The Problem of Search Recall in B2B DIY Product Catalogs: Limitations of Semantic Embeddings and an Entity-Oriented Approach

Fedor Krasnov

***Abstract*—This paper examines the challenge of ensuring high search recall in B2B catalogs for DIY products (tools and materials for construction and repair). In practice, B2B search imposes significantly stricter recall requirements than B2C recommendation scenarios, as a substantial proportion of queries correspond to known-item retrieval involving exact SKUs, model numbers, and precise technical specifications. Even minor deviations in identifiers or numeric attributes may render results unusable in professional procurement contexts.**

The evolution of the search architecture is analyzed, moving from pure dense retrieval based on transformer embeddings (ModernBERT trained with triplet loss) to a hybrid system grounded in entity detection (brand, model, technical specifications) and structured entity matching. It is demonstrated that embedding-based retrieval does not guarantee recall due to smoothing of numerical tokens, semantic compression of identifiers, and the inherent properties of approximate nearest neighbor (ANN) search. Geometric characteristics of embedding spaces introduce invariance to discrete differences that are critical in B2B scenarios.

The proposed anchor-based retrieval architecture yields a substantial improvement in Recall@10 (from 0.65 to 0.97 on real-world data) while preserving acceptable latency and high explainability. Experiments were conducted on an industrial catalog exceeding 10 million items and a dataset of 5,000 real B2B queries. The results demonstrate practical viability for large-scale B2B e-commerce search systems.

***Keywords*—B2B search, DIY products, search recall, dense retrieval, entity matching, hybrid architectures.**

I. Introduction

In B2B scenarios, product search is an integral part of the operational process: procurement, invoicing, tender procedures, and specification development. Unlike B2C, where semantic approximation ("similar products") is acceptable [1–5], precise identification is critical in the B2B context [6–8]. An error in the search results leads not only to reduced user satisfaction but also to direct operational costs.

Formally, let Q be the set of user queries, D the product catalog, and $R(q) \subset D$ the set of relevant documents for query q. Search recall is defined as [9]:

$$Recall@K(q) = \frac{\left|TopK(q) \cap R(q)\right)}{|R(q)|}.$$

In a B2B context, the set of relevant documents for a significant portion of queries degenerates [10]:

$$|R(q)| = 1.$$

Let the single relevant document be denoted as d^*. The recall formula then simplifies to:

$$Recall@K(q) = \begin{cases} 1, & \text{if} d^* \in TopK(q), \\ 0, & \text{otherwise.} \end{cases}\Big)$$

In other words, when $|R(q)| = 1$, the Recall@K metric becomes equivalent to an indicator function:

$$Recall@K(q) = 1\{d^* \in TopK(q)\}.$$

Thus, Recall in B2B tasks acquires a binary character: the system either retrieves the correct entity, or a loss of the relevant object occurs. Unlike B2C scenarios where partial relevance and multiple relevant results are permitted, here a deviation even by a single position represents a factual search failure.

Consider a probabilistic interpretation [11–13]. Let the random variable $X_q = 1\{d^* \in TopK(q)\}$ describe the retrieval success for query q. Then the mathematical expectation:

$$E[X_q] = P\left(d^* \in TopK(q)\right)$$

is interpreted as the probability of successfully retrieving the correct product item. The average Recall@K over a sample of queries is an empirical estimate of this probability:

$$Recall@K = \frac{1}{|Q|}\sum_{q \in Q} 1\{d^* \in TopK(q)\}.$$

Consequently, the task of increasing Recall in B2B catalogs reduces to maximizing the probability of accurate entity extraction. Even a slight decrease in this probability leads to direct business risks.

Such a problem statement imposes significant limitations on the applicability of methods based solely on semantic similarity. In dense retrieval, relevance is modeled through a continuous similarity function in a vector space:

[1]Manuscript received February 23, 2026.
F. Krasnov fedor.krasnov@vseinstrumenti.ru, ViTech R&D, ORCID: 0000-0002-9881-7371.

$$Score(q,d) = cos(e_q, e_d),$$

where e_q, e_d are the query and document embeddings. The optimization of such a function aims to minimize the average distance between semantically similar objects but does not guarantee the preservation of discrete identifiers (model, SKU number, exact numerical value).

Under conditions where the target metric is equivalent to an indicator function, even a small distortion of distances in the embedding space can cause d^* to fall out of the $TopK(q)$ set. Thus, the continuous nature of dense retrieval conflicts with the discrete structure of the target task.

