their past interaction sessions, thus generating new bundles based on those intents [19]. Furthermore, Hu et al. successfully utilized a generative approach with LLM, which was used not only to construct bundles but also for auto-naming and explanatory narratives [20].

However, this shift in BRS methods toward semantic and generative search presents new computational challenges [12, 21]. First, systems often enforce static bundle sizes (e.g., limited to a maximum of three items) to avoid combinatorial explosion [22]. Furthermore, LLM is also prone to hallucinations [19], and it generates item combinations that are causally illogical in the real world or out of dataset [7]. This condition raises a critical question, can BRS adaptively adjust bundle size based on user context and intent, while still providing human-understandable explanations?

2.3 Research Position

Based on this research gap, we position the novelty of this research through the Semantic-Retrieval-Reasoning Pipeline (SRRP) architecture. Unlike previous SOTA studies, SRRP does not separate technical accuracy and cognitive transparency. Instead, we add a Reasoning Validation Unit (RVU) module that filters hallucinations, dynamically adjusts bundle size as needed, and provides naming and explanatory narratives validated directly by humans (Human-in-the-Loop). Comparatively, the position of our research contribution compared to the SOTA literature is comprehensively summarized in Table 1 below:

Table 1: Matrix Comparison of Research Position vis-à-vis State-of-the-Art (SOTA) Methods

Papers	Main Approach	Relevance	Diversity	Item Correlation	Dynamic Size	Interpretability	Auto Naming	Explainable
Yan Fang et al. [7]	Forming correlations between items using <i>constraint/frequent itemset</i>	✓	–	✓	–	–	–	–
Chang et al. [11] / Tao et al. [12]	Integrating <i>bundling</i> strategy by utilizing <i>bundle-item</i> graph (<i>Latent-based</i>)	✓	✓	✓	–	–	–	–
Sun et al. [19]	Forming <i>bundles</i> from user sessions adaptively using LLM (ICL)	✓	✓	✓	–	✓	–	–
Hu et al. [20]	Solving <i>bundle generation</i> , <i>auto naming</i> , and <i>explanation</i> with a generative LLM approach	–	–	✓	–	✓	✓	✓
Ours (SRRP)	Extracting <i>intent</i> with LLM and leveraging it in a <i>hybrid</i> model (<i>prompting & embedding</i>) equipped with a reasoning validation module	✓	✓	✓	✓	✓	✓	✓

3 Proposed Methodology

The method in this research centers on an integrated framework called the Semantic-Retrieval-Reasoning Pipeline (SRRP). Essentially, SRRP is a hybrid approach based

on the proposed Large Language Model (LLM), which innovatively combines prompting and embedding techniques. This hybrid approach is designed to address various weaknesses in conventional Bundle Recommender Systems (BRS), such as black-box issues, monotonous recommendations, and static bundle size limitations.

By combining prompting techniques such as In-Context Learning and Chain-of-Thought for semantic reasoning and intent extraction with embedding to transform interaction data into vector representations for similarity matching, this framework processes user shopping cart data through three main processing units to generate a list of relevant, dynamic, and validated bundle recommendations. An overview of the SRR framework workflow can be seen in Figure 1.

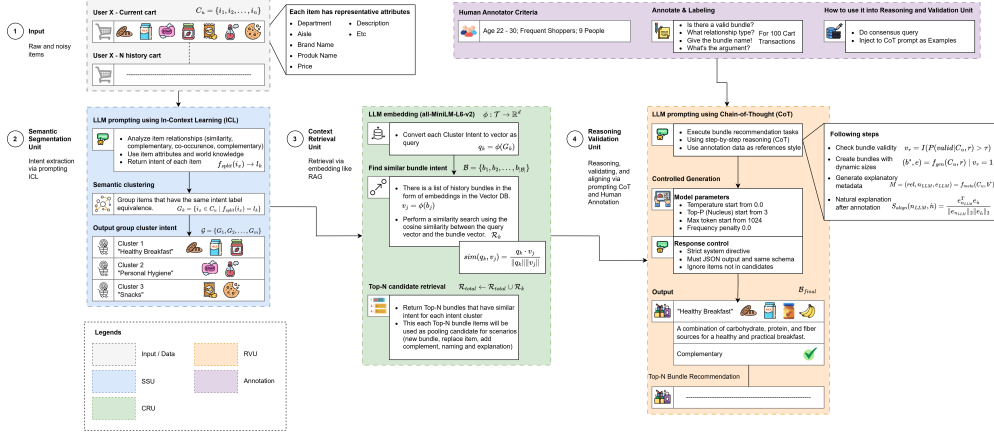


Fig. 1: SRRP System Block Diagram

3.1 Semantic Segmentation Unit (SSU)

To mitigate the signal dilution effect caused by shopping cart heterogeneity, this study proposes the design of a Semantic Segmentation Unit (SSU). Fundamentally, this module relies on the analytical capabilities of a LLM to perform high-level contextual analysis tasks as few-shot learners [23]. The SSU leverages the LLM to act as a cognitive filter through an In-Context Learning (ICL) mechanism inspired by research [19], which uses the Adaptive In-Context Learning method of the LLM to provide bundle recommendations.

Instead of processing the user's entire basket as in the study [19], at SSU we instructed LLM to cluster shopping items based on their syntactic relationships and semantic functions. The result of this clustering process is a set of deterministically isolated intent clusters, each with a very clean, meaningful focus without noise.

3.1.1 Mathematical Modeling

Formally, the current user's shopping cart is considered as a set of items $C_u = \{i_1, i_2, \dots, i_n\}$. Where the main goal of the SSU module is to group the set of items

C_u into a set of clusters *intent* $\mathcal{G} = \{G_1, G_2, \dots, G_m\}$ by labeling them using LLM. In the SSU architecture, we define a semantic mapping function f_{split} that will be executed by LLM, see Equation 1 for more details. This function is then able to map each item i_x in the bucket C_u to a specific intention label l_k collected in the cluster G_m :

$$f_{split}(i_x) \rightarrow l_k \quad (1)$$

where l_k is the cognitive intent descriptor resulting from LLM reasoning (e.g., "Breakfast" or "Office Supplies"). Next, each intent cluster G_k is formed by grouping items that have the same intent label equivalence, as formulated in Equation 2:

$$G_k = \{i_x \in C_u \mid f_{split}(i_x) = l_k\} \quad (2)$$

This mapping is required to satisfy strict partition constraints, namely $\bigcup_{k=1}^m G_k = C_u$ and $G_a \cap G_b = \emptyset$ for $a \neq b$. This mathematical constraint ensures that each clustering process produces a set of intent clusters that have a clean context, before each cluster is transformed into a search vector space.

3.1.2 Theorem and Proof

To justify why the clustering process results in a set of intent clusters that can mathematically improve recommendation quality, we formulate the Semantic Variance Reduction Theorem.

Theorem 1 (Semantic Variance Reduction in Intent Clusters). *Given a multi-intent shopping cart C_u with semantic vector representation $E(C_u)$. If the SSU performs clustering of C_u into isolated intent sub-clusters $\mathcal{G} = \{G_1, \dots, G_m\}$ based on functional relationship proximity [3], then the expected semantic variance within each cluster (intra-cluster variance) is strictly smaller than the semantic variance of the entire cart, thus significantly reducing the retrieval error bound.*

Proof. Let $\phi(x) \in \mathbb{R}^d$ be the spatial embedding function of item x . The semantic variance of the entire cart without clustering is the average squared distance of each item to the cart centroid μ_C , formulated in Equation 3:

$$Var(C_u) = \frac{1}{n} \sum_{x \in C_u} \|\phi(x) - \mu_C\|^2 \quad (3)$$

Based on the Law of Total Variance, the spatial variance of the basket can be decomposed into two main components, namely the expected variance within sub-clusters and the variance between cluster centers, which are formulated in Equation 4:

$$Var(C_u) = \underbrace{E[Var(G)]}_{\text{Intra-cluster}} + \underbrace{Var(E[G])}_{\text{Inter-cluster}} \quad (4)$$

Besides, since the original cart is multi-intent and may contain items that are not semantically related, the semantic distance between cluster centers is strictly positive, i.e., $Var(E[G]) > 0$. Therefore, it is mathematically proven that (See Equation 5):

$$E[Var(G)] = Var(C_u) - Var(E[G]) \implies E[Var(G)] < Var(C_u) \quad (5)$$

Q.E.D. This proof guarantees that representing a query using isolated intent G_k clusters will result in a much denser and more centralized search space, and minimize the effects of noise than forcing an average representation of the full C_u bucket.

3.1.3 SSU Algorithm and Prompt Design

The computational implementation of the bucket decomposition process above is executed using the Sequential Splitting algorithm. This algorithm is specifically designed to guide LLM in executing the clustering process deterministically:

Algorithm 1 *Semantic Segmentation Unit (SSU)*

Input: *User Cart* $C_u = \{i_1, i_2, \dots, i_n\}$

Output: *Intent Clusters* $\mathcal{G} = \{G_1, G_2, \dots, G_m\}$

```

1: Initialize  $\mathcal{G} \leftarrow \emptyset$ 
2: Construct instruction prompt  $P_{ICL}$  defining task and examples
3: // Step 1: LLM Cognitive Inference
4:  $Labels \leftarrow \text{LLM.Inference}(P_{ICL}, C_u)$  ▷ Executes  $f_{split}(i_x) \rightarrow l_k$  (Eq. 1)
5: // Step 2: Deterministic Semantic Grouping
6: Extract unique intent labels  $L = \{l_1, \dots, l_m\}$  from  $Labels$ 
7: for each unique label  $l_k \in L$  do
8:   Form cluster  $G_k \leftarrow \{i_x \in C_u \mid f_{split}(i_x) = l_k\}$  ▷ Formulated in Eq. 2
9:    $\mathcal{G} \leftarrow \mathcal{G} \cup \{G_k\}$ 
10: end for
11: return  $\mathcal{G}$  ▷ Minimizes intra-cluster variance  $E[Var(G)]$  (Eq. 5)

```

To execute the semantic segmentation process on the user’s shopping cart, the SSU module relies on the ICL modeling technique. This technique enables the LLM to adapt to the clustering task instantaneously by simply observing instructions and a few demonstrations, without requiring parameter weight updates (retraining). The instructions are injected through a structured prompt designed to make the LLM respond deterministically. The complete structure of this SSU prompt can be seen in Figure 2. Based on this, the prompt design is built on four main cognitive components to guide LLM reasoning:

