A Hybrid Framework for Conversion-Optimized Query Suggestion in e-Commerce Search

Melis Umut Öztürk, Mustafa Keskin, Selçuk Açıkalın and Halil İbrahim Ergül
Hepsiburada, Turkey

Keywords: Query Suggestion, Related Search Terms, e-Commerce Search, Semantic Retrieval Category Prediction, Hybrid Ranking, User Behavior Modeling, Conversion Optimization.

Abstract: Efficient query suggestion is a key component of modern e-commerce search systems, as it helps customers refine vague or underspecified queries and discover relevant products more effectively. In this work, we present a two-stage retrieval-and-ranking framework for generating related search terms at scale. Our method combines semantic retrieval based on a Turkish BERT encoder with catalog-aware candidate generation constrained by product taxonomy and business rules. Candidate sets are merged, filtered, and ranked using a hybrid scoring function that balances query frequency and conversion rate (CR), while ensuring policy compliance such as brand-safety constraints. We evaluate the system using both offline and online experiments. Offline coverage analysis demonstrates the strength of category-based candidates, while business-critical metrics such as click-through rate (CTR), and CR consistently favor the hybrid scoring method. Large-scale A/B testing further confirms statistically significant improvements in conversion-related KPIs, validating the real-world impact of our approach. The framework has been deployed in a production e-commerce environment, where it enhances search experience, drives engagement, and improves commercial outcomes.

1 INTRODUCTION

The implementation of the related search term method holds substantial importance in the realm of e-commerce, where efficient and effective search functionalities are paramount. As online shopping continues to grow, the ability to quickly and accurately connect customers with relevant products becomes a critical factor in maintaining a competitive advantage. This method addresses the inherent challenges of search query formulation, offering a strategic enhancement to traditional search systems.

In the dynamic landscape of e-commerce, where product catalogs are vast and diverse, customers often encounter difficulties in pinpointing exactly what they are looking for. The related search term method serves as a vital tool in mitigating these challenges by providing intelligent suggestions that guide users towards more precise and meaningful search results. This not only improves the user experience but also fosters a deeper engagement with the platform.

Moreover, the integration of this method into e-commerce platforms can lead to significant business benefits. By facilitating better search outcomes, it potentially increases conversion rates and average order values, as customers are more likely to find and purchase products that meet their needs. Additionally, the method enhances customer satisfaction and loyalty, as users appreciate the ease and efficiency of finding relevant products. Thus, the related search term method is an essential component in the toolkit of modern e-commerce strategies, driving both customer and business success.

The development of the related search term method was driven by the need to enhance the customer experience during the search process on e-commerce platforms. Customers often face challenges in articulating their search queries effectively, leading to suboptimal search results. By suggesting related search terms, the system aims to bridge this gap, providing users with alternative queries that may better capture their intent.

This method contributes significantly to improving search accuracy. Customers are presented with suggestions that align closely with their initial search, increasing the likelihood of discovering relevant products. This not only enhances user satisfaction but also reduces the time and effort required to find desired items, making the shopping experience more efficient and enjoyable.

Umut Öztürk, M., Keskin, M., Açıkalın, S. and Ergül, H. İ.
A Hybrid Framework for Conversion-Optimized Query Suggestion in e-Commerce Search.
DOI: 10.5220/0014287300004848

In *Proceedings of the 2nd International Conference on Advances in Electrical, Electronics, Energy, and Computer Sciences (ICEEECS 2025)*, pages 31-37
ISBN: 978-989-758-783-2

Furthermore, the related search term approach supports customers in exploring the product catalog more comprehensively. By exposing users to a broader range of relevant queries, the system encourages the discovery of products that may not have been initially considered. This can lead to increased engagement and potentially higher conversion rates, benefiting both customers and the e-commerce platform. The method, therefore, plays a crucial role in optimizing the search functionality, ultimately driving customer retention and loyalty.

2 RELATED WORKS

Research on related search terms also referred to as query suggestion, query auto-completion (QAC), query reformulation, or query expansion spans classical IR methods, graph- and log-mining approaches, neural sequence models, and large-scale production systems. Below, we synthesize the literature most relevant to building and evaluating a system that surfaces related search terms.

