Boosting E-commerce Content Diversity: A Graph-based RAG Approach with User Reviews

Jiaxi Yang*
College of Information Sciences and Technology
The Pennsylvania State University
University Park, PA, USA
jmy5701@psu.edu

Yiling Jia*
Google DeepMind
Mountain View, CA, USA
yilingjia@google.com

Carl Yang†
Department of Computer Science
Emory University
Atlanta, GA, USA
j.carlyang@emory.edu

Yi Liang
Google DeepMind
New York, NY, USA
yiliang@google.com

Lu Lin
College of Information Sciences and Technology
The Pennsylvania State University
University Park, PA, USA
lulin@psu.edu

Abstract

In e-commerce, product descriptions and other forms of copywriting play a critical role in shaping consumer purchasing decisions. However, manually crafting such content is both time-consuming and costly, particularly given the vast and diverse item catalogs. Recent advances in large language models (LLMs) have transformed automated text generation, offering immense potential to streamline this process. Despite their capabilities, LLMs continue to face obstacles in e-commerce applications, including a lack of diversity and an inability to fully grasp the nuanced details of specific items. To address these limitations, we propose a novel framework that integrates graph-based knowledge into Retrieval-Augmented Generation (RAG) to enhance content generation. Our approach leverages user reviews to construct an item-feature graph, capturing both explicit and implicit connections between items and features. This structured representation enables the retrieval of diverse, contextually relevant, and factually grounded information, effectively addressing key deficiencies of existing methods. With the constructed graph, we design a graph traversal mechanism that explores a broader range of item-related features, augmenting the generation process with more varied and informative inputs. Extensive experiments demonstrate that our method significantly improves diversity while preserving fidelity, marking a major advancement in automated e-commerce content generation.

Keywords

copywriting generation; e-commerce; large language models; retrieval-augmented generation; graph traversal

*Both authors contributed equally to this research.
†Work done while being a visiting faculty in Google.

KDD '25, Toronto, ON, Canada

ACM ISBN 979-8-4007-1454-2/2025/08
https://doi.org/10.1145/3711896.3736864

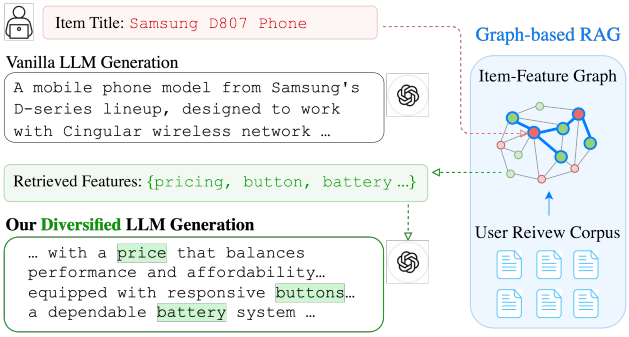

Figure 1: E-commerce content generated by our method incorporates more features traversed from the graph, leading to greater diversity.

ACM Reference Format:
Jiaxi Yang, Yiling Jia, Carl Yang, Yi Liang, and Lu Lin. 2025. Boosting E-commerce Content Diversity: A Graph-based RAG Approach with User Reviews. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2 (KDD '25), August 3–7, 2025, Toronto, ON, Canada.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3711896.3736864

1 Introduction

In e-Commerce, well-crafted content is essential for delivering item information and supporting informed decision-making. Traditionally, this content has been created manually, a time-consuimg and resource-intensive process, particularly as modern e-commerce platforms manage vast and ever-expanding item catalogs [12]. With millions of items across diverse categories, relying solely on manual efforts is unsustainable, making automated content generation a critical area of research [15, 31, 33].

Recent advances in large language models (LLMs) have revolutionized automated text generation, offering the potential to scale the creation of item descriptions while maintaining contextual relevance [24]. However, directly using LLMs to e-commerce content generation presents significant challenges. One major issue is the

lack of *diversity* in generated content, which often results in repetitive, overly generic descriptions that fail to capture the multifaceted nature of the items. Additionally, LLMs typically rely on pre-learned internal knowledge, limiting their ability to incorporate dynamic or domain-specific information, such as evolving item attributes or user feedback.

A promising solution is Retrieval-Augmented Generation [2, 9, 10, 17], which augments LLMs with external knowledge sources to improve both relevance and factual grounding. In e-commerce, user reviews serve as a valuable external knowledge source, providing diverse, authentic insights into item features. These reviews often capture important details, such as item strengths, weaknesses, and situational use cases, that are essential for creating rich and informative content. However, directly applying RAG to unstructured reviews data is challenging due to the absence of an explicit representation of relationships between products and their features, which can lead to suboptimal retrieval and limited diversity in generated content.

In this paper, we propose a novel graph-based RAG framework that transforms unstructured user reviews into structured item-feature graph, enabling more effective retrieval and generation of diverse e-commerce content generation. Specifically, we extract the relationship between item and their features by analyzing user reviews, *e.g.,* how frequently specific features are mentioned in relation to each product. For example, if multiple reviews frequently mention features such as "battery" or "display quality" in association with an item, these associations are encoded as connections within the graph. Additionally, we capture the co-occurrence of features within reviews to model correlations between them.

This graph enables the retrieval process to provide more diverse, contextually relevant, and factually grounded inputs to the LLMs. Furthermore, we introduce a graph traversal mechanism that dynamically explores a wide range of item-related features during retrieval. This approach ensures that the generated content highlights multiple perspectives and nuanced details of items, improving user engagement and the overall comprehensiveness of descriptions. Extensive experiments demonstrate that our method significantly improves the diversity and informativeness of generated content while maintaining high levels of factual accuracy and coherence.

The contributions of our paper are as follows:

- We propose a graph-based RAG framework that captures both explicit and implicit relations between items and their feature, enhancing the diversity for automating e-commerce content generation.
- To further improve diversity, we design a graph traversal mechanism that dynamically explores a broad set of item-related features, enriching the input for LLM-based content generation while preserving factual accuracy.
- We conduct extensive experiments demonstrating that our method significantly improves the diversity and informativeness of item description generation while maintaining high faithfulness and coherence.

2 Related Work

2.1 E-commerce Content Generation

Early studies focused on generating high-quality content in e-commerce and improving customer experience by utilizing natural language generation frameworks that combined statistical approaches with manually crafted structures [29]. With the rise of neural networks, researchers introduced various deep learning frameworks to harness their computational power and generate e-commerce content directly from data [33, 34]. As neural network-based methods advanced, researchers began to explore personalization as a key focus, with many studies leveraging user feedback, such as clicks or purchase history, to tailor e-commerce content that align closely with individual preferences [7, 19, 31]. A gated pointer-generator transformer has been proposed, integrating user attributes and product features through a select-attention mechanism and a copy mechanism to generate personalized e-commerce content with high faithfulness and quality [19]. Similarly, the Personalized Answer Generation (PAGE) method leverages historical user-generated content for multi-perspective preference modeling, combining knowledge-level, aspect-level, and vocabulary-level personalization [7]. Wang *et al.,* [31] propose a reinforcement learning approach with attention-based neural networks to align generated descriptions closely with user click patterns, enhancing both personalization and relevance. Rather than personalization, ensuring the faithfulness of generated e-commerce content has also been a key focus in some studies [15]. Chan *et al.,* [3] propose a fidelity-oriented approach to product description generation that incorporates entity-label-guided LSTMs and a keyword memory, explicitly aligning the generated content with product attributes to improve faithfulness. Guo *et al.,* [11] introduce a prefix-based controllable product copywriting framework that ensures generated descriptions faithfully align with product characteristics. ModICT [18] ensures faithfulness by leveraging multimodal in-context references, integrating visual features and marketing keywords to align generated descriptions with product-specific attributes.

Although personalization and faithfulness are essential, the works mentioned above are orthogonal to ours, which focus on enhancing diversity in e-commerce content generation. The limited existing studies such as [25] attempt to balance diversity and faithfulness by introducing controllable generative models named Apex. Besides, by utilizing graph attention mechanims for product-related knowledge retrieval and combing with individual information, DeepDepict [12] achieves diversity and personalization enhancement. Similarly, KOBE [4] achieves personalization and diversity by fusing user clicks and product attributes with external knowledge using bidirectional attention mechanisms. However, their performance is limited by the effectiveness of information extraction and text generation capabilities. Moreover, they highly depend on user-specific information, such as user click data, while our approach releases this and is more available or practical in scenarios with sparse user data or privacy restrictions.