1. The model is explicitly assigned a persona as a "Semantic Segmentation Unit" with the core task of analyzing and segmenting a list of items based on their usage occasion or semantic category. Defining this persona helps limit the scope of the LLM abstraction to keep it focused on understanding the e-commerce domain.
2. Since LLM is naturally a generative model that produces free-text, the system enforces very strict formatting rules. The first rule forces the output to be purely in JSON structure format (Array of Objects) without any opening or closing sentences (no explanatory text). This is crucial for the response to be directly algorithmically parsed by the system at a later stage. The other rules guarantee that no items are missed (comprehensive inclusion) and force LLM to generate representative intent labels (*cluster_name*).

```

You are a Semantic Segmentation Unit.
Task: Analyze the following shopping list and group the items
      based on their "Usage Occasion" or semantic category.

Rules:
1. Output ONLY a valid JSON format (Array of Objects). Do not
   include any other explanatory text.
2. Every input item must be included in one of the clusters.
3. Provide a descriptive name for the "cluster_name".

---
EXAMPLE 1 (Few-Shot):
Input: ["White Bread", "Peanut Butter", "Powder Detergent", "
       Fabric Softener"]
Output:
[
  { "cluster_name": "Breakfast", "items": ["White Bread", "
      Peanut Butter"] },
  { "cluster_name": "Laundry", "items": ["Powder Detergent",
      "Fabric Softener"] }
]

EXAMPLE 2 (Few-Shot):
Input: ["Spaghetti", "Bolognese Sauce", "Cheddar Cheese", "
       Shampoo", "Body Wash", "AA Batteries"]
Output:
[
  { "cluster_name": "Italian Dinner", "items": ["Spaghetti",
      "Bolognese Sauce", "Cheddar Cheese"] },
  { "cluster_name": "Personal Care", "items": ["Shampoo", "
      Body Wash"] },
  { "cluster_name": "Household Electronics", "items": ["AA
      Batteries"] }
]
---

ACTUAL INPUT:
Input: {cart_items}
Output:

```

Fig. 2: In-Context Learning (ICL) Prompt Template in the SSU module for clustering shopping cart items into semantic intent clusters

3. To align the LLM's reasoning with the expected clustering logic, the prompt is injected with demonstration examples (few-shot learning). The first example demonstrates a simple clustering like breakfast and laundry, while the second one provides a more complex scenario with three different intentions like Italian Dinner,

- Personal Care, and Household Electronics. These demonstrations serve as semantic anchors for the LLM to understand the desired depth (granularity) of clustering.
4. The final part of prompt is a dynamic variable `cart_items` which will be populated with the user’s raw shopping cart list in real-time as the inference is performed.

3.1.4 Real Example of SSU Application

The concrete validation example of about how that the Algorithm 1 works, we consider the following multi-intent problem-solving scenario:

- *Input* query: The user will sequentially place four random items into their the basket, let $C_u = \{\text{Banana, Whole Milk, Cheddar Cheese, Notebook}\}$.
- Conventional process: Traditional models represent this basket as a single mean vector. The center of the vector will fall in the ambiguous area between "Food" and "Stationery", resulting in ambiguous and contextually irrelevant recommendations.
- SSU Process (proposed):
 1. The LLM receives a query input, then uses an ICL prompt to execute cognitive reasoning f_{split} .
 2. LLM analysis results: $Banana \rightarrow \text{Fresh Fruits}$; $Whole\ Milk \rightarrow \text{Dairy Products}$; $Cheddar\ Cheese \rightarrow \text{Dairy Products}$; $Notebook \rightarrow \text{Stationery}$.
 3. The algorithm performs clustering and divides the items into a set of clusters *intent* $\mathcal{G}$:
 - Cluster G_1 (*Fresh Fruits*) = {Banana}.
 - Cluster G_2 (*Dairy Products*) = {Whole Milk, Cheddar Cheese}.
 - Cluster G_3 (*Stationery*) = {Notebook}.
- Conclusion: SSU successfully clustered the initially noisy data into three highly precise clusters. In the next stage, the Contextual Retrieval Unit (CRU), a specific and isolated search for each cluster will be performed. This clustering mechanism drastically improves relevance accuracy (Recall and Hit Rate) without sacrificing diversity (Intra-List Diversity / ILD and Cross-Bundle Diversity / $Div@K$).

3.2 Contextual Retrieval Unit (CRU)

We already know that LLM, which has high-level cognitive reasoning capabilities, its implementation purely as a single recommendation engine, especially on an industrial scale, will face two fundamental weaknesses: massive computational latency and susceptibility to hallucinations [24–26]. LLM’s dangerous and misleading responses when forced to generate recommendations from millions of product catalogs, such as in e-commerce, can even generate bundle combinations that fall outside the actual data [26].

Another problem we see lies in traditional search methods such as keyword-based exact match or conventional collaborative filtering techniques, which lack the ability to understand synonyms and semantics of user needs [27]. For example, if a user searches for "winter gear", they won’t get recommendations for "thick jackets" if the keywords aren’t exactly the same. Therefore, there’s a high urgency to be able to factually limit

the LLM search space using real historical data, while simultaneously capturing the semantics of user transaction intent [28].

Despite its superior cognitive capabilities, LLM implementations run the risk of encountering computational scalability barriers and are susceptible to data hallucinations. To address these challenges, this architecture is integrated with the Contextual Retrieval Unit (CRU), a module that adapts the principles of the Retrieval-Augmented Generation (RAG) architecture, where a vector-based search engine is used to retrieve factual candidates from the database before sending them to the LLM [29, 30].

Hearing this, we proposed that instead of searching for products using keywords, we configure the CRU module to transform the intent clusters generated by the SSU into high-dimensional dense vectors or embeddings using a Sentence-Transformer model such as `all-MiniLM-L6-v2` [31]. These vectors will then be used as queries in the historical *bundle* candidate search process in a vector database such as ChromaDB based on their spatial distance [32]. This will ensure that the retrieved bundle candidates actually exist in fact and are cognitively relevant.

3.2.1 Mathematical Modeling

Formally, given a set of user-deconstructed intent clusters $\mathcal{G} = \{G_1, G_2, \dots, G_m\}$. Also given is a set of historical candidate *bundles* from the entire platform population $\mathcal{B} = \{b_1, b_2, \dots, b_{|\mathcal{B}|}\}$. CRU defines a transformation embedding function $\phi : \mathcal{T} \rightarrow \mathbb{R}^d$ that maps the text $\mathcal{T}$ in the form of descriptions, names, or attributes of items in the intent cluster into a latent vector space of dimension d .

The query vector q_k for each cluster G_k and the document vector v_j for the historical bundle b_j are represented in Equation 6:

$$q_k = \phi(G_k), \quad v_j = \phi(b_j) \quad (6)$$

To measure the semantic relevance between a user query and a bundle candidate, the CRU evaluates the angular proximity between the vectors using the Cosine Similarity [33] metric, formulated in Equation 7:

$$\text{sim}(q_k, v_j) = \frac{q_k \cdot v_j}{\|q_k\| \|v_j\|} = \frac{\sum_{i=1}^d q_{k,i} \cdot v_{j,i}}{\sqrt{\sum_{i=1}^d q_{k,i}^2} \sqrt{\sum_{i=1}^d v_{j,i}^2}} \quad (7)$$

Based on this distance metric, for each query q_k , the CRU will return a set of raw candidates $\mathcal{R}_k$ consisting of the N_{top} historical *bundle* with the highest similarity scores.

3.2.2 Theorem and Proof

To prove that this spatial embedding-based search is more optimal than exact match search, we propose the Semantic Relevance Bound Theorem.

Theorem 2 (Improvement in Relevancy through Semantic Retrieval). *Let $\mathcal{V}_{exact}$ be the search space based on keyword matching, and $\mathcal{V}_{dense}$ be the latent search space using vector q_k . If there exists a candidate bundle b_j that is semantically relevant to q_k but has no keyword overlap, then $\mathcal{V}_{exact}$ will yield a zero hit probability*

(Recall), while $\mathcal{V}_{dense}$ ensures a high hit probability $P(b_j \in \mathcal{R}_k) \rightarrow 1$ as long as $\text{sim}(q_k, v_j) > \tau$, thus guaranteeing a lower bound on the Recall of CRU that is always greater than or equal to traditional search methods.

Proof. Assume $W(x)$ is the set of words in x . For query q_k and document b_j , if $W(q_k) \cap W(b_j) = \emptyset$, then the exact match search model like BM25 or Apriori association will give a relevance score $S_{exact}(q_k, b_j) = 0$. As a result, b_j will never be retrieved or will fail. However, in the latent space $\mathcal{V}_{dense}$, the function $\phi(\cdot)$ is trained using the cosine similarity objective [34, 35], which minimizes the Euclidean distance for positive semantic pairs even without word overlap, see Equation 8:

$$\|\phi(q_k) - \phi(b_j)\|^2 \approx 0 \implies \text{sim}(q_k, v_j) \approx 1 \quad (8)$$

Since $\text{sim}(q_k, v_j) \geq \tau$, where τ is the KNN query threshold, then b_j is guaranteed to be in the solution space of $\mathcal{R}_k$. **Q.E.D.** This proof mathematically guarantees that CRU will not miss valuable candidates that are often overlooked by conventional recommendation systems, thus empirically improving the Hit Rate and Recall values.

3.2.3 CRU Algorithm

The computational implementation of the vector transformation and candidate retrieval process on the vector database is executed using a spatial query-based nearest neighbor search algorithm:

Algorithm 2 Contextual Retrieval Unit (CRU)

Input: *Intent Clusters* $\mathcal{G} = \{G_1, \dots, G_m\}$, *Vector DB* $\mathcal{V}_{DB}$ containing historical bundle embeddings $\{v_1, \dots, v_{|\mathcal{B}|}\}$

Output: *Raw Candidate Bundles* $\mathcal{R}_{total}$

```

1: Initialize  $\mathcal{R}_{total} \leftarrow \emptyset$ 
2: for each intent cluster  $G_k \in \mathcal{G}$  do
3:   // Step 1: Text to Dense Vector Transformation
4:    $q_k \leftarrow \text{EmbeddingModel}(G_k)$  ▷ Executes  $q_k = \phi(G_k)$  (Eq. 6)
5:   // Step 2: Cosine Similarity Scoring
6:    $Scores \leftarrow \emptyset$ 
7:   for each  $v_j \in \mathcal{V}_{DB}$  do
8:      $score \leftarrow \frac{q_k \cdot v_j}{\|q_k\| \|v_j\|}$  ▷ Calculates  $\text{sim}(q_k, v_j)$  (Eq. 7)
9:      $Scores \leftarrow Scores \cup \{(b_j, score)\}$ 
10:  end for
11:  // Step 3: Top- $N_{top}$  Selection
12:   $\mathcal{R}_k \leftarrow \text{Extract } N_{top} \text{ bundles from } Scores \text{ with highest values}$  ▷ Retrieves candidates
   where  $\text{sim}(q_k, v_j) \approx 1$  (Eq. 8)
13:   $\mathcal{R}_{total} \leftarrow \mathcal{R}_{total} \cup \mathcal{R}_k$ 
14: end for
15: return  $\mathcal{R}_{total}$ 

```

3.2.4 Real Example of CRU Application

The concrete validation example of about how that the Algorithm 2 works, consider a real-world search failure scenario:

- SSU results as a query: The segmentation model detects a user intent cluster: $G_1 =$ "Equipment for work productivity".
- Conventional process (Association Rule / Exact Match): The system searches the catalog for products containing the words "Equipment", "Productivity", or "Work". The system fails to provide a satisfactory recommendation and instead suggests a book about "Work Guide" due to its lexical similarity.
- CRU process (proposed):
 1. The sentence "Equipment for work productivity" is converted into a numeric vector representation as $q_1 = [0.114, 0.220, 0.870, \dots]$.
 2. CRU instantly compares q_1 to millions of queries in the vector database (ChromaDB) using cosine similarity.
 3. Vector q_1 will mathematically reside in the latent space filled with electronic devices, because the embedding model already understands world knowledge.
- Conclusion: CRU successfully retrieved Top- N_{top} candidate bundles containing combinations of {Thinkpad Business Laptop, Logitech Wireless Mouse, Standing Desk}. Although none of the items contain the word "productivity", the spatial proximity (vector) successfully captured the implicit user intent. This candidate set was then passed to the final stage (Reasoning Validation Unit / RVU) for validation and narration.