2.1 Query Suggestion from Logs and Graphs

Early approaches exploit large-scale query and click logs to model transitions between queries and to mine semantically related alternatives. The query-flow graph (QFG) represents queries as nodes and session-based transitions as edges; random walks or edge weights yield suggestions, (Boldi et al., 2009), (Boldi et al., 2008), (Bai et al., 2011). A complementary line uses query–document click bipartite graphs with hitting time to balance semantic similarity and tail coverage, (Mei et al., 2008). These methods established that sequence context and click feedback enable high-quality related suggestions beyond co-occurrence signals.

Auto-complete and prefix-sensitive variants extend suggestion to the character-level prefix setting, where candidate generation and ranking must be real-time. Classical works show that adding recent-query context substantially improves short-prefix predictions, (Bar-Yossef and Kraus, 2011). Surveys summarize heuristic and learning-to-rank families, temporal drift handling, and personalization, (Cai and de Rijke, 2016).

2.2 Context-Aware and Neural Sequence Models

Generative neural models capture multi-query session context to suggest the next reformulation. The hierarchical recurrent encoder–decoder (HRED) conditions on the sequence of prior queries, outperforming pairwise methods on next-query prediction, (Sordoni et al., 2015). Subsequent work explores deep language models for low-latency QAC, (Wang et al., 2018) and integrates temporal/user features in learning-to-rank frameworks (e.g., Hawkes/Markov processes and user models), (Li et al., 2017), (Li et al., 2015), (Kharitonov et al., 2013), (Cai et al., 2016).

2.3 Query Expansion and Reformulation

To mitigate vocabulary mismatch, query expansion augments the original query using pseudo-relevance feedback (PRF) and relevance models. Relevance-based language models (RM1/RM3) and PRF remain strong baselines, (Lavrenko and Croft, 2001), (Abdul-Jaleel et al., 2004; Carpineto and Romano, 2012; Zamani and Croft, 2011). Neural document/query expansion further improves first-stage recall: doc2query and docTTTTTquery expand documents with predicted queries using sequence-to-sequence models, boosting downstream ranking (Nogueira et al., 2019; Nogueira and Lin, 2019). Recent studies revisit PRF with modern embeddings and classification signals (Lin et al., 2019; Wang et al., 2022). While expansion targets retrieval effectiveness rather than UI suggestions, the same candidates are valuable as related search or people also search for terms.

2.4 Evaluation and Objectives

Offline evaluation commonly relies on historical logs with held-out user actions, measuring top-k acceptance or reformulation success; user-model–based metrics can better reflect utility in QAC/QS, (Kharitonov et al., 2013; Cai and de Rijke, 2016). For related-term widgets (”People also search for”), diversity and intent coverage are important; diversification for QAC has been studied explicitly, (Cai et al., 2016). Counterfactual learning-to-rank for suggestions from implicit feedback is increasingly relevant to mitigate presentation bias (see surveys and bandit LTR literature referenced therein).

2.5 Industry Practice and Product Considerations

Public posts from Google and Bing describe autosuggest/related-search systems trained on real user queries, leveraging freshness, location, and personalization signals, with policies to remove unsafe or low-quality predictions, (Google, 2020; Google, 2018; Bing Search Blog, 2013; Bing Search Quality Insights, 2020; Inside Search (Google), 2011). Historical posts also reveal how related-search and instant suggestions influence impression counting and UX, (Google, 2010). These accounts highlight non-technical constraints latency, safety, and abuse prevention that shape production-ready related-term systems.

3 METHODOLOGY

We propose a two-stage retrieval-and-ranking framework to generate related search terms for a large-scale e-commerce search system. The approach blends semantic retrieval using a domain-appropriate Turkish BERT encoder with a catalog-aware retrieval based on predicted query categories and hierarchical taxonomy constraints, then fuses, filters, and ranks the combined candidates. The design explicitly integrates behavioral signals CTR and CR and business rules (e.g., brand-only constraints and blacklists) to improve utility and governance. The entire pipeline is executed in batch over the most recent one-month interaction window and produces a compact set of six suggestions per query, suitable for direct UI rendering. As seen in Figure 1 two different prediction algorithms are used in the pipeline.