2.2 Retrieval-Augmented Generation (RAG)

RAG-based LLMs have emerged as a promising solution to mitigate hallucination by grounding outputs in factual data retrieved from external knowledge sources [9]. This approach enhances the reliability and credibility of text generation by aligning generated

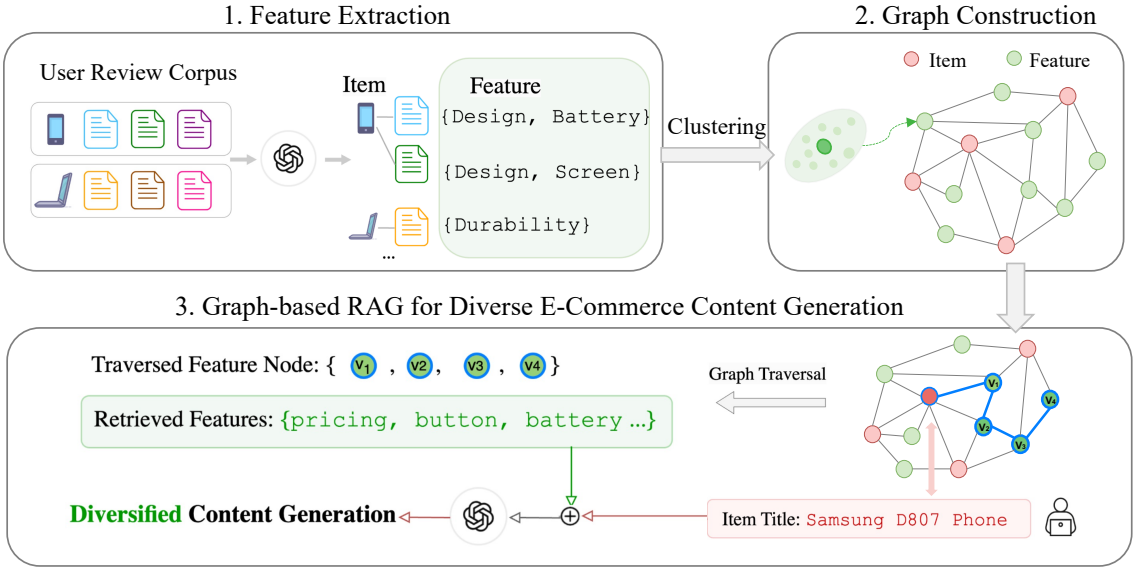

Figure 2: Framework of our method including feature extraction (Sec. 3.1), item feature graph construction (Sec. 3.2), graph exploration (Sec. 3.3), and e-commerce generation generation (Sec. 3.4).

content with accurate and up-to-date information. To further enhance the reasoning capability, more recent studies attempt to incorporate Knowledge Graph (KGs) to help improve LLM reasoning [13, 26]. For instance, GNN-RAG [23] integrates Graph Neural Networks (GNNs) into RAG to retrieve multi-hop reasoning paths from KGs, improving the reasoning ability of LLMs while grounding their outputs in factual data. G-Retriever [13] mitigates hallucination by employing a Prize-Collecting Steiner Tree optimization to retrieve relevant subgraphs, enabling scalable and efficient reasoning for textual graph tasks while ensuring factual grounding. Similarly, RoG [22] uses LLM-based retrievers to generate plausible relation paths for KG retrieval, ensuring accurate knowledge extraction for reasoning tasks. The Knowledge-Driven Chain-of-Thought (KD-CoT) framework [30] further extends this by incorporating a retriever-reader-verifier pipeline that interacts with external knowledge, enabling LLMs to generate faithful reasoning steps and effectively address multi-hop knowledge-intensive tasks. However, these approaches fail to address the need for diversity, making them unsuitable for tasks like product description generation that require varied and engaging outputs.

3 Methodology

Reviews are an essential source for feature extraction, as they capture user-generated insights that reflect diverse and real-world experiences with the e-commerce item. These insights often highlight a wide range of attributes, including item qualities, performance, and limitations, providing a rich and varied resource for identifying key features. To leverage this, we first extract features from user reviews using an LLM and retain only the representative ones as cluster medoids to reduce feature redundancy. Building on this, we construct a heterogeneous graph that captures the relationship between items and features by their co-occurrence in user reviews, such that diverse feature aspects for each item can be accessible by traversing the graph. Subsequently, by exploring the graph to retrieve relevant features for RAG, we enable LLMs to generate e-commerce content that is both diverse and grounded in factual item attributes. The overview of our framework is shown in Figure 2.

3.1 Feature Extraction

User reviews provide valuable insights into item attributes, including quality, functionality, etc. We use LLMs to extract key attributes from user reviews, with the prompt shown in Figure 3. Specifically, given the user review corpus for a set of items $V = \{v_1, \ldots, v_{|V|}\}$, each item v_i is associated with reviews $R_i = \{r_{i,1}, \ldots, r_{i,|R_i|}\}$. With LLM-based extraction, each review $r_{i,j}$ is mapped to a subset of features $S_{i,j} \subset S$, where S denotes the complete feature set extracted from the corpus. This naturally forms an item-feature graph linking items v_i to their associated features in $S_{i,j}$.

Two key challenges arise with raw feature extraction: (1) the large volume of reviews generates an extensive feature set $|S|$, leading to a complex graph, and (2) semantically similar or redundant features inflate the graph unnecessarily, reducing its efficiency. To address these challenges, we apply a clustering approach to group similar features, eliminate redundancy, and maintain a more concise, efficient graph representation. Specifically, we encode each

Input: [Reviews of the item].
Output: [Review Features].

Target: Please extract all the item features mentioned in the following user review, generalizing specific mentions to broader categories where appropriate.

Example:
Input: I love this laptop's sleek design and long battery life. However, … as I'd like.
Output: Design, Battery life, Screen brightness, …

Figure 3: Prompt of Feature Extraction.

feature in a low-dimensional embedding space and cluster them into groups. The cluster medoid serves as the most representative feature for each feature group. This step results in a set of feature medoids $U = \{u_1, u_2, ..., u_{|U|}\}$, and each feature medoid u_k is the center feature of a feature cluster $C_k \subset S$. The size of U is much smaller than S, which offers a more practical construction of graph as detailed in the next section.

3.2 Graph Construction

In the heterogeneous graph $G = \{V, U, A\}$, where the node set V consists of item nodes, the node set U consists of feature medoids, and A denotes the adjacency matrix, which consists of three types of edges: 1) edges between feature medoids $e_{u\leftrightarrow u}$; 2) edges between items $e_{v\leftrightarrow v}$; and 3) edges between feature medoids and items $e_{u\leftrightarrow v}$. We attempt to calculate edge weights to capture the fine-grained relational information between these nodes.

3.2.1 ***Edges between feature medoids.*** Edges between two feature medoid nodes capture the co-occurrence of these two representative features in user reviews, reflecting their relevancy. Consider two feature medoids, u_i and u_j, which are associated with feature clusters C_i and C_j respectively. Each cluster represents a group of features derived from the feature extraction and clustering process. When calculating the edge weight between nodes u_i and u_j, we consider features in their associated clusters. Suppose cluster C_i contains features s_1, and cluster C_j contains features s_2. If features s_1 and s_2 co-occur in n user reviews, then the edge weight contributed by this feature pair is defined as n. The edge weight between u_i and u_j is then calculated as the sum of contributions from all feature pairs across the two clusters:

$$w(u_i, u_j) = \sum\nolimits_{s_1 \in C_i, s_2 \in C_j} n(s_1, s_2), \tag{1}$$

where $n(s_1, s_2)$ is the number of reviews in which feature s_1 and s_2 appear together.

3.2.2 ***Edge between feature medoid and item.*** Edge between feature medoid node u_i and item node v_j captures the importance of this feature aspect to describe the item. Consider an item v_j with reviews R_j. Suppose $m(R_j, s_k)$ denotes the number of reviews in R_j mentioning a feature $s_k \in S$. The edge weight between the feature medoid u_i and the item v_j is calculated as the total number of reviews mentioning any feature in the cluster C_i:

$$w(u_i, v_j) = \sum\nolimits_{s_k \in C_i} m(R_j, s_k). \tag{2}$$

This weight reflects the relevance of the feature medoid u_i to the item v_j based on the feature's appearance in this item's reviews.