3.2.5 Reasoning Validation Unit (RVU)

Conventional recommendation systems, including modern deep learning-based models, often face a fundamental problem known as the black box phenomenon [36]. While the CRU stage is capable of presenting candidate bundles based on vector distance proximity, the system is unable to articulate the logical reasoning behind these combinations to users [37].

This lack of explanation creates a trust gap. Users are skeptical of the system when recommendations are presented in seemingly random items, without any narrative explaining the added value of the bundle [37]. Furthermore, previous generative algorithms enforced static bundle sizes, such as always recommending 3 items [22], which completely misrepresents the dynamics of real-world shopping carts where user needs vary widely. Therefore, there is an urgency to develop a final-stage cognitive filter that not only logically selects candidates, avoiding LLM hallucinations, but also persuasively explains the reasoning behind those recommendations [38].

In the final stage, the gap between mathematical search results and the user's logical understanding is bridged through the Reasoning Validation Unit (RVU) component. This module is specifically designed to maximize the reasoning power of LLM, acting as a cognitive filter and post-search explanation agent [39].

The RVU applies the Chain of Thought (CoT) trigger technique [39–42] to simulate the human analytical thought process. This module evaluates the raw candidates

from the CRU and compares them to the user’s actual shopping cart. If any items are illogical, RVU discards them, automatically adjusts the bundle size to suit the user’s actual needs, and constructs persuasive explanatory text. To prevent LLM from hallucinating, RVU is controlled with strict system directives, the temperature parameter is kept close to 0, and few-shot demonstration examples are injected from an annotated Human-in-the-Loop (HITL) alignment dataset to align machine reasoning with human cognitive consensus. [43]

3.2.6 Mathematical Modeling

Formally, the inputs for RVU are user’s original shopping cart C_u and the raw *bundle* candidates $r \in \mathcal{R}_{total}$ generated by CRU. This cognitive validation process is modeled as a conditional probability function. For each candidate r , the LLM agent will evaluate and estimate the probability of its plausibility or validity $P(valid|C_u, r)$ based on the cart context. The binary validation decision v_r is determined by the indicator function I with a threshold τ , so it can be formulated as Equation 9:

$$v_r = I(P(valid|C_u, r) > \tau) \quad (9)$$

If $v_r = 1$ (the candidate is considered valid and logical), the system will proceed to the generation stage. We define the cognitive generation function f_{gen} which simultaneously generates the optimized bundle pair b^* and the explanation text e , formulated in Equation 10:

$$(b^*, e) = f_{gen}(C_u, r) \mid v_r = 1 \quad (10)$$

where b^* is a subset of r that has been dynamically resized so that $b^* \subseteq r$, and e is a text narrative that dissects causal relationships such as complementary or similarity between the items in b^* .

3.2.7 Theorem and Proof

To prove that dynamic modeling of RVU is mathematically superior to static *bundle* sizing systems, we formulate the Dynamic Sizing Optimality Theorem.

Theorem 3 (Dynamic Bundle Size Optimality). *Let M be the ideal number of complementary items logically required by cart C_u . If a conventional recommendation system always enforces a static bundle size K , then that system is guaranteed to produce inclusion errors greater than zero whenever $K \neq M$. Conversely, the generative function f_{gen} in RVU, which performs conditional pruning of candidates with $v_r = 1$, mathematically minimizes these errors approaching zero.*

Proof. Assume the recommendation utility function $U(b)$ for user cart C_u . In the static sizing system K , if the actual user need is $M < K$, the system must include $(K - M)$ irrelevant items (noise) just to fill the size quota. The presence of noise reduces the Precision of the bundle. On the other hand, if $M > K$, the system will discard $(M - K)$ actually relevant items, thereby reducing the Recall absolutely [17].

In the LLM reasoning function space, $f_{gen}(C_u, r)$ optimizes utility by evaluating the relevance probability of each item $x \in r$ as in Equation 11:

$$b^* = \{x \in r \mid P(x \text{ is logically complementary to } C_u) > \tau\} \quad (11)$$

Since the set b^* is constructed purely from satisfying conditional probabilities without any scalar cardinality (absolute size) constraints, the final size of $|b^*|$ will naturally converge to the original requirement M , that is, $|b^*| \approx M$. Therefore, the expected inclusion error has the value given by Equation 12:

$$||b^*| - M| \approx 0 \quad (12)$$

Q.E.D. This proof ensures that RVU does not impose fanciful (hallucinatory) product recommendations just to fill the gap, thus drastically increasing precision and user satisfaction.

3.2.8 RVU Algorithm and Prompt Design

The implementation of reasoning logic and generation of explanatory text in RVU is executed through the following algorithm:

Algorithm 3 *Reasoning Validation Unit (RVU)*

Input: *User Cart* C_u , *Raw Candidates* $\mathcal{R}_{total}$ dari CRU
Output: *Final Recommendations* $\mathcal{B}_{final}$

```

Initialize  $\mathcal{B}_{final} \leftarrow \emptyset$ 
for each candidate  $r \in \mathcal{R}_{total}$  do
  // Step 1: Prompt Construction
  Construct prompt  $P_{val}$  combining  $C_u$ , candidate  $r$ , CoT instructions, and HITL
  examples
  // Step 2: Validation Inference
   $Output \leftarrow \text{LLM\_Inference}(P_{val})$ 
  Extract validity decision  $v_r$  ▷ Evaluates  $v_r = I(P(valid|C_u, r) > \tau)$  (Eq. 9)
  // Step 3: Conditional Generation
  if  $v_r == 1$  then
    Extract dynamically sized bundle  $b^*$  (discarding noise items) ▷ Executes
     $(b^*, e) = f_{gen}(C_u, r)$  (Eq. 10, 11)
    Extract persuasive explanation narrative  $e$ 
     $\mathcal{B}_{final} \leftarrow \mathcal{B}_{final} \cup \{(b^*, e)\}$ 
  end if
end for
Rank  $\mathcal{B}_{final}$  based on LLM confidence scores
return  $\mathcal{B}_{final}$  ▷ Ensures inclusion error  $||b^*| - M| \approx 0$  (Eq. 12)

```

To execute the cognitive validation process and the formation of the final bundle, the RVU module uses a strict instruction-based prompt that implements the CoT reasoning element. The structure of this RVU prompt can be seen in Figure 3. Based on this, the RVU prompt design is built with several key components to ensure the LLM does not hallucinate and produces logical output:

```

You are an AI Recommender System expert in creating Product Bundling.

CONTEXT:
- Current User Cart: {current_items}
- Recommendation Candidates (from database): {candidate_items}

TASK:
Analyze the candidate items and group them into relevant "
Recommendation Bundles".

IMPORTANT RULES:
1. A "Bundle" must consist of a combination of: USER ITEMS (as
   context) + RECOMMENDED ITEMS (as complements).
2. Do NOT recommend items that are EXACTLY THE SAME as the
   user's items.
3. Provide a creative bundle name.
4. NUMBER OF BUNDLES: Create at least 2 bundles. If the
   candidate items are numerous and diverse, you may create
   up to 5 bundles. Do not force a small number if the
   variety is high.

OUTPUT FORMAT (JSON REQUIRED):
{
  "reasoning_summary": "Brief explanation...",
  "suggested_bundles": [
    {
      "bundle_name": "Bundle Name",
      "user_items": ["User Item A", "User Item B"],
      "recommended_items": ["Recommendation X", "
        Recommendation Y"],
      "reason": "Reason why the user items and
        recommendations are suitable to be combined"
    }
  ]
}

Output ONLY the valid JSON format above.

```

Fig. 3: Reasoning Validation Unit (RVU) Prompt Template for bundle generation and explanatory narratives. The prompt is designed to trigger the Chain-of-Thought reasoning mechanism in LLM, ensuring that the model evaluates the logical relationships between user cart items and candidate recommendations before generating the final bundle and explanation.

1. **System Role and Context Injection:** The model is explicitly assigned a persona as a recommender system expert. Context is dynamically injected through the `current_items` and `candidate_items` variables, taken from previous CRU outputs. This ensures that generated recommendations are always based on actual candidate data (mitigation of hallucination).
2. **Constraints and Rules:** The system applies strict rules to prevent redundancy (prohibiting the recommendation of identical items). Furthermore, the system implements dynamic sizing control by giving the LLM the freedom to create 2 to 5 bundles depending on the level of candidate variation. This rule directly addresses the static bundle size limitations of conventional methods.
3. **Structured CoT Output:** The output is forced into a JSON structure format, requiring the LLM agent to provide the `reasoning_summary` and `reason` parameters. This constraint essentially triggers the CoT reasoning mechanism, where the model must reason and articulate logical reasons (causality) before presenting the final combination. The result of these parameters is the persuasive explanatory text (explainability) to the user.

3.2.9 Real Example of RVU Application

To concretely illustrate how the mathematical Algorithm 3 and the cognitive logic of the RVU work, consider the following scenario on an e-commerce platform:

- **Cart input (C_u):** The user has entered [Sony A6000 Mirrorless Camera].
- **Raw candidate from CRU (r):** The vector system finds a set of items based on historical spatial proximity: {Sony 50mm Lens, NP-FW50 Spare Battery, Mini Tripod, Portable Fan}. (The fan is excluded due to random vector proximity noise.)
- **RVU process (proposed):**
 1. **Validation (reasoning):** The LLM uses CoT to analyze the relationship. The LLM detects that the lens, battery, and tripod have a functionally complementary relationship with the camera. However, "Portable Fan" is deemed to have no correlation with photography intent ($v_r = 0$ for the item).
 2. **Dynamic sizing:** RVU intelligently reduces the candidate size from 4 items to 3 items ($b^* = \{\text{Lens, Battery, Tripod}\}$) or adds and replaces items.
 3. **Explainability:** RVU generates explanatory arguments e using a persuasive lexicon.
- **Final output to the user:**
 - Bundle name: "Sony Photography Essentials Package"
 - Bundle contents: 50mm Lens, NP-FW50 Battery, Mini Tripod.
 - Explanatory text: "To maximize your Sony Camera photography experience, we recommend the 50mm Lens for sharp portraits, a spare battery for longer, continuous shooting sessions, and a very practical mini tripod for image stability".
- **Conclusion:** The system has shifted from a black-box recommendation system to a transparent, intelligent shopping assistant. Users not only see the items but also understand why they should buy them, ultimately increasing trust and transaction conversions.

3.3 Cognitive Metadata Generation and Alignment Testing

Conventional recommendation systems are generally black-box [36]. After the system successfully compiles the optimal combination of items, it still struggles to articulate the rationale for that combination to users. This lack of explanation often raises doubts about the recommendations presented [44].