3.1 Dataset Construction

We extract all user search queries from the last one month of production logs. Queries are normalized to lowercase to reduce sparsity caused by case variants. We then select the top 1,000,000 unique queries by occurrence frequency (impressions) to ensure sufficient behavioral signal for learning and ranking. For model training, we used one month of query and interaction logs, while the subsequent one-week period was reserved for testing. This setup allowed us to capture sufficient behavioral signals for learning while ensuring a temporally disjoint evaluation window.

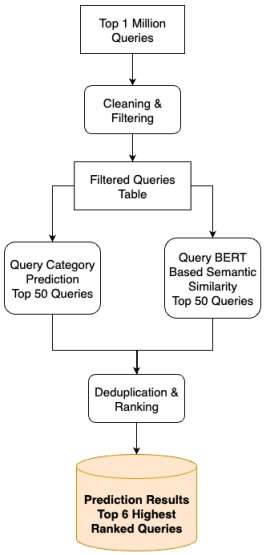

Figure 1: Overview of the related search term pipeline. The pipeline begins with the collection of the top one million user-generated search queries. These queries undergo preprocessing steps, including the application of a Click-Through Rate (CTR) filter, to ensure data quality. Two predictive models are then utilized to generate a base pool of related search terms. Finally, deduplication and reranking processes are applied to refine the selection, reducing the options to the top six queries for final display.

3.1.1 Text Normalization and Cleaning

We apply standard text cleaning suitable for Turkish query logs, including lowercasing, whitespace normalization, and lightweight punctuation handling. We retain a normalized string per query as the canonical key across downstream components.

3.1.2 Behavioral Metrics: CTR and CR

For each query, we compute CTR and CR from the same one-month window. CTR is defined as clicks divided by impressions. CR is defined as purchases divided by clicks, in line with standard e-commerce practice. These metrics provide complementary views of engagement (CTR) and downstream value (CR). They are later used for filtering and ranking.

Queries with zero CTR are removed to avoid surfacing terms with no evidence of user engagement and to reduce noise in the embedding space. This eliminates approximately 100k queries from the initial pool, leaving on the order of 0.9M queries for retrieval.

We compile a lexicon of top-N brands by sales volume to detect brand presence in queries. Queries containing any of these brand tokens are flagged. We distinguish:

- Brand-only queries: the query is a brand name without additional qualifiers.
- Mixed queries: the query includes a brand token plus other terms.

This brand flag is later used to enforce business constraints on taxonomy depth in recommendations.

Each query is assigned a predicted product category using our internal category prediction system, which outputs a probability distribution over catalog categories. We take the top-1 category label per query for downstream catalog-based retrieval.

We reference the catalog's hierarchical levels (e.g., level2/level3/level4/level5) in later steps to control the allowed depth of related terms.

3.2 Candidate Generation Methods

3.2.1 Semantic Candidate Retrieval

We encode each cleaned query using a Turkish BERT model suitable for sentence-level representations. Sentence embeddings are obtained with a standard pooling strategy and used as fixed-length vector representations. Embeddings are computed for all queries retained after the CTR filter.

We build an HNSW (Hierarchical Navigable Small World) index over the query embeddings to enable scalable approximate nearest neighbor search under a vector similarity metric (cosine similarity). HNSW offers sublinear retrieval latency and supports dynamic exploration parameters to balance recall and speed.

For each query, we retrieve the top-50 most similar queries by vector similarity from the HNSW index. This yields semantically proximate candidates that capture paraphrases and closely related intents, even when lexical overlap is low.

3.2.2 Query Category Prediction Based Candidate Generation

Using the top-1 predicted category for each query, we leverage the catalog hierarchy to constrain and prioritize related terms. For each category node, we precompute up to the top 100 most frequent queries within that category from the same one-month window. We assign these category-top queries as candidates for any query mapped to that category, resulting in a catalog-coherent candidate set that reflects user co-search behavior.