3.2.3 ***Edge between item and item.*** Edges between item nodes represent their similarity based on shared features. The edge weight between two items, v_i and v_j, is determined by the number of features they are both associated with. Let $u_p \in \mathcal{N}(v_i)$ denotes the neighboring feature medoids of item node v_i. The item-item edge weight is given by:

$$w(v_i, v_j) = |(\cup_{u_p \in \mathcal{N}(v_i)} C_p) \cap (\cup_{u_q \in \mathcal{N}(v_j)} C_q)|, \tag{3}$$

which captures the total number of features shared by the two items. A higher weight indicates a stronger similarity between the two items in terms of their associated clusters.

3.2.4 ***Edge Weight Normalization.*** To ensure consistency and comparability across edges in the graph, we normalize edge weights so that the total contribution of all edges connected to a node of the same edge type is scaled proportionally. This normalization ensures that the relative importance of edges is preserved, balancing contributions from different nodes and preventing any node from dominating the graph's structure. For a feature medoids u_i, the weights of all edges $e_{u\to u}$ connecting u_i to other feature medoid nodes u_j are normalized as follows:

$$w_{\text{norm}}(u_i, u_j) = \frac{w(u_i, u_j)}{\sum_{u_k \in \mathcal{N}(u_i)} w(u_i, u_k)}, \tag{4}$$

in which $\mathcal{N}(u_i)$ denotes the neighbors of feature medoids u_i . Similarly, the weight of edges between feature medoids and items $e_{u\to v}$ and item-to-item edges $e_{v\to v}$ are presented in equation (5) and (6) respectively.

$$w_{\text{norm}}(u_i, v_j) = \frac{w(u_i, v_j)}{\sum_{v_k \in \mathcal{N}(u_i)} w(u_i, v_k)}. \tag{5}$$

$$w_{\text{norm}}(v_i, v_j) = \frac{w(v_i, v_j)}{\sum_{v_k \in \mathcal{N}(v_i)} w(v_i, v_k)}. \tag{6}$$

3.3 Graph-based Feature Retrieval

Similar items are more likely to share overlapping features, making them an intuitive and effective starting point for traversal. Starting from the most similar item node allows the exploration to efficiently leverage existing feature relationships that are both contextually relevant to the test item and conducive to enhancing diversity. Following this principle, for a given test item, we first identify its most similar item node in the constructed graph G by comparing their item titles (*e.g.*, comparing Sentence-BERT similarity embeddings) and then we perform a K-step traversal to systematically explore the graph for feature retrieval.

3.3.1 ***Objective of Feature Retrieval.*** We utilize U_{visit} to denote the visited feature medoids set, which can be denoted as follows:

$$U_{\text{visit}} = \{u_{\text{visit},1}, u_{\text{visit},2}, ..., u_{\text{visit},k}\} \subseteq U, \tag{7}$$

where U denotes the coalition of all feature medoid nodes. And we use $H(\mathcal{V}_{\text{visit}})$ to represent the internal diversity of the visited feature coalition in equation (8):

$$H(U_{\text{visit}}) = \binom{|U_{\text{visit}}|}{2} \cdot \sum\nolimits_{u_i, u_j \in U_{\text{visit}}} dist(E(u_i), E(u_j)), \tag{8}$$

in which $E(\cdot)$ and $dist(\cdot, \cdot)$ denote a feature encoder and a distance measurement separately. To enhance diversity, our initial goal is to maximize the internal diversity within the visited feature medoid coalition during graph traversal as shown in Eq. (9):

$$\max_{U_{\text{visit}}} H(U_{\text{visit}}) = \max_{U_{\text{visit}}} \binom{|U_{\text{visit}}|}{2} \cdot \sum\nolimits_{u_i, u_j \in U_{\text{visit}}} dist(E(u_i), E(u_j)). \tag{9}$$

However, solely focusing on diversity during traversal may lead to irrelevant features, compromising the factual grounding of the generated e-commerce content. To balance diversity and relevance during the traversal, it is essential to incorporate edge weights into the objective to account for relevance. Therefore, the problem objective can be reformulated as:

$$\max_{U_{\text{visit}}} \quad \alpha \cdot H(U_{\text{visit}}) + (1 - \alpha) \cdot \sum\nolimits_{u_i, u_j \in U_{\text{visit}}} w(u_i, u_j), \tag{10}$$

in which α is a weighting factor with $0 \le \alpha \le 1$.

Algorithm 1 Graph-based RAG for Diversified E-commerce Content Generation

Input: Review set $\mathcal{R}$, test item v_{target}, number of retrieved features T

Output: E-commerce content of item v_{target}

Step 1: (Pre-computed) Graph Construction
Initialize feature set $S = \emptyset$
for r in $\mathcal{R}$ **do**
  Extract features S_r from r by LLMs.
  $S \leftarrow S \cup S_r$.
end for
Obtain feature medoid set U by clustering features S.
Add item nodes V and feature medoids U to graph $G = (V, U)$.
Edge weights calculation for $e_{u\leftrightarrow u}, e_{u\leftrightarrow v}, e_{v\leftrightarrow v}$ by Eq. (1), Eq. (2) and Eq. (3) with normalization.

Step 2: Content Generation by Graph-based RAG
Retrieve the most similar item node v_j to v_{target}.
Initialization:
  Collect the set of neighbors $N(v_j)$
  Compute $f_{v_j \to v_k}$ in Eq.(12) for all $k \in N(v_j)$
  Select the top K features with the highest $f_{v_j \to v_k}$ values to form set U_0
for $t = 1$ to T **do**
  Initialize the set of candidate paths $U_t = \emptyset$
  for each node $u \in U_{t-1}$ **do**
    Collect the set of neighbors $\mathcal{N}(u)$
    for each $m \in N(u)$ **do**
      Compute f in Eq.(12)
      Add $(u, m, f(u, m))$ to U_t
    end for
  end for
  Retain the top K tuples by f values in U_t.
end for
$U_{\text{visit}} = U_T$.
Generate e-commerce content by LLMs with U_{visit}.

Input: [Item Title, Item Details, Features].
Output: [Item Description].

Target: Generate a concise, engaging, and informative item description using the given title, details, and features. Highlight key features without adding unprovided information.

Example:
Input:
Item Title: *Spigen Neo Hybrid Carbon Galaxy S6…*
Item Details: *Product Dimensions: 3.9 x 0.9 x 8…*
Features: *Durability, Durability…*
Output: Elevate your Galaxy S6 Edge Plus…

Figure 4: Prompt of e-Commerce Content Generation.

3.3.2 ***Beam Search Solution***. Finding the global optimum to achieve the objective in Eq. (10) is prohibitively expensive, with a computational complexity of $O((|V| + |U|)^K)$ with K denoting the traversal step. To address this combinatorial problem, we propose an approximate solution leveraging beam search. Specifically, for each traversal step, when moving to the next neighboring node from the current node, we first leverage Leave-One-Out (LOO) [6] principle to measure the marginal diversity increase ΔH_{u_j}, when a new feature medoid u_i is visited, as shown in Equation (11):

$$\Delta H_{u_j} = H(U_{\text{visit}} \cup \{u_j\}) - H(U_{\text{visit}}). \tag{11}$$

To balance diversity and relevance during the traversal, the scoring function for determining the next hop in the graph also considers the edge weights. Therefore, the overall score function of moving from node u_i to node u_j can be presented as follows:

$$f_{u_i \to u_j} = \alpha \cdot \Delta H_{u_j} + (1 - \alpha) \cdot w(u_i, u_j). \tag{12}$$

Beam search systematically explores promising paths by expanding the current set of nodes to include their neighbors and retaining the top-K candidates based on their scores, evaluated using $f_{u_i \to u_j}$. At each step, irrelevant or low-scoring paths are pruned, ensuring the traversal focuses on exploring diverse and relevant features while maintaining computational efficiency.

3.4 Diversified Content Generation by LLMs

After obtaining the item-related features through graph traversal, the next step involves leveraging LLMs to generate detailed and engaging e-commerce content. These features, extracted from the graph, provide additional contextual and factual information that serves as a foundation for the generation process. By incorporating this extra information, the LLM can produce descriptions that are not only diverse and reflective of different aspects of the item but also maintain high standards of factual reliability and relevance. The prompts for e-commerce content generation are shown in Figure 4.

4 Experiments

We conduct the experiments to answer the following **R**esearch **Q**uestions (**RQ**):

- **RQ1:** Does the proposed approach effectively enhance the diversity of the generated e-commerce content?
- **RQ2:** How effectively does the generated e-commerce content balance enhanced diversity with faithfulness?
- **RQ3:** Does the proposed approach ensure textual coherence in the generated e-commerce content?
- **RQ4:** Can the proposed approach maintain robust performance when tested on datasets from domains different from those used in the graph construction?
- **RQ5:** How sensitive is the proposed approach to changes in hyperparameter configurations?