Therefore, this study utilizes LLM not only to compile bundles but also to generate cognitive metadata in the form of relationship types, attractive bundle naming (auto-naming), and persuasive explanatory text [37]. However, evaluating the logicity of a name and description is highly subjective and cannot be measured simply by mathematical metrics of exact match. This raises the urgency of building a ground truth based on the Human-in-the-Loop (HITL) paradigm to measure the alignment between generative reasoning and the consensus of actual human perception.

3.3.1 Metadata Generation Modeling

Formally, after RVU extracts a validated bundle $b^* \subseteq C_u$, the LLM agent is employed as an estimator of the probability distribution $P_\theta(\hat{y}|C_u, P)$ where P is the prompt and θ are the model parameters. We define Equation 13 which is a cognitive generative function f_{meta} that takes a bundle b^* and a user’s bucket C_u to generate cognitive metadata M :

$$M = (rel, n_{LLM}, e_{LLM}) = f_{meta}(C_u, b^*) \quad (13)$$

where rel is a relation type such as complementary, similarity, or contextual, n_{LLM} is a predicted name bundle, and e_{LLM} is an autoregressively assembled persuasive explanatory narrative.

3.3.2 HITL Ground Truth Modeling and Robustness Theorem

Given the high risk of bias when subjective evaluation is based solely on a single individual, the validity of the metadata output M is assessed using a ground truth reference constructed from the Human-in-the-Loop (HITL) paradigm. Through this stage, we involve $K = 9$ independent annotators. Each annotator k assigns their annotation label $y^{(k)}$ to the logicity of the bundle and its metadata. The cognitive ground truth aggregation $\hat{y}$ is calculated using a majority voting mechanism [45] with the equation 14:

$$\hat{y} = \arg \max_{c \in C} \sum_{k=1}^K I(y^{(k)} = c) \quad (14)$$

where $I(\cdot)$ is an indicator function that is 1 if the condition is met, and C is a possible class space or consensus. To mathematically justify that these human annotations are valid and worthy of use as a standard, we formulate the following consensus robustness theorem:

Theorem 4 (HITL Consensus Robustness in Cognitive Evaluation). *Assume each annotator k has an independent probability $p > 0.5$ of correctly evaluating and formulating the cognitive intent T . Then, the probability that the human ground*

truth aggregation $\hat{y}$ fails to represent the actual logical intent will decay exponentially as the number of participants K increases.

Proof. Let $X_k = I(y^{(k)} = T)$ be an independent Bernoulli random variable with success probability p . The total number of correct annotations $X = \sum_{k=1}^K X_k$ follows a Binomial distribution. The majority voting consensus $\hat{y}$ will produce an incorrect decision only if $X \leq \lfloor \frac{K}{2} \rfloor$. Based on the Hoeffding (Chernoff Bound) inequality for bounded random variables, the probability of consensus failure is bounded by the upper limit in Equation 15:

$$P\left(X \leq \frac{K}{2}\right) \leq \exp\left(-2K\left(p - \frac{1}{2}\right)^2\right) \quad (15)$$

Since $p > 0.5$, the value of $\left(p - \frac{1}{2}\right)^2$ is a positive constant. Therefore, when the number of participants is set at $K = 9$, the upper bound of the error probability is drastically suppressed following an exponential decay curve of $O(e^{-K})$. This proof mathematically guarantees that the resulting ground truth is highly robust against individual bias and is a valid representation of human cognition for LLM measurement. **Q.E.D.**

3.3.3 Semantic Alignment Evaluation Formula

Because morphological variations, such as the vocabulary choices between LLM and humans, may differ despite their equivalent meaning, the feasibility of LLM predicted text cannot be measured using absolute character matching metrics. The system tests this alignment using distance calculations in semantic latent space.

The LLM predicted texts n_{LLM} and e_{LLM} and the human ground truth texts $\hat{n}$ and $\hat{e}$ are mapped into a dense vector representation of dimension $\mathbb{R}^d$ using the text encoder model. The semantic alignment between the model and human reasoning is then measured using the cosine similarity S_{align} in Equation 16:

$$S_{align}(n_{LLM}, \hat{n}) = \frac{e_{n_{LLM}}^T e_{\hat{n}}}{\|e_{n_{LLM}}\|_2 \|e_{\hat{n}}\|_2} \quad (16)$$

If $S_{align} \approx 1$, the model is mathematically validated to have human-likeness that aligns with human cognitive perception without indicating text hallucination.

To implement the cognitive evaluation and metadata generation tasks, the LLM agent is instructed to use the Dynamic CoT Prompting approach. The prompt is specifically designed to force the model to evaluate the coherence of shopping carts and align its reasoning with the Human-in-the-Loop (HITL) consensus. Unlike the injection of curated examples from 9 annotators matching the query, the system reads the curated data from 9 annotators programmatically from a CSV format, then injects it into the instruction space dynamically. The complete structure of this alignment prompt can be seen in Figure 4. Based on this, the prompt design is precisely constructed to align machine reasoning with human cognitive perception through several key elements:

1. System Persona and Psychological Framing: The LLM is given an explicit role as a cognitive agent expert in market basket analysis and consumer psychology. Defining this persona significantly helps orient the LLM beyond acting as a text generator,

```

System Persona: "You are an expert cognitive agent in e-commerce market basket analysis and consumer psychology. Your task is to evaluate a list of products purchased together in a single user session and determine if they represent a cohesive, intentional product bundle or just random, noisy purchases. You must return your evaluation strictly in JSON format."

Instructions: "Given the 'List of Products', analyze the underlying user intent and generate your evaluation. Your output must be a valid JSON object containing exactly the following four keys:

1. "valid": Determine if the items logically form a cohesive bundle. Value must be exactly "Yes" if they share a logical intent, or "No" if the cart is noisy and unrelated .
2. "relationship_type": If "valid" is "Yes", classify the primary relationship tying the items together. Choose exactly one from the following options:
  - "Complementary" (items consumed together to achieve a specific goal)
  - "Similarity" (items that share the same category or function)
  - "Contextual" (items bought for a specific routine or time )
3. "category_name": If "valid" is "Yes", generate a short, catchy category name for this bundle (maximum 5 words). If "valid" is "No", value must be "-".
4. "explanation": Provide a brief, logical explanation in Indonesian of why these items are purchased together (if Valid), or why they are considered random and unrelated (if Invalid).

Output only the JSON object. Do not include any preamble, explanations, or markdown blocks."

Examples from Curated Annotators:
{dynamic_cot_snippet}

ACTUAL INPUT:
Input: "List of Products: {list_of_items}"
Output:

```

Fig. 4: Dynamic Few-Shot Prompt Template for Human-LLM Alignment Evaluation and Cognitive Metadata Generation in the RVU module

but rather as a critical behavior analyst, distinguishing intentional transactions from merely random ones.

2. **Deterministic JSON Schema:** To maintain inference stability comparable to measurable testing, the model is forced to respond in an absolute JSON structure with four main parameters: a binary validity decision, a three-way relationship classification (`relationship_type`), a shorthand naming extraction (`category_name`), and a causality argument (`explanation`).
3. **Relational Taxonomy Injection:** The system explicitly defines three basic relation taxonomies recognized in recommendation systems: complementary, similar, and contextual. This rule injection prevents LLM from hallucinating by creating relation categories outside the tested standard.
4. **Dynamic Ground Truth Demonstrations (Dynamic CoT Learning):** The most crucial element is the injection of real-world demonstrations dynamically configured through the dynamic CoT snippet variable. The system automatically loads samples from the nine annotator consensus dataset (HITL). This mechanism not only cleans instructions but also serves as a dynamic cognitive calibration to ensure LLM evaluation aligns with human perception.

3.3.4 Real-Life Examples of Naming Applications and Alignment Tests

To understand the concrete application of the alignment function, consider the following cognitive metadata generation scenario:

- **Input query (C_u):** The user enters $\{Organic Romaine Spinach, Whole Grain Quinoa, Yellow Bell Pepper, Greek Yogurt\}$.
- **Ground truth human ($\hat{n}, \hat{e}$):** Based on the consensus of 9 annotators (via majority voting), the Name is agreed upon: "Healthy and Delicious Salad" and the Description: "The combination of fresh vegetables, quinoa, and the accompanying dressing complement each other to create a well-rounded salad dish".
- **LLM Prediction (n_{LLM}, e_{LLM}):** The LLM model generates the Name: "Nutritious Diet Salad Pack" and the Description: "A collection of organic vegetables and grains mixed with yogurt for a healthy dressing for everyday meals".
- **Alignment calculation:** Although the lexicon used by the annotator is different from that of the LLM, the text encoder model maps these two pairs of texts to adjacent quadrants of the latent vector. The equation 16 returns a score of 0.91, empirically validating that the LLM agent successfully performs naming and argument extraction that are fully aligned with human cognitive reasoning.

4 Experimental Setup

The experiments in this study require data that can represent real-world user shopping behavior, heterogeneity, and interrelationships between products within a single transaction. Therefore, the primary dataset used is the Instacart Online Grocery Shopping Dataset 2017 ¹.

¹<https://github.com/ChristopherCochet/Market-Basket-Analysis>

The Instacart dataset was chosen because it contains large-scale historical online shopping data that records the chronological sequence of users’ market basket transactions, complete with information on shopping times (timestamps) and inter-order intervals. Each transaction in this dataset reflects the natural structure of a shopping basket containing various product combinations (multi-intent), making it ideal for testing the case of organizing baskets into bundle recommendations. Detailed raw statistics data from the Instacart 2017 dataset are outlined in Table 2.

Table 2: Raw Statistics of the 2017 Instacart Dataset

Component	Initial Count	Description
Users	206,209	Each user has a sequence of shopping transactions tied to a timestamp.
Orders	3,421,083	Includes all historical orders, training data, and testing data.
Products	49,688	Unique products provided with text names and IDs.
Aisles	134	Specific product categories (e.g., fresh vegetables, candy chocolate).
Departments	21	High-level categories (e.g., produce, dairy eggs, beverages).

Dataset quality is crucial for the stability of LLM embedding model and reasoning agent training. Therefore, a rigorous preprocessing process is implemented to eliminate outliers and mitigate sparse data issues. The most crucial step in this phase is the implementation of a 5-core filtering method, where the system exclusively filters users and products with a history of at least 5 transactions. This ensures that each analyzed user or product entity has sufficient interaction patterns to be recognized by the machine learning model.

Table 3: Statistics of the 2017 Instacart Dataset After Applying 5-Core Filtering

Component	Filtered Count
Users	175,071
Orders	3,230,181
Products	47,664
Aisles	134
Departments	21

After the 5-core filtering process was executed, the interaction matrix density became more solid. Details of the final statistics used in the testing phase are shown in Table 3.