If the original search is brand-only, we restrict related terms to the level2 level, per business policy. This limits breadth and avoids surfacing overly specific subcategories where brand-only intents are typically exploratory or navigational. For non-brand queries, related terms may originate from deeper levels (h3/h4/h5), allowing more specific, intent-refining suggestions.

3.3 Merging Retrieval Outputs

We take the set union of semantic (BERT/HNSW) and catalog-based candidates for each query. We deduplicate candidates to remove overlaps between the two retrieval streams. We attach per-query CTR and CR to all candidate records so they can be used for scoring and tie-breaking. We also retain original frequency counts.

We materialize the merged candidate set to BigQuery, where we apply a curated blacklist (e.g., policy-sensitive terms, low-quality patterns). We drop intermediate tables and nonessential columns after filtering to reduce storage and memory footprint and to streamline downstream ranking.

3.4 Ranking and Selection

We rank candidates with a hybrid score that balances popularity and value:

$$\text{score} = w_1 \cdot \text{frequency} + w_2 \cdot \text{CR} \quad (1)$$

The weights w1 and w2 are configurable. In this study different weights are used based upon business needs to evade any bias of variables on the score. In practice, they capture a trade-off between surfacing queries that are broadly useful (frequency) and those that are more likely to lead to purchase (CR). To ensure comparability, features are standardized or normalized as appropriate before combining. CTR is available for analysis and guardrails but is not explicitly part of the final score when using the frequency–CR hybrid; it can be incorporated if desired.

Because the UI allocates six slots for related searches, we select the top-6 candidates per query after ranking, subject to previously described business rules and blacklists. This produces a compact, high-quality set of recommendations per originating query.

3.5 Policy Objective

- Prevent cross-brand substitution suggestions within the same product family when the source query is brand-bearing. Example: for a brand-bearing source "asus" or "asus laptop," do not return "macbook" (Apple) or other competing brands in the Laptops family.

- Allow brandless generic refinements (e.g., “asus laptop” → “gaming laptop,” “lightweight laptop”) and intra-brand variants (e.g., “asus zenbook”).
- Permit exceptions only where explicitly sanctioned (e.g., accessories, cross-compatibility domains), based on business rules.

4 EVALUATION

The effectiveness of the proposed query suggestion framework was assessed through both offline and online experiments. The offline evaluation focused on intrinsic performance metrics such as coverage, while the online evaluation measured the impact on user behavior and business metrics through large-scale A/B testing.

4.1 Offline Evaluation

4.1.1 Ground Truth Construction

For the construction of ground-truth related queries, clickstream data was utilized. For each query, the set of products clicked after submission was identified. Queries leading to clicks on overlapping products were grouped together, forming pools of semantically related queries. These pools served as proxies for true related search terms.

4.1.2 Coverage Metric

Coverage was defined as the proportion of system-generated suggestions overlapping with the click-derived related terms. For each query, the top-N (N=50, N=6) related terms were extracted from different models, and the overlap with the click-derived pools was computed as coverage.

4.1.3 Methods Compared

Three approaches were evaluated:

- Category-based suggestions (CP), where related queries were generated using category prediction outcomes.
- BERT-based embeddings (SS), where related queries were identified via semantic similarity in the embedding space.
- Hybrid scoring (HS), which combined category-based and BERT-based similarity scores through a weighted scheme.

4.1.4 Results and Analysis

- Coverage Results: Category-based suggestions achieved the highest coverage, outperforming both BERT-based and hybrid approaches when evaluated solely on overlap with click-derived related terms. This finding held consistently for both top-50 and top-6 suggestions.
- Business Metrics (CR & CTR): Since coverage alone does not fully capture business value, average Conversion Rate (Avg. CR) and average Click-Through Rate (Avg. CTR) were also examined. To ensure confidentiality, all KPI values were normalized and indexed at 100 as shown in Table 1.
  - For the top-50 related terms, the hybrid method yielded CR improvements of +7.10 over BERT and +4.27 over category-based.
  - For the top-6 related terms, the hybrid method achieved CR improvements of +4.75 over BERT and +4.55 over category-based.
  - CTR results similarly showed the hybrid method outperforming both baselines.