4.1 Experiment Settings

4.1.1 ***Datasets.*** We used the Amazon Reviews 2023 dataset [14] and Airbnb dataset [1], both of which include item titles, user reviews, and item descriptions. Specifically, we selected the "cell phone and accessories" subset from Amazon 2023 dataset to ensure a focused and representative evaluation of our approach.

To ensure the dataset is comprehensive and reliable with minimal noise, we filtered the metadata to retain items with descriptions exceeding 100 characters, ratings with more than 50 user reviews,

[1] https://insideairbnb.com/get-the-data/

Models	Metrics	Amazon					Airbnb				
		Ours	RAG (w)	RAG (w/o)	Transformer	Bart	Ours	RAG (w)	RAG (w/o)	Transformer	Bart
ChatGPT-4o	LLM	**69.11 ± 6.28**	60.67 ± 8.97	51.92 ± 11.26	21.37 ± 8.56	21.56 ± 11.11	**76.24 ± 5.37**	62.28 ± 7.81	59.73 ± 10.55	17.81 ± 10.72	18.63 ± 11.45
	SS	**0.55 ± 0.04**	0.49 ± 0.047	0.48 ± 0.07	0.33 ± 0.06	0.35 ± 0.13	**0.54 ± 0.03**	0.48 ± 0.04	0.42 ± 0.04	0.24 ± 0.06	0.24 ± 0.06
	IE	**7.25 ± 0.17**	6.41 ± 0.29	5.17 ± 0.45	4.17 ± 0.73	4.18 ± 0.79	**7.21 ± 0.20**	5.90 ± 0.32	4.34 ± 0.31	1.43 ± 0.92	1.44 ± 0.94
Deepseek-V3	LLM	**65.42 ± 7.14**	46.37 ± 9.66	35.03 ± 9.85	21.91 ± 9.36	22.16 ± 10.25	**74.85 ± 4.37**	59.33 ± 8.31	55.86 ± 11.15	17.81 ± 10.72	18.63 ± 11.45
	SS	**0.54±0.05**	0.46±0.07	0.47±0.09	0.33±0.06	0.35±0.13	**0.53 ± 0.03**	0.46 ± 0.02	0.41 ± 0.03	0.24 ± 0.06	0.24 ± 0.06
	IE	**6.87 ±0.21**	5.17 ± 0.41	4.61 ± 0.49	4.17 ± 0.73	4.18 ± 0.79	**7.12±0.14**	5.33 ± 0.30	4.33±0.34	1.43±0.92	1.44±0.94
Gemini	LLM	**66.0 ± 7.04**	47.01 ± 11.05	35.71 ± 10.28	20.65 ± 8.73	23.19 ± 9.85	**70.17 ±5.05**	47.66 ±11.59	27.84±13.10	17.81 ± 10.72	18.63 ± 11.45
	SS	**0.54 ± 0.05**	0.49 ± 0.08	0.49 ± 0.09	0.33 ± 0.06	0.35 ± 0.13	**0.51±0.03**	0.43±0.03	0.33±0.08	0.24±0.06	0.24±0.06
	IE	**6.96 ± 0.18**	4.94 ± 0.42	4.54 ± 0.47	4.17±0.73	4.18±0.79	**6.97±0.18**	4.17±0.42	2.47±1.01	1.43 ± 0.92	1.44 ± 0.94
Claude	LLM	**66.65 ± 7.92**	54.12 ± 10.06	46.55 ± 10.44	20.78 ± 9.01	19.32 ± 8.68	**73.99±5.75**	58.12±9.27	57.28±12.70	17.81 ± 10.72	18.63 ± 11.45
	SS	**0.53 ± 0.05**	0.48 ± 0.07	0.50 ± 0.10	0.33 ± 0.06	0.35 ± 0.13	**0.53±0.04**	0.46±0.03	0.42±0.04	0.24 ± 0.06	0.24 ± 0.06
	IE	**7.0 ± 0.11**	5.43 ± 0.33	4.91 ± 0.34	4.17 ± 0.73	4.18 ± 0.79	**6.97±0.12**	5.08±0.27	4.24±0.33	1.43 ± 0.92	1.44 ± 0.94
Llama-3.2-8B	LLM	**66.68 ± 6.15**	46.19 ± 11.12	45.68 ± 11.34	22.02 ± 8.33	21.84 ± 9.29	**73.13±4.98**	53.33 ±10.91	52.96±12.69	17.81 ± 10.72	18.63 ± 11.45
	SS	**0.57±0.06**	0.45 ± 0.06	0.46 ± 0.07	0.33 ± 0.06	0.35 ± 0.13	**0.61±0.06**	0.44±0.03	0.41±0.04	0.24 ± 0.06	0.24 ± 0.06
	IE	**7.34±0.26**	4.95 ± 0.37	4.84 ± 0.35	4.17 ± 0.73	4.18 ± 0.79	**7.62±0.30**	4.68±0.39	4.11±0.40	1.43 ± 0.92	1.44 ± 0.94
QWen-2	LLM	**63.84±8.12**	53.47±11.00	42.59±11.53	21.32 ± 8.85	23.07±10.69	**68.69±5.97**	56.87±8.34	46.76±13.68	17.81 ± 10.72	18.63 ± 11.45
	SS	**0.62±0.05**	0.50±0.07	0.46 ± 0.08	0.33 ± 0.06	0.35 ± 0.13	**0.61±0.05**	0.47±0.04	0.38±0.05	0.24 ± 0.06	0.24 ± 0.06
	IE	**7.79 ± 0.36**	6.24 ± 0.42	4.98 ± 0.56	4.17 ± 0.73	4.18 ± 0.79	**7.74±0.36**	5.55±0.54	3.89±0.49	1.43 ± 0.92	1.44 ± 0.94
Vicuna-7B	LLM	**61.39 ± 8.62**	53.32 ± 9.30	40.31 ± 11.27	21.31 ± 9.37	21.97 ± 9.47	**65.26±6.69**	53.43±8.97	47.55±13.42	17.81 ± 10.72	18.63 ± 11.45
	SS	**0.62±0.06**	0.51 ± 0.05	0.46±0.07	0.33±0.06	0.35±0.13	**0.59±0.06**	0.50±0.04	0.41±0.06	0.24 ± 0.06	0.24 ± 0.06
	IE	**7.72±0.36**	6.58±0.31	5.19±0.69	4.17 ± 0.73	4.18 ± 0.79	**7.45±0.39**	6.18±0.44	4.42±1.14	1.43 ± 0.92	1.44 ± 0.94

Table 1: (RQ1) Diversity comparison by various metrics: LLM, Similarity Spread (SS), and Information Entropy (IE).

Dataset	Reviews	features	Items	Edges	Test Data
Amazon	10,000	5596	6339	9,917,414	100
Airbnb	10,000	2486	3647	9,496,960	100

Table 2: Statistics for Graph Construction of Amazon and Airbnb, including the number of original user reviews, extracted features by LLM, items, graph edges, and test samples.

and combined review text lengths greater than 100 characters for both datasets

Furthermore, to verify RQ4, we utilize web crawlers to collect electronic item data belonging from "cell phone and accessories category" on BestBuy [2], extracting details such as item titles, expert reviews, and e-commerce content.

4.1.2 ***Baselines.*** We compare the proposed method with several strong baselines to generate e-commerce content.

- ***Response by LLMs directly:*** LLMs are leveraged to generate e-commerce content directly from item titles.
- ***Vanilla RAG:*** The details of the most similar items, along with the item title, are utilized by LLMs to generate the e-commerce content.
- ***BART [16]:*** A large encoder-decoder model, as our baseline for generating e-commerce content by a composite of item titles and attributes.
- ***Transformer [28]:*** Vanilla transformer model is used to take item titles as input for generating e-commerce content.

Experiments are conducted across various LLMs, including the current well-known closed-source LLMs, *i.e.,* ChatGPT-4o [1], Gemini [27], and Deepseek-V3 [20], and open-source LLMs, *i.e.,* LLaMA-3.2 [8], Vicuna-7B, and QWen-2 [5].

[2]https://www.bestbuy.com/

4.2 Diversity Evaluation (RQ1)

We evaluate whether our graph-based RAG framework improves the diversity of generated e-commerce content compared to the baseline methods.