This contradiction is particularly evident in the DIY product segment. DIY catalogs are characterized by the following features:

- A high proportion of numerical characteristics (power, dimensions, voltage, RPM).
- The presence of unique identifiers (model, SKU number, modification).
- Transliteration and mixed alphabets (Latin/Cyrillic).
- A pronounced long tail with a significant number of rare SKUs.

Queries often have a structured character and simultaneously contain the product type, model, and technical specifications, for example:

"Hammer drill HR 3200 C 850 W SDS+"

In such cases, the search task reduces not to determining semantic similarity but to precise matching of a set of entities and parameters. This circumstance serves as the basis for the hypothesis regarding the fundamental limitations of pure dense retrieval in ensuring B2B search Recall and motivates the transition to entity-oriented architectures.

II. Conflict Between Continuous Similarity Functions and Discrete Target Functions

The target function in B2B search when $|R(q)| = 1$ is discrete and can be written as:

$$L_{\text{task}}(q) = 1 - 1\{d^* \in TopK(q)\}.$$

This function is discontinuous with respect to the model parameters: an arbitrarily small change in the scoring function $Score(q,d)$ can change the document ranking and lead to a step-like change in L_{task}.

At the same time, dense retrieval is trained using a continuous surrogate loss function, such as triplet loss [14]:

$$L_{\text{triplet}} = max(0, d(e_q, e_{d^+}) - d(e_q, e_{d^-}) + m)$$

where $d(\cdot,\cdot)$ is a continuous metric in the embedding space [15]. Optimizing L_{triplet} minimizes the average distance between positive and negative pairs but does not directly minimize L_{task}.

Thus, a *loss-mismatch* arises:

$$minL_{\text{triplet}} not \Rightarrow minL_{\text{task}}.$$

Even if $d(e_q, e_{d^+}) < d(e_q, e_d)$ holds for most d, it does not guarantee that d^* will enter $TopK(q)$ in the presence of a large number of candidates close in distance. In conditions of high embedding space density, small fluctuations in distance lead to the relevant document falling out of the top, which is critical given the binary nature of the target function.

III. The Impact of Approximate Nearest Neighbors

In practice, search is performed not using an exact metric but through Approximate Nearest Neighbors (ANN) [16,17]. Let $N_K(q)$ be the set of true K nearest neighbors, and $\hat{N}_K(q)$ be the result of the ANN search. Then there exists a probability of error:

$$P\left(\hat{N}_K(q) \neq N_K(q)\right) > 0.$$

Let $p_{ann} = P\left(d^* \in \hat{N}_K(q) \vee d^* \in N_K(q)\right)$ be the probability of correctly returning the true neighbor using the ANN algorithm.

Then the total probability of retrieving the relevant document is upper-bounded:

$$P\left(d^* \in \hat{N}_K(q)\right) \leq P\left(d^* \in N_K(q)\right) \cdot p_{ann}.$$

Consequently,

$$Recall^{ANN}@K \leq Recall^{Exact}@K \cdot p_{ann}.$$

Even with an ideal embedding model ($Recall^{Exact}@K \approx 1$), the constraint $p_{ann} < 1$ induces an upper bound on the achievable Recall. In tasks where $|R(q)| = 1$, this represents a fundamental limitation on the probability of success.

IV. The Necessity of a Structured Framework

The examined contradictions indicate that the search task in B2B catalogs is not only metric but also structured in nature. Let a query be represented as a set of entities:

$$q \to y = (y_{\text{type}}, y_{\text{brand}}, y_{\text{model}}, y_{\text{tx}_1}, \dots, y_{\text{tx}_m}),$$

where each component belongs to a finite discrete set. Then the search task can be interpreted as a structured prediction problem:

$$\hat{y} = \underset{y \in Y}{argmax} F(q,y),$$

where F is a scoring function accounting for the compatibility between the query and the product structure. Unlike the continuous embedding space, the space Y is discrete and factorizable, which allows for

- Decomposition of the task by entities.
- Use of deterministic matching rules.

- Control over Recall at the level of individual components.

Thus, moving from pure dense retrieval to a structured search model is not an empirical heuristic but a theoretically motivated step arising from the binary nature of the target function and the limitations of ANN.

Dense retrieval optimizes semantic similarity. However, B2B requires precise entity matching. Semantic similarity is not equivalent to identity.