4.1 Baseline Model

The modified Instacart dataset is then used to test whether our proposed SRRP model is superior to current State-of-the-Art (SOTA) methods. Given the focus of

this dissertation on bundle recommendation and generation, we compare the proposed framework with five baseline models representing the evolution from classical data mining to modern graph models:

1. Apriori. A classic association rule mining algorithm that extracts frequent itemsets (combinations of items frequently purchased together) to statistically construct bundles.
2. FP-Growth. An advanced frequent itemset mining algorithm that builds an FP-tree structure. It is computationally more efficient than Apriori in formulating bundle associations by eliminating the need for repeated candidate generation steps.
3. BundleNet. A neural network model based on Graph Convolutional Networks (GCN) that unifies users, items, and bundles into a single heterogeneous tripartite graph to model high-order connectivity.
4. BGCN (Bundle Graph Convolutional Network). An advanced model that deconstructs user-bundle interactions into two distinct graph views (bundle view and item view), applying node propagation to capture deeper collaborative signals.
5. CrossCBR (Cross-view Contrastive Learning for Bundle Recommendation). A state-of-the-art model that extends the BGCN architecture by incorporating contrastive learning. It enforces alignment between the bundle view and item view to learn robust and self-supervised feature representations.

These five models were chosen to demonstrate the evolutionary roadmap for bundle recommendation. They range from purely statistical and non-personalized approaches (Apriori, FP-Growth) to modern, robust graph collaboration-based Deep Learning architectures (BundleNet, BGCN, CrossCBR). However, the underlying reason for this comparison is to highlight the fundamental limitations of these models: Apriori and FP-Growth are limited to statistical emergence, which is susceptible to data sparsity, while BundleNet, BGCN, and CrossCBR rely heavily on black-box architectures based on ID (graph node) connectivity, which ignores the rich semantics of language attributes and cognitive reasoning. This test aims to prove that our LLM-based hybrid architecture, SRRP, is able to solve this stagnation with more accurate, dynamic, and interpretable performance.

4.2 Three Splitting and Masking Strategies

In real-world retail environments, user interactions are rarely ideal. Users often randomly add items, shop in an illogical order, or split their shopping intentions across multiple sessions. To evaluate the robustness of the proposed model in the face of such uncertain data, we modified the test cart set into three main splitting or masking scenarios:

1. Sequential Splitting (SS). Sorts the cart contents by time (chronologically), then splits the end as the prediction target (Ground Truth), and uses the initial items as input context. This strategy tests the model’s ability to understand the evolution of user needs and suggest item bundles precisely based on historical information.
2. Random Masking (RM). Stochastically (randomly) hides approximately 30% of the items in the shopping cart at any position without regard to time order. This

strategy aims to test the reasoning model’s ability to capture the macro-intent of the shopping cart and suggest bundles as complementary products to the hidden items.

3. Random Cart Order (RCO). Similar to RM, but the entire order of items in the basket is completely randomized before being fed to the system model. The goal is to prove that the proposed architecture understands the context of the items through the holistic meaning generated by LLM, and not solely due to biases in the items’ sequential positions.

4.3 Research Scenario

After the architecture was built and the Instacart dataset was prepared using three slicing strategies (SS, RM, RCO), an experimental protocol was run specifically to answer the three main Research Questions (RQs) of this study:

- The scenario addresses RQ1 (Optimal Relevance and Diversity). RQ1 asks how a generative algorithm-based approach can construct bundles with high relevance and optimal item diversity. Validation of the first research question (RQ1) is realized through empirical comparisons between the proposed architecture and five baseline models, focusing on the Relevance, Ranking, and Diversity metrics.
- The scenario addresses RQ2 (Dynamic Bundle Size Determination). RQ2 asks how a generative-based model can be developed to dynamically determine the number of items to avoid excessive item combinations. This experimental scenario focuses on applying the LLM model to three noisy masking data conditions (SS, RM, RCO). This test validates the effectiveness of LLM in preventing the system from forming rigid, static sizes, allowing bundle sizes to naturally adapt to basket needs, distribute them predominantly within a reasonable range, and be resilient to accommodate long-tail distribution.
- The scenario item addresses RQ3 (Automated Feature Engineering, Interpretability, and Alignment Testing): RQ3 focuses on how utilizing LLM as Automated Feature Engineering, integrated with a reasoning validation mechanism, can suppress hallucinations and provide interpretable recommendations that align with user perceptions. This scenario is answered by analyzing the intent features, naming, and explanatory narratives automatically generated by the LLM agent. The goal is not only to demonstrate that the reasoning for bundle formation is successfully explained transparently but also to test the alignment between the model output and human cognitive consensus. This test uses Human-in-the-Loop (HITL)-based ground truth to ensure that the machine’s logical reasoning is truly valid, so that the recommendation decision-making process is no longer black-box and can be fully trusted by users.

4.4 Evaluation Protocols and Metrics

Testing an LLM-based Recommender System model is not sufficient by relying solely on classification loss calculations. The system must be evaluated using real-world protocols to ensure its ability to handle the complexity of user behavior. In this study, the evaluation protocol used is the Leave-One-Out Evaluation approach. In this protocol:

1. The last interaction or purchase session of each user is set aside as ground truth.
2. All items that have never been interacted with by the user are considered negative samples.
3. The model is tasked with predicting the next item or bundle based on a series of historical user contexts.

In line with the shifting evaluation paradigm in modern Bundle Recommender Systems, the performance of this framework is measured not only partially, but through three holistic evaluation dimensions that combine accuracy, diversity, and cognitive text quality.

4.4.1 Ranking and Relevance Metrics

Historically, the performance of recommendation systems has been evaluated exclusively using quantitative mathematical metrics focusing on prediction accuracy, ranking, and diversity. To evaluate the quality of item composition in a generated bundle, the literature has standardly adopted metrics such as Recall. According to the research of *Hu et al.* [20], this metric measures the degree of similarity between the predicted bundle (b^*) and the actual target bundle ($\hat{b}$) in the evaluation dataset ($\mathcal{D}$). Mathematically, this metric is formulated as Equation 17:

$$Recall = \frac{1}{|\mathcal{D}|} \sum_{(u,q,b^*,\hat{b}) \in \mathcal{D}} \frac{|b^* \cap \hat{b}|}{|\hat{b}|} \quad (17)$$

In the equation 17, u represents the user and q is the query or search context. The operator $|\cdot|$ represents the number of items in the set bundle. Furthermore, to measure the effectiveness of the recommendation list as a whole (ranking quality), the literature review [13] uses the Hit Rate (HR) and Mean Reciprocal Rank (MRR) metrics which are respectively formulated in the equations 18, and 19:

$$HR = \frac{\sum_{u,b} I(\hat{y}_{ub} > threshold \wedge y_{ub} > 0)}{\sum_{u,b} I(y_{ub} > 0)} \quad (18)$$

$$MRR = \frac{1}{|U|} \sum_{u=1}^{|U|} \frac{1}{rank(\hat{b}_{u*})} \quad (19)$$

in the ranking formula above, $\hat{y}_{ub}$ is the model's predicted probability score of user u 's interaction on bundle b , while y_{ub} is the actual interaction status. The indicator function $I(\cdot)$ is 1 if the condition is met, and 0 otherwise. The variable $|U|$ represents the total number of users evaluated, and $rank(\hat{b}_{u*})$ is the target bundle's ranking position in the recommendation list.

4.4.2 Diversity Metrics

To ensure the system does not recommend monotonous items or bundles, the evaluation should also consider diversity metrics. Diversity within a single bundle Intra-List Diversity (ILD) measures the average pairwise distance between items in a set of recommendations $\mathcal{R}$, as shown in Equation 20:

$$ILD = \frac{1}{|\mathcal{R}|(|\mathcal{R}| - 1)} \sum_{i \in \mathcal{R}} \sum_{j \in \mathcal{R}} d(i, j) \quad (20)$$

where $d(i, j)$ is the dissimilarity score between items i and j .

4.4.3 Text and Cognitive Evaluation Metrics

To ensure the objectivity of text evaluation, this study uses a quantitative approach by measuring linguistic parameters in the generated output. We formally define two main metrics. The first is Lexical Diversity (Distinct-N), which measures vocabulary variation to detect the level of repetition in sentences. We use Distinct-2 (bigrams), which is defined as the ratio of the number of unique word pairs to the total number of word pairs in the corpus B . The mathematical equation is:

$$\text{Distinct-2} = \frac{||\text{unique}(B)||}{||\text{total}(B)||} \quad (21)$$

The second metric is the Persuasiveness Index, which measures the density of words containing value propositions. The persuasiveness score for a text E is calculated using the following indicator function:

$$S_{pers}(E) = \sum_{w \in E} \mathbb{I}(w \in \mathcal{L}_{expanded}) \quad (22)$$

Unlike previous research that often only used a list of generic positive adjectives, the $\mathcal{L}_{expanded}$ lexicon in this study was constructed specifically based on e-commerce copywriting principles. This lexicon encompasses four key psychological dimensions that drive purchasing decisions:

1. **Functional Relevance.** Keywords that emphasize the suitability and usefulness of recommended items to the user’s basket (e.g., "suitable," "complementary," "ideal," "combination").
2. **Sensory Quality.** Keywords that describe attributes of taste, aroma, or texture, which are crucial in the grocery domain (e.g., "delicious," "fresh," "crunchy," "savory").
3. **Health Benefits.** Keywords that target users’ health consciousness and lifestyle (e.g., "healthy," "nutritious," "organic," "low-fat").
4. **Economic Value & Practicality.** Keywords that offer incentives in the form of time or cost efficiency (e.g., "economical," "convenient," "affordable," "ready-to-eat").

The use of this multidimensional lexicon ensures that high persuasiveness index scores truly reflect the system’s ability to construct substantive sales narratives, not merely polite platitudes.

G-Eval (LLM-as-a-Judge) is a qualitative evaluation framework using a Likert-based probability aggregation system (1-5) to assess the Coherence and Fluency aspects of recommendation rationales. The calculation is directly based on the equation 23.

$$G\text{-Eval Score} = \sum_{v=1}^5 p(v) \cdot v \quad (23)$$

where v is a score option (e.g., 1 to 5) and $p(v)$ is the probability of occurrence of a token with score v generated by the LLM based on the cognitive rubric within that range.

To capture the qualitative nuances and subjectivity of bundle logic that are difficult to measure mathematically, alignment evaluation is performed by comparing the predictions of offline cognitive agents (Gemma-3, LLaMA-3, Qwen-3) with the ground truth annotation results from Human-in-the-Loop (HITL) with 9 annotators. This alignment dataset is built from the consensus of nine independent annotators, measured by Cosine Similarity in the semantic latent space. This measurement dissects alignment into two specific metrics: category naming alignment and explanatory argument alignment.

5 Results and Discussion

This section presents the results of a comprehensive evaluation of the proposed Semantic-Retrieval-Reasoning Pipeline (SRRP) architecture. A multidimensional analysis is conducted to demonstrate the research’s scientific contributions, including its relevance and diversity, dynamic bundle size adjustment, explainability, and alignment of machine reasoning with human cognitive perception.

5.1 Relevance, Ranking, and Diversity Analysis

The system performance is evaluated using metrics such as *Hit Ratio* (HR), *Recall*, and *Mean Reciprocal Rank* (MRR) to measure accuracy, along with *Intra-List Diversity* (ILD) to assess the level of item variety within the *bundle*. This test compares three derived strategies of SRRP: *Sequential Splitting* (SS), *Random Masking* (RM), and *Random Cart Order* (RCO), against five *baseline* models (*Apriori*, *FP-Growth*, *BundleNet*, *BGCN*, and *CrossCBR*). The comparative performance results for the *Top-20* recommendations are summarized in Table 4.