4.2.1 Evaluation Metrics. To measure diversity, we consider both lexical and semantic perspectives. Instead of solely relying on traditional diversity metrics, such as k-distinct, which often introduces biases by penalizing longer sequences [21], we adopt *Information Entropy* to capture the richness and balance of n-gram distribution. The entropy score, defined in Eq. (13), quantifies diversity by assigning higher values to more uniform and diverse n-gram usage, while lower values indicate concentrated or repetitive patterns.

$$\text{Entropy Score} = -\sum_{i=1}^{N} P(i) \cdot \log P(i) \tag{13}$$

Additionally, we use Similarity Spread to assess semantic diversity by calculating the range of cosine similarity between word embeddings, where a wider spread reflects greater semantic variation in the generated descriptions. To complement these metrics, we utilize the powerful natural language understanding capabilities of the current state-of-the-art LLM, ChatGPT-4o [1], for LLM-based diveristy evaluation. This evaluation provides a diversity score on a scale from 0 to 100, with higher score indicating greater diversity in the generated outputs, ensuring a comprehensive and holistic understanding of the output diversity.

4.2.2 Results Analysis. The results presented in Table 1 clearly demonstrate that our graph-based RAG significantly outperforms all baseline methods across multiple diversity metrics. Specifically, our method achieves higher Information Entropy, indicating richer and more balanced n-gram distributions, as well as greater Similarity Spread, reflecting broader semantic variation. Additionally, the LLM-based diversity scores further validate the effectiveness of our method. Furthermore, when applying different LLMs for content

Models	textbfMetrics	Amazon					Airbnb				
		Ours	RAG (w)	RAG (w/o)	Transformer	Bart	Ours	RAG (w)	RAG (w/o)	Transformer	Bart
ChatGPT-4o	LLM	73.62 ± 13.29	71.4 ± 14.59	70.14 ± 17.02	69.21 ± 18.93	59.32 ± 20.76	49.10 ± 18.30	57.85 ± 17.17	47.23 ± 18.30	11.88 ± 16.78	11.40 ± 16.03
	BS	0.83 ± 0.02	0.84 ± 0.02	0.84 ± 0.02	0.83 ± 0.04	0.82 ± 0.04	0.81 ± 0.02	0.83 ± 0.02	0.83 ± 0.02	0.79 ± 0.02	0.79 ± 0.02
Deepseek-V3	LLM	71.27 ± 13.81	68.47 ± 16.44	69.10 ± 17.37	69.47 ± 19.19	59.49 ± 20.69	49.94 ± 16.54	37.16 ± 19.51	25.69 ± 19.70	13.52 ± 17.82	10.37 ± 15.17
	BS	0.84 ± 0.02	0.84 ± 0.03	0.84 ± 0.03	0.83 ± 0.04	0.82 ± 0.04	0.82 ± 0.02	0.83 ± 0.02	0.83 ± 0.02	0.79 ± 0.02	0.79 ± 0.02
Gemini	LLM	73.62 ± 13.29	71.4 ± 14.59	70.14 ± 17.02	69.21 ± 18.93	59.32 ± 20.76	50.05 ± 17.51	35.74 ± 19.03	45.2 ± 21.53	10.9 ± 15.89	10.52 ± 15.42
	BS	0.83 ± 0.02	0.84 ± 0.02	0.84 ± 0.03	0.83 ± 0.04	0.82 ± 0.04	0.82 ± 0.01	0.84 ± 0.02	0.84 ± 0.02	0.79 ± 0.02	0.79 ± 0.02
Claude	LLM	72.74 ± 14.33	69.11 ± 16.84	74.44 ± 14.81	70.2 ± 17.25	60.51 ± 19.78	49.43 ± 16.38	44.68 ± 19.75	25.25 ± 20.38	11.55 ± 15.23	9.6 ± 15.62
	BS	0.83 ± 0.02	0.84 ± 0.02	0.84 ± 0.02	0.83 ± 0.04	0.82 ± 0.04	0.83 ± 0.02	0.84 ± 0.02	0.83 ± 0.02	0.79 ± 0.02	0.79 ± 0.02
Llama-3.2-8B	LLM	70.09 ± 14.20	66.63 ± 16.51	71.75 ± 16.16	68.98 ± 19.03	59.58 ± 20.73	47.93 ± 16.75	36.26 ± 19.36	26.88 ± 18.51	9.97 ± 14.98	10.67 ± 16.29
	BS	0.83 ± 0.02	0.84 ± 0.02	0.84 ± 0.02	0.83 ± 0.04	0.82 ± 0.04	0.82 ± 0.02	0.84 ± 0.01	0.84 ± 0.02	0.79 ± 0.02	0.79 ± 0.02
QWen-2	LLM	64.54 ± 14.20	69.91 ± 14.08	69.74 ± 17.25	70.3 ± 17.25	59.8 ± 19.78	46.1 ± 19.07	34.48 ± 19.05	20.12 ± 18.17	11.28 ± 15.43	11.10 ± 16.30
	BS	0.83 ± 0.02	0.84 ± 0.02	0.85 ± 0.03	0.83 ± 0.04	0.82 ± 0.04	0.82 ± 0.02	0.83 ± 0.02	0.83 ± 0.02	0.79 ± 0.02	0.79 ± 0.02
Vicuna-7B	LLM	63.28 ± 12.82	68.69 ± 14.92	70.33 ± 15.43	69.25 ± 17.75	61.19 ± 20.42	43.46 ± 19.93	32.86 ± 21.40	18.77 ± 19.0	11.75 ± 17.06	12.07 ± 16.23
	BS	0.82 ± 0.02	0.84 ± 0.02	0.85 ± 0.03	0.83 ± 0.04	0.82 ± 0.04	0.82 ± 0.02	0.83 ± 0.02	0.83 ± 0.02	0.79 ± 0.02	0.79 ± 0.02

Table 3: (RQ2) Faithfulness comparison by LLM and Bert Score (BS).

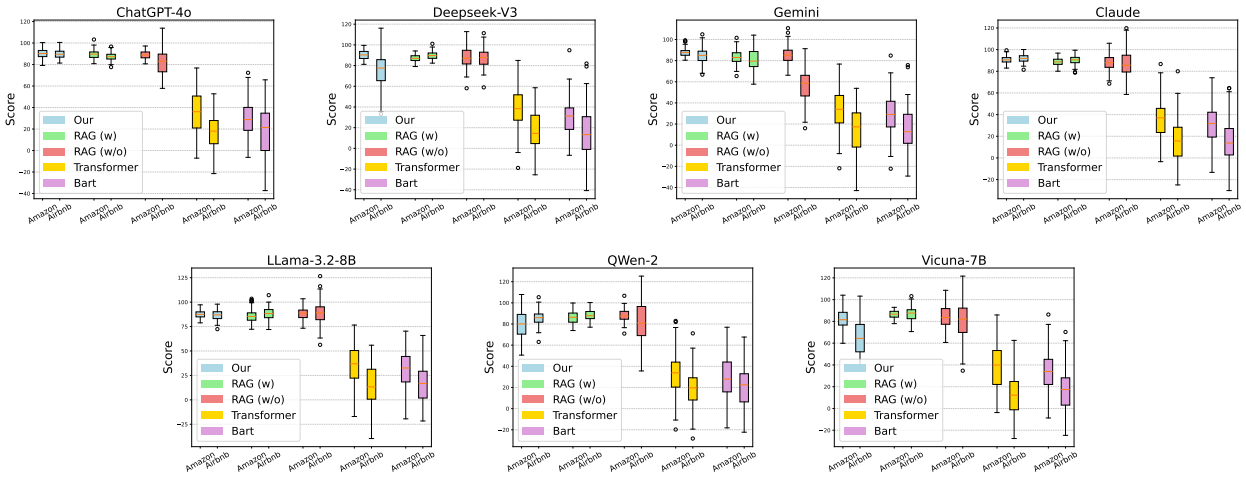

Figure 5: (RQ3) Textual Coherence comparison by LLM.

generation after retrieving item-related features, our approach consistently achieves superior results, demonstrating the robustness and generalizability across different models and settings.

4.3 Faithfulness Evaluation (RQ2)

We evaluate whether our graph-based RAG framework maintains factual accuracy and aligns with the ground-truth e-commerce content compared to baseline methods.