Although the category-based method delivered the highest coverage (COV), the hybrid method achieved superior performance on business-critical KPIs (CR and CTR). Based on these outcomes, the hybrid method was selected for live testing.

Table 1: Comparison of results for top-50 and top-6 related terms.

Metric	Top-50 Value	Top-6 Value
COV for SS	0.21	0.07
COV for CP	0.49	0.34
COV for HS	0.38	0.12
Avg. CR for SS	100.00	100.00
Avg. CR for CP	102.83	100.20
Avg. CR for HS	107.10	104.75
Avg. CTR for SS	100.00	101.97
Avg. CTR for CP	124.13	100.00
Avg. CTR for HS	152.07	104.55

4.2 Online Evaluation (A/B Testing)

4.2.1 Experimental Setup

An A/B test was conducted in the production environment to validate the offline results. Users were randomly assigned to control (baseline system) and treatment (hybrid scoring method) groups. The test was run for 12 days.

- Listing Search Conversion Rate: Ratio of completed purchase transactions from listing searches.

- Listing Click CR for Search: Ratio of completed purchase transactions to the total number of product clicks from search listings.
- Overall Conversion Rate: Overall conversion across sessions influenced by search.

4.2.2 Results

The hybrid scoring method consistently improved the monitored metrics:

- Listing Search CR: +0.42 (p-value = 0.07, marginal significance)
- Listing Click CR for Search: +0.43
- Overall Conversion Rate: +0.39

Online results confirmed that hybrid scoring not only enhanced user engagement with suggested queries but also led to measurable business gains.

4.3 Deployment Decision

Given the superior CR and CTR observed in offline evaluation and the statistically significant improvements achieved during the A/B test, the hybrid scoring method was promoted to production. This decision was supported by both quantitative coverage analysis and business KPI validation, establishing the method as a robust enhancement to the e-commerce search experience.

5 CONCLUSION

This work introduced a two-stage retrieval-and-ranking framework for generating related search terms in a large-scale e-commerce environment. By integrating semantic retrieval using a Turkish BERT encoder with catalog-based candidate generation guided by taxonomy constraints, the system produces high-quality suggestions that align with both user intent and business objectives. Offline experiments demonstrated that while category-based candidates provided stronger coverage, the hybrid scoring method delivered superior performance on business-critical metrics such as CTR and CR. Online A/B testing further validated these findings, showing statistically significant improvements in conversion-related KPIs.

The deployment of the proposed framework in production confirms its practical effectiveness it enhances user experience by surfacing relevant refinements, supports product discovery across the catalog, and drives measurable gains in engagement and revenue. Overall, this study underscores the value of combining semantic and behavioral signals with domain-specific business rules in building robust, production-ready related search systems for e-commerce.

6 FUTURE WORKS

Future research can explore several promising directions to enhance the proposed query suggestion framework. Since the BERT-based model that accounts for semantic context showed relatively lower performance compared to the category-based method, testing alternative embedding models, particularly more recent transformer architectures or domain-specific embeddings trained on e-commerce data could, provide improvements. In addition, the hybrid scoring approach currently relies on manually determined weights; these could be optimized through a regression-based or learning-to-rank method, allowing data-driven calibration of component contributions and potentially yielding higher effectiveness. Furthermore, the coverage metric in this study was based on labels derived from user interaction events such as clicks and orders. Alternative strategies for label construction, for instance using large language models or advanced query clustering techniques, may offer richer ground truth and allow deeper insights into system performance.

ACKNOWLEDGEMENTS

This project was made possible by the individual contributions of each member of the recommendation team within Hepsiburada technology group. Also, this project would not have been possible if the technology group management of Hepsiburada had not supported and encouraged the data science team in innovation.

REFERENCES

Abdul-Jaleel, N., Allan, J., Croft, W. B., Diaz, F., et al. (2004). Umass at trec 2004: Novelty and hard. In *Proceedings of TREC*.