Table 4: Performance Comparison of Relevance and Diversity on Top-20 Recommendation

Model / Approach	Hit Rate	Recall	MRR	Diversity (ILD)
Apriori	0.0200	0.2500	0.0200	0.0548
FP-Growth	0.0200	0.2500	0.0200	0.0548
BundleNet	0.0100	0.0461	0.0008	0.6362
BGCN	0.0400	0.0896	0.0029	0.6346
CrossCBR	0.0500	0.0942	0.0065	0.6359
SRRP - Random Masking	0.1100	0.1298	0.0683	0.5421
SRRP - Random Cart Order	0.0600	0.1317	0.0475	0.5314
SRRP - Sequential Splitting	0.0700	0.1502	0.0500	0.5358

Based on the actual data in Table 4, the SRRP architecture demonstrates a clear superiority in recommendation accuracy compared to all baseline models in the Top-20 test. In terms of Hit Rate and Recall, the RM approach surpassed 11.0%, and the SS strategy surpassed the Recall metric by over 15.02%. These results convincingly outperform state-of-the-art models like CrossCBR, which only achieved a Hit Rate of 5.0% and a Recall of 9.42%. Pure graph models (BundleNet and BGCN) struggle significantly on this dataset in predicting the correct package relevance for user queries.

In terms of diversity (*Diversity*), graph models like BundleNet and CrossCBR do indeed record the highest scores (around 0.63), but this is insignificant considering their very low accuracy levels. On the other hand, SRRP is able to provide the most optimal trade-off. All SRRP experimental scenarios consistently maintain *Intra-List Diversity* values above 0.53 (reaching 0.5421 in RM). This achievement proves that the SRRP system not only excels in accurately guessing preferences but is also capable of presenting a semantically rich variety of *bundles* to prevent user boredom. Furthermore, expanding the search space in SRRP has been shown to consistently increase *Recall*, indicating the effectiveness of the *Contextual Retrieval* unit in capturing targets in the latent space.

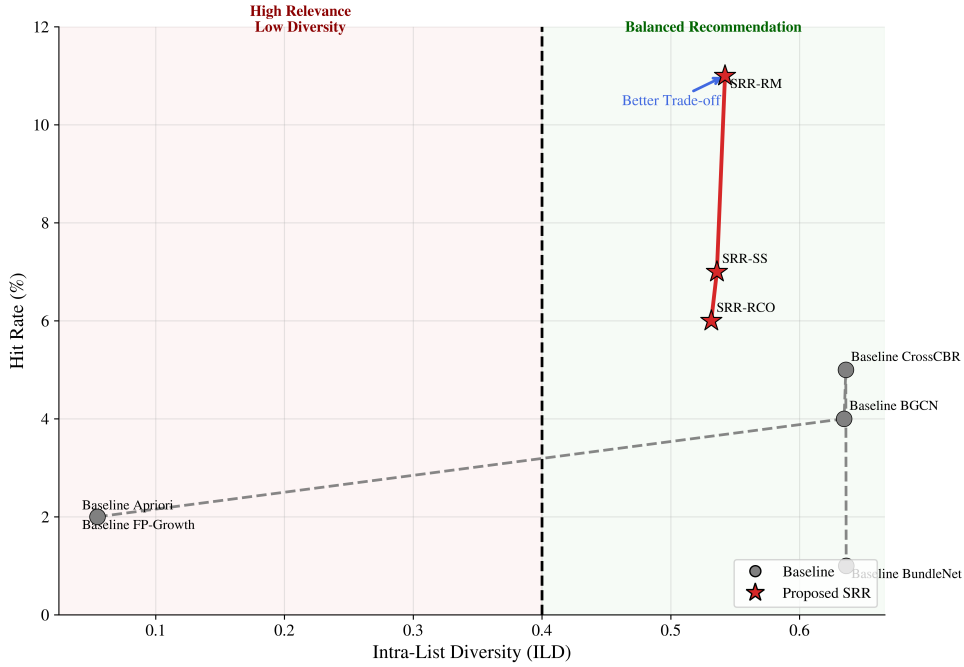


Fig. 5: Pareto trade-off comparison between accuracy and diversity across baseline models and SRRP strategies

In Figure 5, it is clearly visible that all SRRP strategies successfully the *Pareto Trade-off* boundary formed by the *baseline* models. The *Pareto Trade-off* is the point where improving one metric cannot be achieved without sacrificing another metric. In other words, this system is able to significantly improve recommendation accuracy without compromising diversity. This finding demonstrates that SRRP is not only superior in predicting relevance but also capable of maintaining user experience quality through a richer variety of bundles.

5.2 Dynamic Bundle Size Distribution Analysis

A comparative analysis of *bundle* size distributions aims to prove the hypothesis that SRRP is able to avoid static item restrictions and replicate users' natural shopping structures. Table 5 summarizes the size distributions generated by the dynamic reasoning validation module compared to the SOTA algorithm, which tends to be static.

Table 5: Dynamic Bundle Size Distribution in the SRRP Architecture

Item Size	RCO (Mean: 4.27)	RM (Mean: 4.39)	SS (Mean: 4.24)
1 item	0.19%	0.31%	0.24%
3 items	26.16%	25.69%	28.78%
4 items	29.44%	25.32%	26.50%
5 items	23.34%	28.13%	23.45%
> 6 items	Covered up to 45 items (Long-tail)		

The data shows that the majority of generated bundles fall within a reasonable range of 3 to 5 items. This range collectively covers approximately 75 to 79% of the total recommendation population. The frequency of single-item bundles is negligible (below 0.31%), verifying that the reasoning algorithm logically avoids isolated product recommendations. Most notable is the system's ability to precisely maintain long-tail distributions up to 45 items. This confirms that SRRP is robust in accommodating bulk purchasing behavior without being distorted by forced size fixations.

Compared to the baseline model in Figure 6, the SRRP architecture is shown to generate significantly more dynamic bundle sizes. This system is not tied to rigid criteria, is not biased towards minimum pair sizes, and is able to rationally adjust package sizes according to user spending preferences. The advantages of this distribution can be analyzed through three key findings:

First, conventional methods such as Apriori and FP-Growth have been shown to have limitations in forming *bundles* of more than 3 items. This limitation is due to the reliance of these statistical methods on very rigid thresholds (*support* and *confidence* values). As the shopping basket grows, the frequency of co-occurrence between items decreases drastically, so these data mining methods stop at an average size of 1.75 items per bundle.

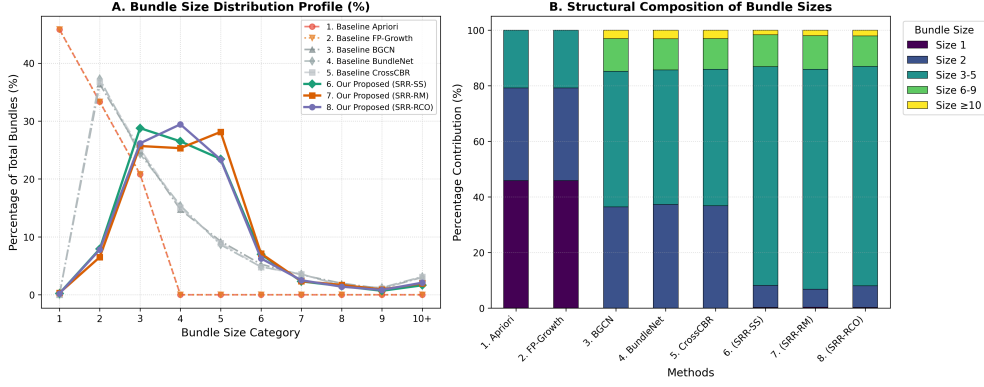


Fig. 6: Comparison of dynamic bundle sizes generated by SRRP against the *baseline* model

Second, Deep Learning and Graph Neural Network-based recommendation models (such as BundleNet, BGCN, and CrossCBR) tend to play it safe by heavily dominating recommendations for 2-item pairs. In these models, 2-item bundles account for a substantial portion of the total distribution, reaching 36.79 to 37.28% of the total. In contrast, our SRRP approach effectively suppresses this minimum pair bias to just 6.5 to 7.9%. As the dominance of 2-item bundles decreases, the system shifts its focus to semantically richer multi-item combinations, ultimately significantly increasing the average bundle size to 3 to 5 items.

Third, SRRP methods are the only family of methods that stably maintain the existence of recommendations across the entire spectrum of shopping sizes. The system is able to adaptively maintain recommendations of single-item *bundle* (of size 1) at a very small number (below 0.31%) to accommodate the absence of logical correlation, pair *bundle* (of size 2), concentrate 75 to 79% of the recommendation population in the ideal range (of size 3 to size 5), and maintain precise distribution of large-scale *bundle* in the range of 10 to 45 items. These achievements confirm that SRRP is highly robust in accommodating a wide spectrum of shopping behaviors, from small impulse purchases to bulk stockpiling, without being distorted by forced size fixation.

5.2.1 Analysis of the Quality of Explanatory Texts

In addition to prediction accuracy, this study evaluates the quality of the explanatory narratives generated by the Reasoning Validation Unit (RVU) to ensure the transparency of recommendations. This evaluation uses a quantitative linguistic metrics approach (Distinct-2 and Persuasion Index) and an LLM-as-a-Judge (G-Eval) approach.

Based on Table 6, the fluency score (*Fluency*) reached 4.79 on a 5.0 scale, indicating that the system is capable of producing sentences with natural grammar without any linguistic degradation. On the other hand, the *Persuasiveness* score of 3.00 indicates that the system’s narrative style is more like an informative and objective professional

Table 6: Evaluation Results of Linguistic Metrics and G-Eval on Explanatory Texts

Evaluation Metric	Average Score	Performance Interpretation
Word Count	25.65 words	Concise and efficient for user interfaces.
Lexical Diversity (<i>Distinct-2</i>)	0.1261	Dynamic phrase structure variation (not rigid).
Persuasion Index	4.03 words/text	16% of words convey value-add propositions.
G-Eval: Fluency (1-5)	4.79	Highly natural grammar equivalent to native speakers.
G-Eval: Coherence (1-5)	3.88	Rational and logical causal explanations.
G-Eval: Persuasiveness (1-5)	3.00	Professional and informative delivery style.

consultant, rather than a manipulative and pushy sales style (*hard-selling*). This concise and concise narrative quality successfully bridges the gap between the *black-box* algorithm and human rational understanding.

5.3 Human-LLM Alignment Analysis

To capture the qualitative nuances and subjectivity of *bundle* logic that are difficult to measure mathematically, alignment evaluation is performed by comparing the predictions of offline cognitive agents (*Gemma-3*, *LLaMA-3*, *Qwen-3*) with the *ground truth* annotation results of *Human-in-the-Loop* (HITL). This *alignment* dataset is built from the consensus of nine independent annotators measured by *Cosine Similarity* in semantic latent space. This measurement dissects alignment into two specific metrics: category name alignment (*Category Name*) and explanation argument alignment (*Explanation*). A summary of the alignment comparison results from the three LLM agents is presented in Table 7.

