Search Query Embeddings via User-behavior-driven Contrastive Learning

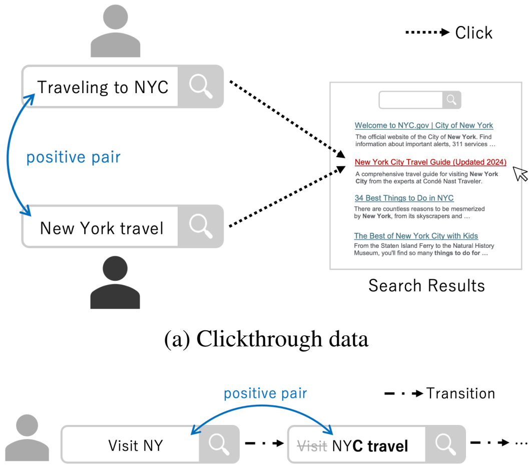
Sosuke Nishikawa[*] **Jun Hirako**[*] **Nobuhiro Kaji** **Koki Watanabe**
Hiroki Asano **Souta Yamashiro** **Shumpei Sano**
LY Corporation
`{sonishik,jhirako,nkaji,kokwatan,hiroasan,soyamash,shsano}@lycorp.co.jp`

Abstract

Universal query embeddings that accurately capture the semantic meaning of search queries are crucial for supporting a range of query understanding (QU) tasks within enterprises. However, current embedding approaches often struggle to effectively represent queries due to the shortness of search queries and their tendency for surface-level variations. We propose a user-behavior-driven contrastive learning approach which directly aligns embeddings according to user intent. This approach uses intent-aligned query pairs as positive examples, derived from two types of real-world user interactions: (1) clickthrough data, in which queries leading to clicks on the same URLs are assumed to share the same intent, and (2) session data, in which queries within the same user session are considered to share intent. By incorporating these query pairs into a robust contrastive learning framework, we can construct query embedding models that align with user intent while minimizing reliance on surface-level lexical similarities. Evaluations on real-world QU tasks demonstrated that these models substantially outperformed state-of-the-art text embedding models such as mE5 and SimCSE. Our models have been deployed in our search engine to support QU technologies.

1 Introduction

Query understanding (QU) tasks, such as query classification and suggestion, play a crucial role in improving user search experiences by interpreting users' search intents and supporting search behavior (Shneiderman et al., 1997; Lau and Horvitz, 1999). Embedding-based approaches have gained prominence in addressing these tasks due to their robustness to lexical variations (Zhang et al., 2019). Building tailored embeddings for every QU task is costly, making universal query embeddings essential. Such universal embeddings enable accurate representation of search intent and provide a versatile solution applicable across QU tasks, which is highly valuable for enterprises.


Click
Traveling to NYC
positive pair
New York travel
Welcome to NYC.gov | City of New York
The official website of the City of New York. Find information about important alerts, 311 services …
New York City Travel Guide (Updated 2024)
A comprehensive travel guide for visiting New York City from the experts at Condé Nast Traveler.
34 Best Things to Do in NYC
There are countless reasons to be mesmerized by New York, from its skyscrapers and …
The Best of New York City with Kids
From the Staten Island Ferry to the Natural History Museum, you'll find so many things to do for …
Search Results
positive pair
Transition
Visit NY
~~Visit~~ NYC travel
…

(a) Clickthrough data

(b) Session data

Figure 1: Illustrations of user interactions used to construct positive query pairs in UBIQUE.

Despite their importance, developing query embeddings that well reflect users' intent presents unique challenges. Since search queries are typically short, they lack rich contextual information, making it difficult to precisely capture users' search intent (Hashemi, 2016). This shortness also means that minor wording changes in queries, e.g., replacing even a couple of words with their synonyms, can noticeably alter their appearances. For example, "buy car" and "purchase an automobile" express the same intent but differ substantially in wording. These challenges highlight the need to consider suitable learning embedding approaches for search queries.

A widely recognized approach for learning robust text embeddings is contrastive learning, which has demonstrated notable success in this field. State-of-the-art (SOTA) contrastive learning approaches typically use large-scale weak supervision from web sources, such as question-answer

[*] Equal contribution.

Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Industry Track), pages 138–147
April 30, 2025 ©2025 Association for Computational Linguistics

pairs from QA forums or title-passage pairs from encyclopedic articles (Wang et al., 2024a,b; Li et al., 2023b). However, as these datasets primarily consist of longer, contextually detailed sentences, models trained on them struggle to handle short, context-poor queries.

An alternative approach is to use unsupervised contrastive learning models, such as Unsup. SimCSE (Gao et al., 2021), directly on a large corpus of search queries. Unsup. SimCSE generates pseudo-positive examples by encoding the same sentence twice with different dropout noise. A model trained on such positive examples tends to overemphasize lexical overlap as a cue for semantic equivalence; we observed that Unsup. SimCSE struggles to capture semantic similarities between queries with different appearances but the same intent, such as "buy car" and "purchase an automobile" (§5.1).