4.3.1 Evaluation Metrics. To evaluate the fidelity of the generated e-commerce content, we focus on semantic-level evaluation rather than relying solely on lexical overlap. We employ BERT score [32], which measures the alignment between the generated text and reference ground-truth descriptions using token-level similarity based on contextual embeddings from a pre-trained BERT model. This metric provides precision, recall, and F1 scores to capture nuanced differences. In addition, similar to the LLM-based evaluation for Q1, we use ChatGPT-4o to assign fidelity scores (ranging from 0 to 100), where higher scores indicate greater fidelity. These evaluation metrics ensure a comprehensive assessment of the factual consistency and accuracy of the generated descriptions.

4.3.2 Results Analysis. The experimental results, shown in Table 3, clearly demonstrate that our approach consistently outperforms all baseline methods in terms of LLM-based fidelity scores. Additionally, our method maintains consistently high performance on BERTScore when compared to the baselines. These findings highlight that our approach effectively enhances the diversity of the generated e-commerce content without causing significant sacrifices in factual accuracy or fidelity.

4.4 Textual Coherence Evaluation (RQ3)

In this section, we investigate whether the generated descriptions exhibit logical consistency, maintain a natural flow of information, and avoid contradictions across their content.

4.4.1 Evaluation Metrics. Following LLM-based evaluation in RQ1 and RQ2, we utilize ChatGPT-4o to assign scores of coherence for generated e-commerce content from 0 to 100. Higher coherence scores indicate stronger logical flow and greater semantic consistency of the generated e-commerce content.

RQ	Metrics	Methods	ChatGPT-4o	Deepseek-V3	Gemini	Claude	LLama-3.2-8B	QWen-2	Vicuna-7B
Diversity (RQ1)	LLM	Our	72.54±4.76	68.72±6.10	68.25 ± 5.47	66.65±7.92	68.74±5.57	63.79±7.03	61.71±9.05
		RAG (w)	61.84 ± 6.86	54.40 ± 9.96	48.34± 11.01	54.12±10.06	51.58±12.22	55.07±8.84	57.0±9.13
		RAG (w/o)	54.12 ± 11.04	47.37 ± 11.04	29.37 ± 10.46	46.55 ± 10.44	51.82±10.36	37.83±10.92	38.33±10.26
		Transformer	14.77 ± 5.37	14.51 ± 5.94	14.77 ± 5.37	20.78±9.01	14.80±6.96	13.68±5.57	13.48±5.15
		Bart	14.51 ± 5.94	4.77 ± 5.37	4.51 ± 5.94	19.32±8.68	14.34±6.58	14.33±5.92	14.33±6.18
	SS	Our	0.56 ± 0.03	0.56 ± 0.03	0.55±0.03	0.57 ± 0.03	0.60 ± 0.05	0.62±0.04	0.63±0.06
		RAG (w)	0.48 ± 0.03	0.45±0.04	0.47±0.03	0.47 ± 0.04	0.45 ± 0.04	0.48±0.04	0.50 ± 0.06
		RAG (w/o)	0.46 ± 0.03	0.46±0.04	0.48 ±0.05	0.47 ± 0.03	0.47 ± 0.04	0.44±0.05	0.48±0.10
		Transformer	0.25 ± 0.04	0.25±0.04	0.25±0.04	0.25 ± 0.04	0.25 ± 0.04	0.25±0.04	0.25 ± 0.04
		Bart	0.27 ± 0.05	0.27±0.05	0.27 ± 0.05	0.27 ± 0.05	0.27 ± 0.05	0.27 ± 0.05	0.27 ± 0.05
	IE	Our	7.24 ± 0.16	6.99±0.14	6.89 ± 0.16	6.99 ± 0.10	7.54±0.25	7.75±0.37	7.70±0.53
		RAG (w)	4.88 ± 0.24	5.32 ± 0.25	4.51 ± 0.36	5.25 ± 0.54	4.73 ± 0.33	6.01 ± 0.48	6.22 ± 0.50
		RAG (w/o)	6.17 ± 0.27	4.34 ± 0.28	3.84 ± 0.30	4.55 ± 0.31	4.59 ± 0.27	4.14 ± 0.41	4.59 ± 0.88
		Transformer	2.99 ± 0.30	2.99 ± 0.30	2.99 ± 0.30	2.99 ± 0.30	2.99 ± 0.30	2.99 ± 0.30	2.98 ± 0.30
		Bart	3.05 ± 0.33	3.05 ± 0.33	3.05 ± 0.33	3.05 ± 0.33	3.05 ± 0.33	9.05 ± 0.33	3.05 ± 0.33
Fidelity (RQ2)	LLM	Our	59.08 ± 9.84	58.73 ± 11.57	58.99 ± 11.0	55.7±10.64	58.47±10.73	56.21±10.77	54.86±12.84
		RAG (w)	61.84 ± 6.86	53.56 ± 13.46	53.87 ± 14.44	41.52 ± 14.93	50.70 ± 12.02	56.30 ± 13.21	53.85 ± 15.13
		RAG (w/o)	54.12 ± 11.04	53.13 ± 16.88	51.45 ± 14.65	52.36 ± 16.48	49.14 ± 15.18	50.88 ± 15.73	50.21 ± 14.76
		Transformer	11.04 ± 6.54	48.75 ± 21.06	48.77 ± 18.81	49.59 ± 19.33	48.82 ± 19.66	47.42 ± 21.03	49.11 ± 19.77
		Bart	14.19 ± 6.24	47.02 ± 17.43	47.00 ± 18.48	46.05 ± 17.09	49.27 ± 16.25	48.15 ± 16.83	48.86 ± 17.04
	BS	Our	0.83 ± 0.01	0.83 ± 0.01	0.83 ± 0.01	0.83±0.01	0.82±0.01	0.82±0.01	0.82±0.01
		RAG (w)	0.84 ± 0.01	0.84 ± 0.02	0.84 ± 0.02	0.83 ± 0.01	0.84 ± 0.02	0.84 ± 0.01	0.83 ± 0.01
		RAG (w/o)	0.84 ± 0.02	0.84 ± 0.02	0.84 ± 0.02	0.83 ± 0.01	0.84 ± 0.02	0.84 ± 0.02	0.84 ± 0.02
		Transformer	0.82 ± 0.02	0.82 ± 0.02	0.82 ± 0.02	0.82 ± 0.02	0.82 ± 0.02	0.82 ± 0.02	0.82 ± 0.02
		Bart	0.82 ± 0.02	0.82 ± 0.02	0.82 ± 0.02	0.82 ± 0.02	0.82 ± 0.02	0.82 ± 0.02	0.82 ± 0.02
Coherence (QR3)	LLM	Our	89.92 ± 3.48	89.83 ± 3.64	86.28 ± 2.68	88.65 ± 3.59	86.74 ± 3.42	86.49 ± 5.69	85.61 ± 6.72
		RAG (w)	88.73 ± 2.95	88.96 ± 3.75	80.15 ± 9.69	85.72 ± 4.72	81.50 ± 11.06	84.02 ± 7.99	85.07 ± 6.62
		RAG (w/o)	89.73 ± 4.93	91.22 ± 7.20	85.20 ± 13.47	89.60 ± 6.89	91.30 ± 6.03	86.35 ± 11.51	83.77 ± 13.02
		Transformer	24.05 ± 22.08	24.05 ± 22.08	24.05 ± 22.08	24.05 ± 22.08	24.05 ± 22.08	24.05 ± 22.08	24.05 ± 22.08
		Bart	27.90 ± 21.82	27.90 ± 21.82	27.90 ± 21.82	27.90 ± 21.82	27.90 ± 21.82	27.90 ± 21.82	27.90 ± 21.82

Table 4: (RQ4) Generalizability evaluation on BestBuy dataset: Diversity comparison using LLM, Similarity Spread (SS), and Information Entropy (IE); Fidelity comparison using LLM and BERTScore (BS); Coherence comparison using LLM.

4.4.2 Results Analysis. The results in Figure 5 demonstrate that our proposed approach ("Our") consistently achieves high coherence across both Amazon and BestBuy datasets, regardless of the underlying language model used for comparison. This highlights the robustness and adaptability of our method in generating e-commerce content that maintain logical flow and semantic consistency.

Experiments conducted with various large language models, including ChatGPT-4o, Deepseek-V3, and Gemini, further validate the effectiveness of our approach. Across different models, our method consistently outperforms or matches the coherence levels of baseline methods, demonstrating its strong generalization capabilities. These results underscore the versatility and reliability of our approach in producing coherent, high-quality e-commerce content across diverse scenarios and settings.