Table 7: Semantic Alignment Evaluation Results (Cosine Similarity) of LLM Agents against Annotator Consensus

Model Agent Cognitive	S_{align} (Category Name)	S_{align} (Explanation)	Overall Alignment	Test Sample
LLaMA-3	0.5233	0.6580	0.5907	65 <i>Bundle</i>
Qwen-3	0.5547	0.6154	0.5850	57 <i>Bundle</i>
Gemma-3	0.4555	0.6924	0.5740	80 <i>Bundle</i>

Based on Table 7, the test results show that all three models achieved significant positive alignment consistency with human reasoning. The LLaMA-3 model leads with the highest overall alignment score of 0.5907. This performance is driven by the model’s balance in naming categories and constructing explanations, which closely approximates the thinking style of human annotators.

The most interesting aspect of this data breakdown is the discovery of cognitive specialization within each model. The Qwen-3 agent proved to be the best at abstracting

package names (*Category Name*) with the highest alignment score of 0.5547, indicating its ability to extract concise, easily understood meanings. Conversely, the Google Gemma-3 agent dominated decisively in constructing explanatory arguments (*Explanation*) with the highest alignment score of 0.6924, although its performance declined slightly in the short naming aspect.

Overall, the Cosine Similarity values within the strongly positive range empirically validate that the compact LLM reasoning used in this architecture module aligns with human cognitive perception and logic, without falling prey to irrelevant textual hallucinations. This alignment represents a significant milestone in delivering a rational, credible, and trustworthy bundle recommendation system for real-world users.

While quantitatively, all three LLM cognitive agents demonstrated semantic alignment scores within the positive range (close to 0.6), a more in-depth qualitative analysis of the prediction data revealed anomalies and certain limitations in the machine’s reasoning. The most predominant cognitive error found among the model’s predictions was forced rationalization (over-justification). Since LLMs are architecturally trained to consistently provide informative output, the model often falls prey to attempts to find common threads or apparent correlations in a collection of shopping carts that are actually purely random (noise category).

To illustrate the successes and failures of this machine reasoning, Table 8 presents a direct comparison between the human annotator consensus (*Ground Truth*) and the LLM agent predictions (taking the example from the Gemma-3 model) on two different basket scenarios.

Table 8: Case Study: Comparison of Human Cognitive Reasoning vs. LLM Predictions

<i>Parameter</i>	Case 1: Perfect Alignment (True Positive)	Case 2: Forced Rationalization (False Positive)
<i>Shopping List Items</i>	Spinach, Quinoa, Bell Peppers, Greek Yogurt, etc.	Pasta, squash, spinach, milk, apples, kombucha, butter, etc.
<i>Human Ground Truth</i>	<i>Validity:</i> Yes <i>Category Name:</i> Healthy and Delicious Salad <i>Explanation:</i> The dressing, quinoa, vegetables, and lettuce complement each other to make a salad bowl.	<i>Validity:</i> No <i>Category Name:</i> Vegetable Lasagna Menu <i>Explanation:</i> Pasta, vegetables, milk, and butter do not form a consistent shopping theme.
<i>Machine Prediction (LLM)</i>	<i>Validity:</i> Yes <i>Category Name:</i> Healthy Bowl Ingredients <i>Explanation:</i> These items are commonly used together to make a healthy and nutritious quinoa salad.	<i>Validity:</i> Yes <i>Category Name:</i> Healthy Home Cooking <i>Explanation:</i> Pasta, squash, spinach, milk, apples, kombucha, and butter are often used together for cooking healthy meals at home.

Based on Table 8, in Case 1, the model successfully demonstrated highly precise alignment with human reasoning. Both the annotators and the LLM agent agreed that the collection of items formed a logical bundle. Interestingly, although the LLM used English (*Healthy Bowl Ingredients*) and the annotators used Indonesian (*Healthy*

and *Delicious Salad*), the semantic meaning of both names and explanatory arguments were equivalent. This confirms the model’s capability to accurately understand complementary relationships between items.

However, in Case 2, the LLM agent’s cognitive failure was clearly evident. The shopping cart in this case contained a mix of heavy food items (pasta), milk, vegetables, and even a fermented beverage (kombucha). The human annotator consensus firmly rejected this cart as a valid *bundle* (category "No"), arguing that the combination of products was too random, overlapping, and did not form a consistent culinary theme. Instead, the LLM agent enforces rationalization by labeling the basket as "Healthy Home Cooking" and justifying that the items are frequently used together.

This error in Case 2 demonstrates that small-parameter LLMs sometimes have too broad a level of abstraction or generalization (too loose). As long as the items within the basket fall under the same taxonomic umbrella—in this case, "food" or "consumables"—the model tends to treat them as a coherent bundle. The model ignores unusual real-world functional combinations (such as mixing kombucha with pasta and butter together) that are easily rejected by practical human reasoning.

This qualitative finding indicates that while the SRRP architecture is capable of linguistically fluent transparency and explanation, the probability threshold parameter on the Reasoning Validation Unit (RVU) still needs more rigorous calibration. Advanced prompt engineering that specifically asks the model to be more skeptical of baskets with high category dispersion will be crucial to suppressing future false positives.

6 Conclusion and Future Works

We introduce a Semantic-Retrieval-Reasoning Pipeline (SRRP) architecture, it’s a framework of hybrid Bundle Recommender System using LLM-based that are Prompting and Embedding. Evaluation of this research shows, SRRP outperforms baseline models, achieving a Recall up to 0.1502 and an MRR of 0.0683 in top-20 candidate searches without sacrificing diversity, as shown by a stable 0.5421 ILD score. Our system can capture 75 - 79% of recommended bundle sizes between 3 and 5 items, robustly accommodates bulk purchases up to 45 items, and reduces single-item recommendations to below 29.44%. Beyond predictive precision, integrating LLMs for reasoning extraction and validation resolves interpretability issues, generating fluent explanatory narratives with a 4.79 out of 5.0 G-Eval score. Finally, human evaluation with nine annotators confirms high cognitive alignment (0.5907 Cosine Similarity) without textual hallucinations, proving that the SRRP architecture delivers rational, credible, and trustworthy recommendations.

Although the proposed SRRP architecture demonstrates significant empirical benefits, targeted enhancements are essential to maximize its performance in production environments. First, latency and computational overhead must be reduced by implementing model quantization, asynchronous queuing, or knowledge distillation into compact neural architectures like GNNs to resolve the 3.41-second cold-start delay during real-time inference. Second, mitigating forced rationalization on unrelated items requires tighter calibration of the rejection threshold within the reasoning validation

module, potentially leveraged through RLHF to train a more skeptical model. Finally, transitioning from offline historical data to online A/B testing on live e-commerce platforms is crucial to measure actual real-world conversion impacts. Together, these steps will enhance the framework’s interactivity, logical rigor, and real-world applicability.

Acknowledgements. This research was funded by the Directorate of Research, Technology and Community Service (DRTPM) under the Dissertation Doctor Research Scheme, with contract number 131/C3/DT.05.00/PL-MULTITAHUN LANJUTAN. We express our gratitude for their support, which made this research possible.

Declarations

- **Funding:** This research was funded by the Directorate of Research, Technology and Community Service (DRTPM) under the Dissertation Doctor Research Scheme, contract number 131/C3/DT.05.00/PL-MULTITAHUN LANJUTAN. The authors gratefully acknowledge this support.
- **Conflict of interest:** The authors declare no competing interests or conflicts of interest.
- **Ethics approval and consent to participate:** Not applicable.
- **Consent for publication:** All authors have approved the manuscript and consented to its publication.
- **Data availability:** The data used in this study are available at <https://huggingface.co/datasets/recommender-system/bundle-instacart-dataset/tree/main>.
- **Materials availability:** Not applicable.
- **Code availability:** The source code is available at <https://huggingface.co/recommender-system/SRRP/tree/main>.
- **Author contributions:** **Andy Maulana Yusuf:** Modeling (LLM), literature review, and writing. **Adiwijaya** Conceptualization, validation, and proofreading. **Agung Toto Wibowo** Modeling (RS and GNN). **Z. K. A. Baizal** Experimental design and statistical testing.

References

- [1] Fang, Y., Wang, R., Guo, M., Hou, Y.: Product bundling for online supermarkets by frequent itemset mining and optimization approach. *Procedia Computer Science* **207**, 4434–4441 (2022) <https://doi.org/10.1016/j.procs.2022.09.507> . Knowledge-Based and Intelligent Information and Engineering Systems: Proceedings of the 26th International Conference KES2022
- [2] Jena, S.K., Ghadge, A.: Product bundling and advertising strategy for a duopoly supply chain: a power-balance perspective. *Annals of Operations Research* **315**(2), 1729–1753 (2022) <https://doi.org/10.1007/s10479-020-03861-9>
- [3] Sun, Z., Yang, J., Feng, K., Fang, H., Qu, X., Ong, Y.S.: Revisiting bundle recommendation: Datasets, tasks, challenges and opportunities for intent-aware

- product bundling. In: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval. SIGIR '22, pp. 2900–2911. Association for Computing Machinery, New York, NY, USA (2022). <https://doi.org/10.1145/3477495.3531904>
- [4] Banerjee, A., Mahmudov, T., Adler, E., Aisyah, F.N., Wörndl, W.: Modeling sustainable city trips: integrating CO₂e emissions, popularity, and seasonality into tourism recommender systems. *Information Technology & Tourism* **27**(1), 189–226 (2025) <https://doi.org/10.1007/s40558-024-00303-1>
 - [5] Cheng, X.: A travel route recommendation algorithm based on interest theme and distance matching. *EURASIP Journal on Advances in Signal Processing* **2021**(1), 57 (2021) <https://doi.org/10.1186/s13634-021-00759-x>
 - [6] Gasmi, I., Soui, M., Barhoumi, K., Abed, M.: Recommendation rules to personalize itineraries for tourists in an unfamiliar city. *Applied Soft Computing* **150**, 111084 (2024) <https://doi.org/10.1016/j.asoc.2023.111084>
 - [7] Feng, K., Sun, Z., Yang, J., Fang, H., Qu, X., Liu, W.: Does knowledge distillation matter for large language model-based bundle generation? *ACM Transactions on Information Systems* **44**(4) (2026) <https://doi.org/10.1145/3808223>
 - [8] Pereira, A.M., Moura, J.A.B., Costa, E.D.B., Vieira, T., Landim, A.R.D.B., Bazaki, E., Wanick, V.: Customer models for artificial intelligence-based decision support in fashion online retail supply chains. *Decision Support Systems* **158**, 113795 (2022) <https://doi.org/10.1016/j.dss.2022.113795>
 - [9] Klenitskiy, A., Vasilev, A.: Turning dress into gold loss: is bert4rec really better than sasrec? In: Proceedings of the 17th ACM Conference on Recommender Systems. RecSys '23, pp. 1120–1125. Association for Computing Machinery, New York, NY, USA (2023). <https://doi.org/10.1145/3604915.3610644>
 - [10] Liu, Z., Zhang, H., Chen, B., Jiang, Z., Zhao, Y., Tao, Y., Yang, T., Cui, B.: Cafe+: Towards compact, adaptive, and fast embedding for large-scale online recommendation models. *ACM Trans. Inf. Syst.* **43**(3), 1–42 (2025) <https://doi.org/10.1145/3713072>
 - [11] Chang, J., Gao, C., He, X., Jin, D., Li, Y.: Bundle recommendation and generation with graph neural networks. *IEEE Transactions on Knowledge and Data Engineering* **35**(3), 2326–2340 (2023) <https://doi.org/10.1109/TKDE.2021.3114586>
 - [12] Tao, Q., Liu, C., Xia, Y., Xu, Y., Li, L.: Adaptive multi-graph contrastive learning for bundle recommendation. *Neural Networks* **181**, 106832 (2025) <https://doi.org/10.1016/j.neunet.2024.106832>
 - [13] Sun, M., Li, L., Li, M., Tao, X., Zhang, D., Xie, Q., Wang, P., Huang, J.X.: A