The primary hypothesis of this work is:

> *Embedding-based retrieval does not provide sufficient Recall in B2B DIY catalogs due to information loss regarding entities; entity-based matching provides higher Recall.*

The following sections consider the implementation of an entity-oriented architecture and its integration with embedding models in a hybrid search system.

V. Research Methodology

Given a product catalog $D = \{d_1, \dots, d_N\}$, and a set of user queries $Q = \{q_1, \dots, q_M\}$.

For each query $q \in Q$, a set of relevant documents is defined:

$$R(q) \subseteq D.$$

In the examined B2B scenario, the condition $|R(q)| = 1$ typically holds, corresponding to a search for a specific model, SKU number, or product with a given specification. The task is to construct a ranking function:

$$f: Q \times D \to R,$$

maximizing retrieval recall (Recall@K) under constraints on computational resources and latency.

The baseline model used is the transformer architecture ModernBERT-base [18,19] (149M parameters), mapping text into a vector space of dimension 768:

$$\phi: T \to R^{768}.$$

Similarity between query q and document d is determined by the cosine measure:

$$s(q,d) = cos\big(\phi(q), \phi(d)\big).$$

Ranking is performed in descending order of $s(q,d)$.

Fine-tuning of the model was performed on "user query – product name" pairs using a triplet loss function:

$$\mathcal{L}(a,p,n) = max(0, d(a,p) - d(a,n) + m),$$

where:

- a is the query embedding (anchor),
- p is the relevant product (positive),
- n is an irrelevant product (negative),
- $d(\cdot,\cdot)$ is the cosine distance,
- $m = 0.3$ is the margin.

Empirical risk is minimized:

$$L = E_{(a,p,n)\sim S} L(a,p,n).$$

Negative examples were formed using a hard-negative mining strategy within the batch.

Without augmentation, rapid saturation of the loss function was observed, indicating a predominance of easy negative examples. To increase the complexity of the training sample, the following query transformations were applied:

- Typo simulation.
- Transliteration (Latin vs. Cyrillic).
- Token permutation.
- Token deletion.
- Replacement with synonyms and technical equivalents.

Formally, augmentation implements a mapping $A: q \mapsto \tilde{q}$, where the distribution of $\tilde{q}$ approximates the empirical distribution of distorted user formulations.

For scalable search, the Approximate Nearest Neighbors (ANN) algorithm based on Hierarchical Navigable Small World (HNSW) graphs was used [17,20] via the uSearch framework [21].

Let $I_{ANN}(q,K)$ be the set of documents returned by the ANN algorithm. Then the upper bound of Recall is limited by the probability of the true nearest neighbor falling into the resulting set:

$$Recall@K \le P\left(d^* \in I_{ANN}(q,K)\right),$$

where d^* is the true nearest neighbor in the full space. With a catalog size of approximately 10^7 products, the average response time was about 50 ms.

VI. Entity-based Retrieval

Taking into account the identified conflict between the continuity of similarity functions in embedding space and the discrete nature of relevance in B2B search, an additional entity-oriented retrieval layer was introduced.

The central assumption of this layer is that a substantial fraction of B2B queries contains explicitly structured components (brand, model, SKU number, technical specification), and that exact matching of these components constitutes a necessary condition for relevance.

Unlike dense retrieval, which approximates similarity in a continuous vector space, the entity-based layer operates in a

discrete symbolic space and enforces structural constraints prior to semantic ranking.

A. Brand Extraction

Let B denote a dictionary of brands, $|B| > 10^4$, constructed from the normalized catalog metadata and extended with common spelling variants and transliterations.

A deterministic matching function is defined as:

$$BrandMatch(q,d) = \begin{cases} 1, & \text{if} brand(q) = brand(d), \\ 0, & \text{otherwise.} \end{cases}$$

Brand detection is performed via dictionary-based lookup with normalization (case folding, transliteration, removal of legal suffixes).

This component enforces categorical consistency and eliminates cross-brand semantic proximity that frequently occurs in embedding space due to shared product descriptions.

B. Model and SKU Number Extraction

Models and SKU identifiers are described by a regular language L_{model} that captures alphanumeric patterns, hyphenated codes, and mixed-letter sequences typical for industrial equipment.