4.5 Generalizability Evaluation (RQ4)

We also evaluate whether our constructed graph exhibits strong generalizability, ensuring its applicability in practical scenarios beyond the dataset it was originally built on.

4.5.1 Evaluation Methodology. To evaluate the generalizability of our approach, we use the BestBuy dataset as the test dataset, while allowing the RAG framework operate on the graph constructed from the Amazon dataset. This setup allows us to evaluate whether the structured knowledge representation in our graph remains effective when applied to a different domain. We analyze the performance on the key metrics of Diversity (RQ1), Fidelity (RQ2), and Coherence (RQ3) when transitioning from Amazon-based knowledge to the BestBuy dataset.

4.5.2 Results Analysis. The results demonstrated in Table 4 show that our approach consistently outperforms baselines in diversity, faithfulness, and coherence when applied to the BestBuy dataset, despite the graph being built on Amazon dataset. These findings highlight the strong generalization capability of our method, suggesting that it can be effectively adapted to practical applications across different domains with minimal performance degradation.

4.6 Ablation Study (RQ5)

This evaluation investigates how varying the number of traverse steps for getting features on the graph impacts the diversity score of the generated e-commerce content. Furthermore, we change α in Eq. (10) to investigate the inner diversity of captured features.

4.6.1 Evaluation Methodology. We employ the diversity score by LLMs as the metric to evaluate the diversity changes of responses by providing a different number of features. For the observation of changes of α, we leverage the cosine similarity by feature embeddings to calculate the inner diversity of traversed features.

4.6.2 Results Analysis. We vary the number of traverse steps in the graph, which in turn affects the number of features inputted

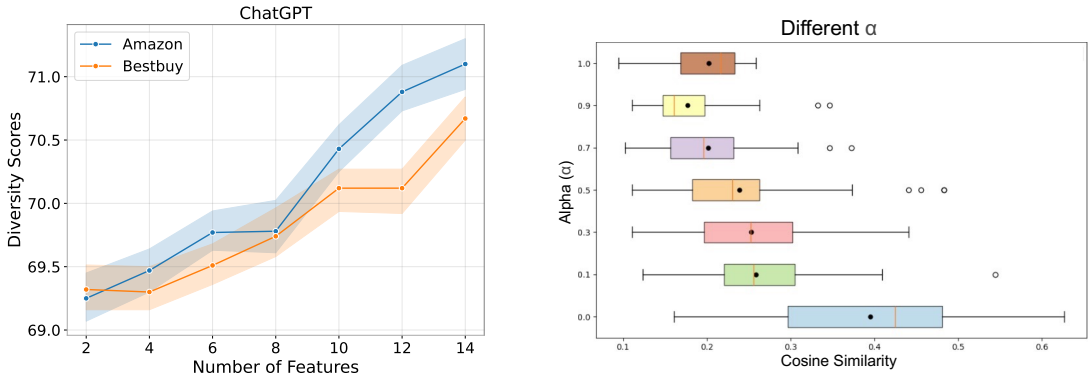

Figure 6: (RQ5) *Left:* Changes of the number of features on Amazon and Bestbuy dataset using various LLMs for the first seven figures; *Right:* Changes in inner diversity of traversed features across different α values.

into the LLM. Experiments are conducted on both the Amazon and BestBuy datasets using LLMs mentioned in RQ1 for experiments. The results are shown in Figure 6 (Left) and Figure 7 in the appendix demonstrates that as the number of traverse steps increases, the diversity score also shows a corresponding improvement. This trend highlights the importance of deeper graph exploration, enabling the model to access a richer set of features and generate more diverse e-commerce content.

5 Conclusion

In this work, we introduced a graph-based RAG framework to generate product descriptions using large language models. By integrating product-related features into a heterogeneous graph and employing a structured exploration strategy, our approach effectively enhance the diversity of generated product descriptions. Extensive experiments validate the method's ability to improve diversity while maintaining fidelity and coherence, offering a significant advancement in automated product description generation for e-commerce.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774* (2023).

[2] Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, et al. 2022. Improving language models by retrieving from trillions of tokens. In *International conference on machine learning*. PMLR, 2206–2240.

[3] Zhangming Chan, Xiuying Chen, Yongliang Wang, Juntao Li, Zhiqiang Zhang, Kun Gai, Dongyan Zhao, and Rui Yan. 2019. Stick to the facts: Learning towards a fidelity-oriented e-commerce product description generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 4959–4968.

[4] Qibin Chen, Junyang Lin, Yichang Zhang, Hongxia Yang, Jingren Zhou, and Jie Tang. 2019. Towards knowledge-based personalized product description generation in e-commerce. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 3040–3050.

[5] Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, et al. 2024. Qwen2-audio technical report. *arXiv preprint arXiv:2407.10759* (2024).

[6] R Dennis Cook. 1977. Detection of influential observation in linear regression. *Technometrics* 19, 1 (1977), 15–18.

[7] Yang Deng, Yaliang Li, Wenxuan Zhang, Bolin Ding, and Wai Lam. 2022. Toward personalized answer generation in e-commerce via multi-perspective preference modeling. *ACM Transactions on Information Systems (TOIS)* 40, 4 (2022), 1–28.

[8] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783* (2024).

[9] Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. 2024. A survey on rag meeting llms: Towards retrieval-augmented large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 6491–6501.

[10] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997* (2023).

[11] Xiaojie Guo, Qingkai Zeng, Meng Jiang, Yun Xiao, Bo Long, and Lingfei Wu. 2022. Automatic controllable product copywriting for e-commerce. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2946–2956.

[12] Shaoyang Hao, Bin Guo, Hao Wang, Yunji Liang, Lina Yao, Qianru Wang, and Zhiwen Yu. 2021. DeepDepict: enabling information rich, personalized product description generation with the deep multiple pointer generator network. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 15, 5 (2021), 1–16.

[13] Xiaoxin He, Yijun Tian, Yifei Sun, Nitesh V Chawla, Thomas Laurent, Yann LeCun, Xavier Bresson, and Bryan Hooi. 2024. G-retriever: Retrieval-augmented generation for textual graph understanding and question answering. *arXiv preprint arXiv:2402.07630* (2024).

[14] Yupeng Hou, Jiacheng Li, Zhankui He, An Yan, Xiusi Chen, and Julian McAuley. 2024. Bridging language and items for retrieval and recommendation. *arXiv preprint arXiv:2403.03952* (2024).

[15] Fajri Koto, Jey Han Lau, and Timothy Baldwin. 2022. Can pretrained language models generate persuasive, faithful, and informative Ad text for product descriptions?. In *Proceedings of the Fifth Workshop on e-Commerce and NLP (ECNLP 5)*. 234–243.

[16] M Lewis. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461* (2019).

[17] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems* 33 (2020), 9459–9474.

[18] Yunxin Li, Baotian Hu, Wenhan Luo, Lin Ma, Yuxin Ding, and Min Zhang. 2024. A Multimodal In-Context Tuning Approach for E-Commerce Product Description Generation. *arXiv preprint arXiv:2402.13587* (2024).

[19] Yu-Sen Liang, Chih-Yao Chen, Cheng-Te Li, and Sheng-Mao Chang. 2024. Personalized Product Description Generation With Gated Pointer-Generator Transformer. *IEEE Transactions on Computational Social Systems* (2024).

[20] Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437* (2024).

[21] Siyang Liu, Sahand Sabour, Yinhe Zheng, Pei Ke, Xiaoyan Zhu, and Minlie Huang. 2022. Rethinking and refining the distinct metric. *arXiv preprint arXiv:2202.13587* (2022).

[22] Linhao Luo, Yuan-Fang Li, Gholamreza Haffari, and Shirui Pan. 2023. Reasoning on graphs: Faithful and interpretable large language model reasoning. *arXiv preprint arXiv:2310.01061* (2023).

[23] Costas Mavromatis and George Karypis. 2024. GNN-RAG: Graph Neural Retrieval for Large Language Model Reasoning. *arXiv preprint arXiv:2405.20139* (2024).

[24] Haoran Que, Feiyu Duan, Liqun He, Yutao Mou, Wangchunshu Zhou, Jiaheng Liu, Wenge Rong, Zekun Moore Wang, Jian Yang, Ge Zhang, et al. 2024. Hellobench: Evaluating long text generation capabilities of large language models. *arXiv preprint arXiv:2409.16191* (2024).

[25] Huajie Shao, Jun Wang, Haohong Lin, Xuezhou Zhang, Aston Zhang, Heng Ji, and Tarek Abdelzaher. 2021. Controllable and diverse text generation in e-commerce. In *Proceedings of the Web Conference 2021*. 2392–2401.