- survey on bundle recommendation: Methods, applications, and challenges. *ACM Computing Surveys* **58**(11) (2026) <https://doi.org/10.1145/3802820>
- [14] Lubos, S., Tran, T.N.T., Felfernig, A., Polat Erdeniz, S., Le, V.-M.: Llm-generated explanations for recommender systems. In: *Adjunct Proceedings of the 32nd ACM Conference on User Modeling, Adaptation and Personalization. UMAP Adjunct '24*, pp. 276–285. Association for Computing Machinery, New York, NY, USA (2024). <https://doi.org/10.1145/3631700.3665185>
 - [15] Deldjoo, Y., He, Z., McAuley, J., Korikov, A., Sanner, S., Ramisa, A., Vidal, R., Sathiamoorthy, M., Kasrizadeh, A., Milano, S., Ricci, F.: Recommendation with generative models. *Foundations and Trends in Information Retrieval* **20**(1–2), 1–176 (2026) <https://doi.org/10.1108/FTINR-06-2025-0109>
 - [16] Yusuf, A.M., Adiwijaya, Wibowo, A.T., Baizal, Z.K.A.: Enhancing sequential recommendation with multi-intent detection and preference scoring from implicit transactions. *Ingénierie des Systèmes d'Information* **30**(12), 3093–3102 (2025) <https://doi.org/10.18280/isi.301202>
 - [17] Sun, Z., Feng, K., Yang, J., Fang, H., Qu, X., Ong, Y.-S., Liu, W.: Revisiting bundle recommendation for intent-aware product bundling. *ACM Transactions on Recommender Systems* **2**(3) (2024) <https://doi.org/10.1145/3652865>
 - [18] Jiang, Y., Yuan, K., Shang, J., Chen, X., Jiang, Y.: From sparsity to precision: Enhancing bundle recommendations via large language model augmentation and multi-graph fusion. *Decision Support Systems* **209**, 114732 (2026) <https://doi.org/10.1016/j.dss.2026.114732>
 - [19] Sun, Z., Feng, K., Yang, J., Qu, X., Fang, H., Ong, Y.-S., Liu, W.: Adaptive in-context learning with large language models for bundle generation. In: *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval. SIGIR '24*, pp. 966–976. Association for Computing Machinery, New York, NY, USA (2024). <https://doi.org/10.1145/3626772.3657808>
 - [20] Hu, J., Zhu, S., Cui, C., Shen, Q., Ji, Y., Wei, Z.: Text2bundle: Towards personalized query-based bundle generation. *ACM Transactions on Recommender Systems* **4**(2) (2025) <https://doi.org/10.1145/3706125>
 - [21] Kaban, E., Darmawiguna, I.G.M., Kesiman, M.W.A.: Optimizing customer purchase insights: Apriori algorithm for effective product bundle recommendations. *Brilliance: Research of Artificial Intelligence* **4**(2), 747–756 (2024) <https://doi.org/10.47709/brilliance.v4i2.4981>
 - [22] Wei, P., Liu, S., Yang, X., Wang, L., Zheng, B.: Towards personalized bundle creative generation with contrastive non-autoregressive decoding. In: *Proceedings of the 45th International ACM SIGIR Conference on Research and Development*

- in Information Retrieval. SIGIR '22, pp. 2634–2638. Association for Computing Machinery, New York, NY, USA (2022). <https://doi.org/10.1145/3477495.3531909>
- [23] Wang, Z.: Empowering few-shot recommender systems with large language models-enhanced representations. *IEEE Access* **12**, 29144–29153 (2024) <https://doi.org/10.1109/ACCESS.2024.3368027>
 - [24] He, Z., Xie, Z., Jha, R., Steck, H., Liang, D., Feng, Y., Majumder, B.P., Kallus, N., McAuley, J.: Large language models as zero-shot conversational recommenders. In: *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management. CIKM '23*, pp. 720–730. Association for Computing Machinery, New York, NY, USA (2023). <https://doi.org/10.1145/3583780.3614949>
 - [25] Li, L., Zhang, Y., Chen, L.: Prompt distillation for efficient llm-based recommendation. In: *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management. CIKM '23*, pp. 1348–1357. Association for Computing Machinery, New York, NY, USA (2023). <https://doi.org/10.1145/3583780.3615017>
 - [26] Park, J., Kim, S., Lee, S.: A user preference and intent extraction framework for explainable conversational recommender systems. In: *Companion Proceedings of the 2023 ACM SIGCHI Symposium on Engineering Interactive Computing Systems. EICS '23 Companion*, pp. 16–23. Association for Computing Machinery, New York, NY, USA (2023). <https://doi.org/10.1145/3596454.3597178>
 - [27] Ostendorff, M., Blume, T., Ruas, T., Gipp, B., Rehm, G.: Specialized document embeddings for aspect-based similarity of research papers. In: *Proceedings of the 22nd ACM/IEEE Joint Conference on Digital Libraries. JCDL '22*, pp. 7–1712. Association for Computing Machinery, New York, NY, USA (2022). <https://doi.org/10.1145/3529372.3530912>
 - [28] Zeng, H., Kallumadi, S., Alibadi, Z., Nogueira, R., Zamani, H.: A personalized dense retrieval framework for unified information access. In: *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval. SIGIR '23*, pp. 121–130. Association for Computing Machinery, New York, NY, USA (2023). <https://doi.org/10.1145/3539618.3591626>
 - [29] Wei, C., Duan, K., Zhuo, S., Wang, H., Huang, S., Liu, J.: Enhanced recommendation systems with retrieval-augmented large language model. *Journal of Artificial Intelligence Research* **82** (2025) <https://doi.org/10.1613/jair.1.17809>
 - [30] Luo, S., Xu, J., Zhang, X., Wang, L., Liu, S., Hou, H., Song, L.: Rallrec+: Retrieval augmented large language model recommendation with reasoning. *Expert Systems with Applications* **297**, 129508 (2026) <https://doi.org/10.1016/>

- [31] Aly, A., Ibrahim, A., Kashef, R.: Rag-rec: Retrieval augmented generation for robust personalized recommendation systems. In: Proceedings of the 7th International Conference on AI in Computational Linguistics, vol. 275, pp. 1–8 (2026). <https://doi.org/10.1016/j.procs.2026.01.002>
- [32] Hachicha, M., Omezine, M.A., Zghal, M.K., Riahi, M.H., Ncib, L.: Recommendation module based on web scraping and llm models with the rag technique. In: Proceedings of the 2024 Artificial Intelligence Revolutions (AIR). AIR '24, pp. 68–73. IEEE, ??? (2024). <https://doi.org/10.1109/AIR63653.2024.00025>
- [33] Juni Permana, A.H., Wibowo, A.T.: Movie recommendation system based on synopsis using content-based filtering with tf-idf and cosine similarity. International Journal on Information and Communication Technology (IJoICT) **9**(2), 1–14 (2023) <https://doi.org/10.21108/ijoiect.v9i2.747>
- [34] Ren, Y., Zhang, H., Fu, L., Wang, X., Zhou, C.: Distillation-enhanced graph masked autoencoders for bundle recommendation. In: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval. SIGIR '23, pp. 1660–1669. Association for Computing Machinery, New York, NY, USA (2023). <https://doi.org/10.1145/3539618.3591666>
- [35] Ma, Y., He, Y., Zhang, A., Wang, X., Chua, T.-S.: Crosscbr: Cross-view contrastive learning for bundle recommendation. In: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. KDD '22, pp. 1233–1241. Association for Computing Machinery, New York, NY, USA (2022). <https://doi.org/10.1145/3534678.3539229>
- [36] Kumar, B., Raj, S., Anand, M., Verma, V.K., Tripathi, P., Jha, A.: Enhancing conversational recommendations through contrastive user preference modeling. Discover Artificial Intelligence **6**(1), 543 (2026) <https://doi.org/10.1007/s44163-026-00912-1>
- [37] Ouyang, S., Lawlor, A.: Improving explainable recommendations by deep review-based explanations. IEEE Access **9**, 67444–67455 (2021) <https://doi.org/10.1109/ACCESS.2021.3076146>
- [38] Deldjoo, Y., He, Z., McAuley, J., Korikov, A., Sanner, S., Ramisa, A., Vidal, R., Sathiamoorthy, M., Kasirzadeh, A., Milano, S.: A review of modern recommender systems using generative models (gen-recsys). In: Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. KDD '24, pp. 6448–6458. Association for Computing Machinery, New York, NY, USA (2024). <https://doi.org/10.1145/3637528.3671474>
- [39] Yusuf, A.M., Musrinah, M.: Integrating llm intent-aware approach for enhancing the quality of bundle recommendations. Computer Science (CO-SCIENCE) **6**(2),

144–153 (2026) <https://doi.org/10.31294/co-science.v6i2.12101>

- [40] He, Z., Zhao, H., Yu, T., Kim, S., Du, F., McAuley, J.: Bundle mcr: Towards conversational bundle recommendation. In: Proceedings of the 16th ACM Conference on Recommender Systems. RecSys '22, pp. 288–298. Association for Computing Machinery, New York, NY, USA (2022). <https://doi.org/10.1145/3523227.3546755>
- [41] Yue, W., Yin, Y., Zhang, X., Shi, B., Liang, T., Wan, J.: Cot4rec: Revealing user preferences through chain of thought for recommender systems. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 39, pp. 13142–13151 (2025). <https://doi.org/10.1609/aaai.v39i12.33434>
- [42] Yang, F., Yue, Y., Li, G., Payne, T.R., Man, K.L.: Chain-of-thought prompting empowered generative user modeling for personalized recommendation. *Neural Computing and Applications* **36**(34), 21723–21742 (2024) <https://doi.org/10.1007/s00521-024-10364-2>
- [43] Saeed, T., Wang, B.: Large language models for generative recommendation: a systematic review of data-centric taxonomy, evaluation, and human-centric analytics. *International Journal of Data Science and Analytics* **22**(1), 130 (2026) <https://doi.org/10.1007/s41060-026-01106-9>
- [44] Jiang, K., Pan, A., Jiang, Y., Cheng, S.: Dynamic intent-aware and cross-session integration for session-based recommendation. *Discover Artificial Intelligence* **5**(1), 399 (2025) <https://doi.org/10.1007/s44163-025-00650-w>
- [45] Alam, M., Iana, A., Grote, A., Ludwig, K., Müller, P., Paulheim, H.: Towards analyzing the bias of news recommender systems using sentiment and stance detection. In: Companion Proceedings of the Web Conference 2022. WWW '22, pp. 448–457. Association for Computing Machinery, New York, NY, USA (2022). <https://doi.org/10.1145/3487553.3524674>