The corresponding matching function is defined as:

$$ModelMatch(q,d) = 1\{model(q) = model(d)\}.$$

This component introduces a hard structural constraint. In contrast to embedding-based similarity, which tends to smooth numeric tokens and partially ignore symbol-level differences, exact identifier matching guarantees preservation of discrete distinctions (e.g., 850W vs 800W, HR3200C vs HR3200).

Formally, this module reduces the probability of false positives arising from local geometric proximity in vector space.

C. Technical Specification Processing

Let TX_q, TX_d denote the sets of normalized technical attributes extracted from the query and the document, respectively.

Attributes are normalized by:

- unit standardization (e.g., W, kW → unified representation),
- numeric canonicalization,
- synonym mapping for attribute names.

A partial structural similarity score is then defined as:

$$Score_{tx} = |TX_q \cap TX_d|.$$

Unlike binary brand and model matching, this component allows graded compatibility while remaining interpretable and symbolically grounded.

D. Hybrid Architecture

The final scoring function is defined as:

$$\begin{gathered} \text{Score}(q,d) = 3 \cdot \text{ModelMatch} \\ +2 \cdot \text{BrandMatch} + 1 \cdot \text{TypeMatch} \\ +\alpha \cdot \text{Score}_{tx} + \beta \cdot s(q,d) \end{gathered}$$

where $s(q,d)$ denotes the cosine similarity between dense embeddings of the query and document.

The weighting scheme reflects the structural hierarchy of identifiers in B2B search:

- Model matching has the highest priority (exact identity),
- Brand matching enforces categorical alignment,
- Type matching ensures product class consistency,
- Technical overlap refines compatibility,
- Dense similarity acts as a secondary reranking signal.

Importantly, the dense component is applied predominantly to a filtered subset $D'(q) \subset D$, obtained after entity-based pruning. Thus, the retrieval process follows a two-stage architecture:

1. Structural filtering (entity-based pruning).
2. dense reranking within $D'(q)$.

This design separates the discrete identification phase from the continuous similarity phase. Such separation aligns the optimization objective with the binary nature of recall in B2B search.

The proposed methodology enables empirical validation of the hypothesis that continuous semantic embeddings alone do not ensure sufficient recall in large-scale B2B DIY catalogs. The primary causes are:

- smoothing of discrete identifiers in embedding space,
- insensitivity to fine-grained numeric variations,
- approximation errors introduced by ANN indexing,
- geometric proximity between semantically similar but structurally distinct products.

By introducing an explicit entity-oriented layer, the system restores structural determinism while preserving the ranking flexibility of dense models.

VII. Experimental Setup

The goal of the experiment is a quantitative verification of the hypothesis regarding the insufficient Recall of dense retrieval in B2B DIY catalogs and an assessment of the effect of entity-

oriented and hybrid architectures. The experiment was conducted on an industrial catalog of construction and technical products.

- Catalog size: million SKUs.
- Training sample: "query–product" pairs.
- Test sample: real-world B2B queries.

The test set was formed from production logs and included the following types of queries:

1. Precise model or SKU number search.
2. Queries specifying technical characteristics.
3. Generalized category queries.
4. Noisy and incomplete formulations.

The distribution of queries reflects the real behavior of the B2B audience, where entity-defined formulations dominate. Quality assessment was performed using the following metrics:

- $Recall@10$ — retrieval Recall.
- $P@10$— precision in the top 10.
- $Latency$ — average response time.
- $Explainabilityscore$ — expert assessment of interpretability (scale 1–5).

The Explainability score reflects the degree of transparency in the ranking logic and the reproducibility of the reasons a document appeared in the results. Four architectures were implemented and compared:

1. Dense retrieval (ModernBERT + uSearch HNSW).
2. BM25.
3. Entity-based retrieval.
4. Hybrid (entity pruning + dense reranking).

VIII. Results

Dense retrieval demonstrated limited recall ($Recall@10 = 0.65$), as shown in Table 1, confirming the hypothesis of discrete identifier smoothing in the embedding space.

Table 1. Comparison of Search Architectures

Approach	Recall@10	P@10	Latency	Explainability
Dense	0.65	0.70	50 ms	2
BM25	0.80	0.75	30 ms	3
Entity	0.95	0.85	40 ms	5
Hybrid	0.97	0.88	60 ms	5

BM25 significantly outperformed the dense approach in completeness (+23% relative to dense), indicating the importance of exact token matching in B2B scenarios.