Bai, L. et al. (2011). Query recommendation by modelling the query-flow graph. In *Advances in Information Retrieval, ECIR*.

Bar-Yossef, Z. and Kraus, N. (2011). Context-sensitive query auto-completion. In *Proceedings of the 20th International Conference on World Wide Web (WWW)*, pages 107–116.

Bing Search Blog (2013). A deeper look at auto-suggest. https://blogs.bing.com/search/March-2013/A-Deeper-Look-at-Autosuggest.

Bing Search Quality Insights (2020). Introducing the next wave of ai at scale innovations in bing. https://blogs.bing.com/search-quality-insights/september-2020/Introducing-the-next-wave-of-AI-at-Scale-innovations-in-Bing.

Boldi, P., Bonchi, F., Castillo, C., Donato, D., and Vigna, S. (2009). Query suggestions using query-flow graphs. In *Proceedings of the Workshop on Web Search Click Data (WSCD)*.

Boldi, P., Bonchi, F., Castillo, C., and Vigna, S. (2008). The query-flow graph: Model and applications. In *Proceedings of the 17th International World Wide Web Conference (WWW) Workshop on Web Search Click Data*.

Cai, F. and de Rijke, M. (2016). A survey of query auto completion in information retrieval. *Foundations and Trends in Information Retrieval*, 10(4):273–363.

Cai, F., Reinanda, R., and de Rijke, M. (2016). Diversifying query auto-completion. *ACM Transactions on Information Systems*, 34(4):Article 25.

Carpineto, C. and Romano, G. (2012). A survey of automatic query expansion in information retrieval. *ACM Computing Surveys*, 44(1):1–50.

Google (2010). Google instant. Google Blog (Official announcement). Introduced real-time search results as users type (launched September 8, 2010).

Google (2018). How google autocomplete works in search. Google Keyword/Products Blog. https://blog.google/products/search/how-google-autocomplete-works-search/.

Google (2020). How google autocomplete predictions work. Google Keyword/Products Blog. https://blog.google/products/search/how-google-autocomplete-predictions-work/.

Inside Search (Google) (2011). Organizing lists of related searches. https://search.googleblog.com/2011/06/organizing-lists-of-related-searches_16.html.

Kharitonov, E., Macdonald, C., Serdyukov, P., and Ounis, I. (2013). User model-based metrics for offline query suggestion evaluation. In *Proceedings of SIGIR*.

Lavrenko, V. and Croft, W. B. (2001). Relevance-based language models. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 120–127.

Li, L., Deng, H., Chen, J., and Chang, Y. (2017). Learning parametric models for context-aware query auto-completion via hawkes processes. In *Proceedings of WSDM*.

Li, L., Deng, H., Dong, A., Chang, Y., Baeza-Yates, R., and Zha, H. (2015). Analyzing user's sequential behavior in query auto-completion via markov processes. In *Proceedings of SIGIR*.

Lin, J. et al. (2019). Pseudo-relevance feedback using text classification. *arXiv preprint arXiv:1904.08861*.

Mei, Q., Zhou, D., and Church, K. (2008). Query suggestion using hitting time. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM)*.

Nogueira, R. and Lin, J. (2019). From doc2query to doctttttquery. *arXiv preprint arXiv:1910.10683*.

Nogueira, R., Yang, W., Lin, J., and Cho, K. (2019). Document expansion by query prediction. *arXiv preprint arXiv:1904.08375*.

Sordoni, A., Bengio, Y., Vahabi, H., Lioma, C., Simonsen, J. G., and Nie, J.-Y. (2015). A hierarchical recurrent encoder-decoder for generative context-aware query suggestion. In *Proceedings of the 24th ACM International Conference on Information and Knowledge Management (CIKM)*.

Wang, P.-W. et al. (2018). Realtime query completion via deep language models. In *International Conference on Learning Representations (ICLR) Workshop*.

Wang, X. et al. (2022). Improving zero-shot retrieval using dense external expansion. *Information Processing & Management*, 59(6).

Zamani, H. and Croft, W. B. (2011). Pseudo-relevance feedback based on matrix factorization. In *CIIR Technical Report IR-1010*.