[26] Jiashuo Sun, Chengjin Xu, Lumingyuan Tang, Saizhuo Wang, Chen Lin, Yeyun Gong, Heung-Yeung Shum, and Jian Guo. 2023. Think-on-graph: Deep and responsible reasoning of large language model with knowledge graph. *arXiv preprint arXiv:2307.07697* (2023).

[27] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805* (2023).

[28] A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems* (2017).

[29] Jinpeng Wang, Yutai Hou, Jing Liu, Yunbo Cao, and Chin-Yew Lin. 2017. A statistical framework for product description generation. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*. 187–192.

[30] Keheng Wang, Feiyu Duan, Sirui Wang, Peiguang Li, Yunsen Xian, Chuantao Yin, Wenge Rong, and Zhang Xiong. 2023. Knowledge-driven cot: Exploring faithful reasoning in llms for knowledge-intensive question answering. *arXiv preprint arXiv:2308.13259* (2023).

[31] Yongzhen Wang, Jian Wang, Heng Huang, Hongsong Li, and Xiaozhong Liu. 2020. Evolutionary product description generation: A dynamic fine-tuning approach leveraging user click behavior. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 119–128.

[32] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675* (2019).

[33] Tao Zhang, Jin Zhang, Chengfu Huo, and Weijun Ren. 2019. Automatic generation of pattern-controlled product description in e-commerce. In *The World Wide Web Conference*. 2355–2365.

[34] Xueying Zhang, Yanyan Zou, Hainan Zhang, Jing Zhou, Shiliang Diao, Jiajia Chen, Zhuoye Ding, Zhen He, Xueqi He, Yun Xiao, et al. 2022. Automatic product copywriting for e-commerce. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 12423–12431.

A Ablation Study

To further analyze the impact of the number of traversed features on diversity, we extend the results from Figure 6 (left) in RQ5, which primarily focuses on ChatGPT, by presenting additional results for other LLMs in Figure 7. Results in Figure 7 illustrate the diversity score variations across different numbers of features for various LLMs on the Amazon and BestBuy datasets. The trends consistently show that increasing the number of features leads to higher diversity scores, reinforcing the importance of incorporating more traversed features during graph exploration.

B Graph Analysis

To better understand the structure of our graph, we present statistical analysis on node degrees and neighbor distributions of the graph constructed by Amazon dataset. The top figure in Figure 8 shows the overall node degree distribution, while the remaining two figures present the distributions of product-type neighbors and cluster-type neighbors, respectively. These distributions provide insights into the structural properties of our graph, which influence the traversal and feature selection processes.

C Case Study

To further illustrate the effectiveness of our method, we conducted case studies on both the Amazon and BestBuy datasets. The examples in Figure 9, Figure 10 and Figure 11 for three different dataset highlight how our approach leverages features extracted during graph traversal to generate coherent and diverse product descriptions.

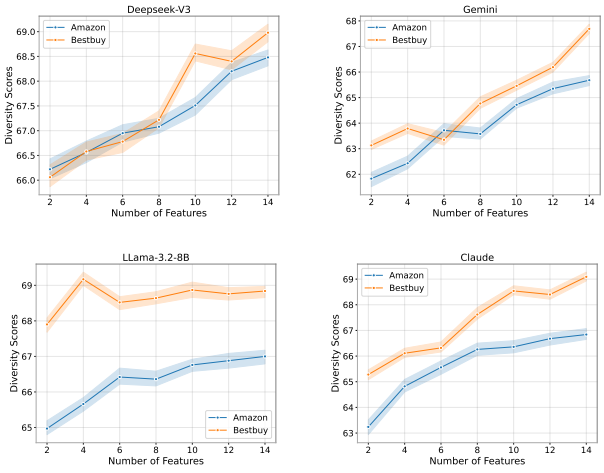

Figure 7: (RQ5) Changes of the number of features on Amazon and Bestbuy dataset using various LLMs.

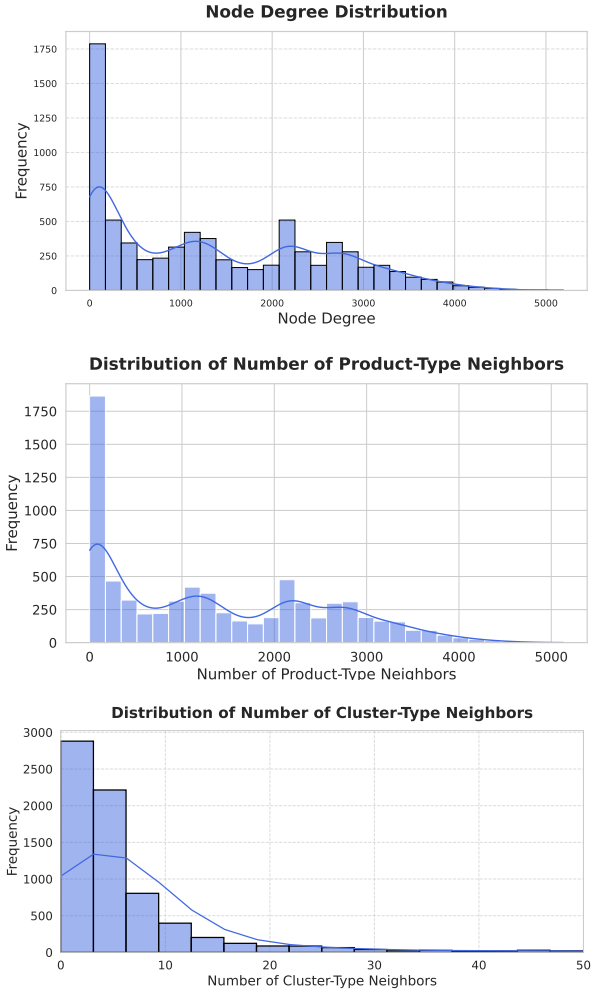

Figure 8: Graph Analysis.

Product Title

AFTRYOUGO Battery Case for iPhone 12/12 Pro(6.1 inch), [7200mAh] Portable Charging Rechargeable External Pack Protective Extended Backup Charger Pro(XDL-640MW)-Black"

Response by our method

Response: The AFTRYOUGO Battery Case for iPhone 12/12 Pro is your ultimate power companion, featuring a robust 7200mAh battery that extends your device's usage time significantly. Designed with full protection and a sleek black rubber finish, this case ensures your phone remains safe and stylish. It supports simultaneous charging for your iPhone and AirPods, and is compatible with Apple Pay and CarPlay. With its portable design and reliable power functionality, this case is perfect for on-the-go users who need extended battery life without compromising on protection or convenience.

Figure 9: Case study on Amazon dataset.

Product Title

Cozy Central Suite (with breakfast and more)

Response by our method

Response: Discover the perfect blend of comfort and convenience at our Cozy Central Suite, where your stay includes breakfast to start each day on a delightful note. Located in a prime area, guests have consistently praised its immaculately clean room and proximity to a myriad of attractions, making this a highly sought-after spot for explorers and city adventurers alike. Embrace practicality infused with modern technology; our suite incorporates face recognition for a seamless check-in experience, ensuring both security and ease of access. Despite the central location, attention has been paid to noise insulation, providing a peaceful retreat right in the city's vibrant heart. With thoughtful design touches that consider case type and smooth edges, your stay promises an elegant and safe environment. Experience the warmth and friendliness our past guests have highlighted, and enjoy a relaxed and laid-back atmosphere that makes you feel at home the moment you step in.

Figure 10: Case study on Airbnb dataset.

Product Title

Apple - iPhone 16 Plus 128GB - Apple Intelligence - Ultramarine (Verizon)

Response by our method

Response: Introducing the Apple iPhone 16 Plus in a captivating Ultramarine finish, available with 128GB of storage for Verizon users. This masterpiece of modern technology is designed with impeccable build quality, ensuring durability and resilience that users adore. Not only does it present an impressive aesthetic, but it also promises to endure the rigors of daily use without faltering. Coupled with its competitive pricing, this device brings premium value within reach. The iPhone 16 Plus boasts exceptional battery life, keeping you powered throughout the day, while its innovative charging capabilities ensure you spend more time enjoying your phone than waiting for it to power up. With an intelligently designed charging cable that aligns with Apple's focus on user-centric design, you'll experience optimal charging speed and convenience, making it an essential companion for both work and play.

Figure 11: Case study on Bestbuy dataset.