Entity-based retrieval provided a significant increase in completeness (+46% relative to dense), reaching . The most significant growth was observed in queries containing:

- Precise model specification.
- Numerical technical characteristics.
- SKU identifiers.

The Hybrid architecture provided maximum quality ($Recall@10 = 0.97$) with a moderate increase in latency (up to 60 ms). Thus, adding dense reranking after structural filtering allows for high recall while improving ranking within the entity-consistent set.

IX. Error Analysis

A. Dense Retrieval

Primary errors included:

- Substitution of similar models (e.g., difference in one digit).
- Smoothing of numerical characteristics (2800 W vs 3000 W).
- Ignoring SKU numbers as "noise" tokens.

These errors directly illustrate the loss-mismatch between the continuous similarity function and the discrete target relevance function.

B. Entity-based Retrieval

Errors were primarily technical in nature:

- Incorrect normalization of units of measurement.
- Incomplete extraction of the model from the query.

C. Hybrid

Errors were limited to rare long-tail scenarios:

- Rare brands not in the dictionary.
- Non-standard abbreviations.
- Multi-component composite queries.

X. Experimental Conclusions

The results confirm the initial hypothesis: continuous semantic embeddings in isolation do not provide the required search Recall in B2B DIY catalogs. Entity-oriented filtering eliminates systematic errors of dense retrieval, while a hybrid architecture allows for the combination of discrete rigor and semantic flexibility, achieving maximum quality within industrially acceptable latency limits.

XI. Discussion

A. Why Embeddings Lose Recall

An embedding model optimizes a continuous similarity function in Euclidean or spherical space, minimizing the average distance between relevant pairs. Formally, the

empirical risk of the form$E_{(q,d^+)}, L\big(s(q,d^+)\big)$ is minimized, where $s(q,d)$ is a continuous similarity function. However, the target relevance function in B2B scenarios is discrete:

$$Rel(q,d) \in \{0,1\},$$

where in most cases $|R(q)| = 1$. Consequently, the task reduces to the exact retrieval of a single element from the set. Optimizing similarity leads to invariance across numerical values and identifiers if their contribution to the average loss function is statistically small. The vector space tends to smooth local differences to minimize global error. In the context of industrial search services, this manifests in several ways: Close numerical characteristics become nearly indistinguishable. SKU numbers and model identifiers are often interpreted as noise tokens.

A fundamental conflict arises between continuous approximation and the discrete retrieval objective. As observed in the implementation of the similarity service, this geometric limitation necessitates retrieving a significantly larger candidate pool (e.g., 1024 items) via the HNSW index to ensure the target item is present before applying deterministic filters. Thus, the loss of recall is not a random error but a direct consequence of the geometric properties of the embedding space and the chosen optimization objective.

B. Why Entity Matching Provides High Recall

The entity-oriented approach is based on discrete matching checks of identifiers and attributes. Its properties differ fundamentally from dense retrieval:

- **Determinism:** matching a model or article number is uniquely determined.
- **Direct Identifier Verification:** there is no approximation through a continuous metric.
- **Structural Explainability:** every ranking factor is transparent and explainable.

Formally, if , the probability of relevance approaches unity: making this component a powerful anchor in the retrieval architecture. It is the deterministic nature of entity verification that explains the sharp increase in compared to the pure embedding approach.

C. Engineering Constraints

The architecture was developed considering production constraints:

- Latency ms for catalogs with SKUs.
- Index memory consumption limits.
- Horizontal scalability.

Dense retrieval requires ANN structures and increases memory consumption. Entity-based filtering, conversely, allows for reducing the search space to a structurally consistent subset:

The hybrid architecture ensures a balance between quality and computational efficiency. Based on the study, the following recommendations are formulated for B2B catalogs of technical products:

1. Ensure Recall through entity matching before applying semantic models.
2. Use dense retrieval primarily for reranking within structurally filtered sets.
3. Monitor Recall metrics separately from precision, as a reduction in recall in B2B leads to direct business losses.
4. Explicitly account for the difference between semantic search tasks and precise entity extraction tasks.

XII. CONCLUSION

This work demonstrates that embedding-based retrieval is insufficient for B2B DIY catalogs when retrieval Recall is prioritized. It has been experimentally shown that increases from to when transitioning to a hybrid architecture with entity anchoring.

The results confirm the hypothesis of a fundamental conflict between continuous similarity optimization and the discrete nature of relevance in precise product search tasks. The key conclusion is as follows: in B2B tasks, the retrieval architecture must be built around entities (models, article numbers, technical specifications), while semantic similarity should play a supporting role.

Future research directions include:

- Graph models of entities and their relationships.
- Use of LLMs for generating and normalizing synonyms.
- Learning-to-rank on top of entity-filtered candidates.
- Formalizing the task as structured prediction with discrete constraints.

REFERENCES

[1] Krasnov F. V. Embedding-based retrieval: measures of threshold recall and precision to evaluate product search // Business Informatics. – 2024. – Vol. 18. – No. 2. – P. 22–34.

[2] Krasnov F., Kurushin F., Mogilevich E. Custom shared encoder for enhanced recall in e-commerce product search task // Second International Conference on Computing, Machine Learning, and Data Science (CMLDS 2025). – SPIE, 2025. – Vol. 13730. – P. 84–91.

[3] Krasnov F. V. Improving recall and precision of product search on online marketplaces // Applied Informatics. – 2024. – Vol. 19. – No. 2. – P. 118–136.

[4] Gan Y. et al. Binary embedding-based retrieval at Tencent // Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. – 2023. – P. 4056–4067.

[5] Li S. et al. Embedding-based product retrieval in

Taobao Search // Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. – 2021. – P. 3181–3189.

[6] Weller O. et al. On the theoretical limitations of embedding-based retrieval // arXiv preprint arXiv:2508.21038. – 2025.

[7] Lin J. et al. Enhancing relevance of embedding-based retrieval at Walmart // Proceedings of the 33rd ACM International Conference on Information and Knowledge Management. – 2024. – P. 4694–4701.

[8] Ren Z. et al. Information Discovery in E-commerce // Foundations and Trends in Accounting. – 2024. – Vol. 18. – No. 4–5. – P. 417–690.

[9] Schütze H., Manning C. D., Raghavan P. Introduction to Information Retrieval. – Cambridge: Cambridge University Press, 2008. – P. 234–265.

[10] Azzopardi L., De Rijke M., Balog K. Building simulated queries for known-item topics: an analysis using six European languages // Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. – 2007. – P. 455–462.

[11] Park L. A. F. Confidence intervals for information retrieval evaluation // ADCS 2010. – 2010. – P. 97.

[12] Hull D. Using statistical testing in the evaluation of retrieval experiments // Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. – 1993. – P. 329–338.

[13] Robertson S. E. The probability ranking principle in IR // Journal of Documentation. – 1977. – Vol. 33. – No. 4. – P. 294–304.

[14] Schroff F., Kalenichenko D., Philbin J. FaceNet: A unified embedding for face recognition and clustering // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. – 2015. – P. 815–823.

[15] Karpukhin V. et al. Dense passage retrieval for open-domain question answering // Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). – 2020. – P. 6769–6781.

[16] Malkov Y. A., Yashunin D. A. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 2018. – Vol. 42. – No. 4. – P. 824–836.

[17] Aumüller M., Bernhardsson E., Faithfull A. ANN-Benchmarks: A benchmarking tool for approximate nearest neighbor algorithms // Information Systems. – 2020. – Vol. 87. – P. 101374.

[18] Devlin J. et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding // NAACL. – 2019.

[19] Warner B. et al. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory-efficient, and long-context fine-tuning and inference // Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). – 2025. – P. 2526–2547.

[20] Johnson J. et al. Billion-scale similarity search with GPUs // arXiv preprint. – 2019.

[21] Vardanyan A. USearch by Unum Cloud: software. Version 2.24.0. – 2023. – Available at: https://github.com/unum-cloud/usearch (accessed: 21.02.2026). DOI: 10.5281/zenodo.7949416.

Проблема полноты поиска в B2B-каталогах DIY-товаров: ограничения семантических эмбеддингов и сущностно-ориентированный подход

Ф. Краснов

Аннотация -- В статье рассматривается задача обеспечения высокой полноты поиска в B2B-каталогах DIY-товаров (инструменты и материалы для строительства и ремонта). На практике B2B-поиск предъявляет существенно более строгие требования к полноте по сравнению с B2C-рекомендательными сценариями, поскольку значительная доля запросов носит характер known-item retrieval и связана с точными артикулами, моделями и техническими параметрами. Анализируется эволюция архитектуры поиска: от чистого dense retrieval на базе трансформерных эмбеддингов (ModernBERT, triplet loss) к гибридной системе, основанной на детекции сущностей (бренд, модель, технические характеристики) и структурированном сопоставлении (entity matching).

Показано, что embedding-based retrieval не гарантирует полноту вследствие сглаживания числовых токенов, семантической компрессии идентификаторов и ограничений ANN-поиска. Геометрические свойства embedding-пространства приводят к инвариантности по отношению к локальным дискретным различиям, критичным для B2B-сценариев. Предложенная архитектура anchor-based retrieval обеспечивает значительный рост Recall@10 (с 0.65 до 0.97 на реальных данных) при сохранении приемлемой латентности и высокой объяснимости.

Экспериментальная часть выполнена на промышленном каталоге объёмом более 10 млн позиций и выборке из 5 тыс. реальных B2B-запросов. Работа ориентирована на практическое применение и может быть использована при проектировании поисковых систем в сегменте B2B e-commerce.

Ключевые слова -- B2B-поиск, DIY-товары, полнота поиска, recall, dense retrieval, entity matching, гибридные архитектуры.

Литература:

1. Краснов Ф. В. Embedding-based retrieval: measures of threshold recall and precision to evaluate product search //Бизнес-информатика. – 2024. – Т. 18. – №. 2. – С. 22-34.
2. Krasnov F., Kurushin F., Mogilevich E. Custom shared encoder for enhanced recall in e-commerce product search task //Second International Conference on Computing, Machine Learning, and Data Science (CMLDS 2025). – SPIE, 2025. – Т. 13730. – С. 84-91.
3. Краснов Ф. В. Повышение полноты и точности поиска товаров на торговых интернет-площадках //ПРИКЛАДНАЯ ИНФОРМАТИКА Учредители: Московский университет"Синергия". – 2024. –Т. 19. – №. 2. – С. 118-136.
4. Gan Y. et al. Binary embedding-based retrieval at Tencent //Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. – 2023. – С. 4056-4067.
5. Li S. et al. Embedding-based product retrieval in taobao search //Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. – 2021. – С. 3181-3189.
6. Weller O. et al. On the theoretical limitations of embedding-based retrieval //arXiv preprint arXiv:2508.21038. – 2025.
7. Lin J. et al. Enhancing relevance of embedding-based retrieval at walmart //Proceedings of the 33rd ACM International Conference on Information and Knowledge Management. – 2024. – С. 4694-4701.
8. Ren Z. et al. Information Discovery in E-commerce //Foundations and Trends in Accounting. – 2024.– Т. 18. – №. 4-5. – С. 417-690.12
9. Schütze H., Manning C. D., Raghavan P. Introduction to information retrieval. – Cambridge : Cambridge University Press, 2008. – Т. 39. – С. 234-265.
10. Azzopardi L., De Rijke M., Balog K. Building simulated queries for known-item topics: an analysis using six european languages //Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval. – 2007. – С. 455-462.
11. Park L. A. F. Confidence intervals for information retrieval evaluation //ADCS 2010. – 2010. – С. 97.
12. Hull D. Using statistical testing in the evaluation of retrieval experiments //Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval. –1993. – С. 329-338.
13. Robertson S. E. The probability ranking principle in IR //Journal of documentation. – 1977. – Т. 33. –№. 4. – С. 294-304.
14. Schroff F., Kalenichenko D., Philbin J. Facenet: A unified embedding for face recognition and clustering //Proceedings of the IEEE conference on computer vision and pattern recognition. –

2015. – С. 815-823.
15. Karpukhin V. et al. Dense passage retrieval for open-domain question answering //Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP). – 2020. – С. 6769-6781.
16. Malkov Y. A., Yashunin D. A. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs //IEEE transactions on pattern analysis and machine intelligence. – 2018. – Т. 42. – №. 4. – С. 824-836.
17. Aumüller M., Bernhardsson E., Faithfull A. ANN-Benchmarks: A benchmarking tool for approximate nearest neighbor algorithms //Information Systems. – 2020. – Т. 87. – С. 101374.
18. Devlin J. et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. NAACL, 2019.
19. Warner B. et al. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference //Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). – 2025. – С. 2526-2547.
20. Johnson J. et al. Billion-scale similarity search with GPUs. arXiv, 2019.
21. Варданян А. USearch by Unum Cloud : программное обеспечение. Версия 2.24.0. 2023. URL:https://github.com/unum-cloud/usearch (дата обращения: 21.02.2026). DOI: 10.5281/zenodo.7949416.