Overall, current contrastive learning approaches are suboptimal when creating effective positive examples for search query embeddings: typical weakly supervised approaches struggle to generalize to context-poor queries, while the representative unsupervised approach results in models that are overly sensitive to surface-level variations. To address these problems, we propose **U**ser **B**ehavior-driven contrastive learning with **I**ntent alignment for search **QU**ery **E**mbeddings (UBIQUE). UBIQUE directly aligns embeddings according to user intent, using intent-aligned query pairs derived from real-world user interactions as positive examples. As shown in Figure 1, we explore two types of user interactions. (1) **Click-through data** are records of users' clicking on web pages after submitting search queries. Queries are considered to have the same intent if they lead to clicks on the same URL, as users tend to click on results that satisfy similar information needs. (2) **Session data** are sequences of queries a single user takes on a search engine within a given time frame. Queries within the same user session are assumed to share the same search intent. By using a robust contrastive learning framework (Chen et al., 2020) on these intent-aligned query pairs, UBIQUE constructs models that precisely capture the inherent intent of context-poor queries. This approach also minimizes reliance on appearances, as these intent-aligned query pairs are constructed independently of surface-level similarities.

For our experiments, we built four practical QU datasets using real-world search queries to evaluate UBIQUE from multiple perspectives (§4). The results indicate that our click-based model (UBIQUE$_{\text{click}}$) and session-based model (UBIQUE$_{\text{session}}$) substantially outperformed baselines such as mE5$_{\text{large}}$ and Unsup. SimCSE. Specifically, compared to mE5$_{\text{large}}$, UBIQUE$_{\text{click}}$ achieved an average improvement of 8.7 points in task-performance metrics across all tasks, while UBIQUE$_{\text{session}}$ showed strengths in a query-suggestion task, achieving an improvement of 5.3 points in NDCG@10 score. Our analysis also confirmed their robustness to lexical variations, effectively capturing semantic similarities where unsupervised models fail (§5.1). These findings highlight the effectiveness of leveraging user behavior data in learning universal query embeddings.

2 Related Work

Query Understanding QU aims to enhance search experiences by effectively processing user queries (Shneiderman et al., 1997; Lau and Horvitz, 1999). Due to the shortness and challenges in capturing their intent, user behavior logs have traditionally supported each QU task before the emergence of deep learning. For instance, mutual query suggestions have been derived from co-occurring session queries (Huang et al., 2003). Similarly, query classification and clustering have leveraged clicked URLs (Cao et al., 2009; Beeferman and Berger, 2000).

More recently, pre-trained language models have advanced QU. Jiang et al. (2022) mitigated context absence in queries via extended token classification, while Li et al. (2023a) proposed a pre-training framework using a query-URL bipartite graph. We fine-tuned pre-trained language models using user interactions to construct fixed-size text embeddings for general QU tasks. Our approach can be combined with these pre-training techniques.

Closely related is the study by Zhang et al. (2019), who proposed a Bi-GRU-based GEN encoder to compute intent similarity using click-through data and task-specific human annotations. Unlike their method, UBIQUE constructs general-purpose search query embedding models that rely solely on automatically collected user interactions.

Contrastive Learning Contrastive learning has proven effective for learning text embeddings by pulling similar pairs closer and pushing dissimilar pairs apart (Hadsell et al., 2006). Prior research typically focused on constructing positive examples. Early studies relied on annotated datasets, such as the NLI dataset (Gao et al., 2021; Zhang et al., 2021), while more recent studies used large-scale weak supervision from web resources, achieving SOTA results (Wang et al., 2024a,b; Li et al., 2023b). While these datasets consist of longer texts, we focused on handling short-text queries using weak supervision from user interactions.

To reduce reliance on annotated data, unsupervised approaches have also been explored. A prominent example is Unsup. SimCSE, which uses dropout as minimal noise to generate positive pairs (Gao et al., 2021; Liu et al., 2021). While Wu et al. (2022) addressed the length biases inherent in Unsup. SimCSE, we examined its ineffectiveness with search queries, particularly its sensitivity to surface-level variations.

3 UBIQUE

This section introduces UBIQUE for constructing universal query-embedding models.

3.1 Overview

UBIQUE uses query pairs (q, q^+) as positive examples, which match the same search intent, regardless of differences in their surface forms. These pairs are mined from user-interaction logs, which capture detailed records of search activities and engagement patterns with a search engine (§3.2 and §3.3). Given a set of query pairs $D = (q_i, q_i^+)_{i=1}^m$, UBIQUE models are trained using the InfoNCE loss over in-batch negatives (Chen et al., 2017):

$$L_i = -\log \frac{e^{\text{sim}(\mathbf{q_i}, \mathbf{q_i^+})/\tau}}{\sum_{j=1}^{N} e^{\text{sim}(\mathbf{q_i}, \mathbf{q_j^+})/\tau}}, \tag{1}$$

where N denotes the mini-batch size, τ the temperature hyperparameter, $\text{sim}(\cdot)$ the cosine similarity, and $\mathbf{q_i}$ and $\mathbf{q_i^+}$ the embeddings of q_i and q_i^+, respectively. In the following sections, we explain the construction of these positive examples (q, q^+) from user interactions. [1]

3.2 Clickthrough Data

Clickthrough data consist of records of user clicks on web pages after submitting search queries. Queries leading to clicks on the same URL are presumed to share similar search intent, as user clicks generally reflect fulfillment of informational needs (Beeferman and Berger, 2000; Croft et al., 2009).

However, simply mining query pairs that co-clicked on a single URL can produce false positive pairs, as records include unreliable information such as user misclicks or clicks to generic sites (e.g., news portals) that attract diverse queries. To mitigate these types of noise, we mined query pairs in which sets of clicked URLs are similar. By leveraging set similarity, we reduce the impact of noise, as the reliable click information within the sets helps identify appropriate query pairs. Following previous literature (Beeferman and Berger, 2000; Huang et al., 2023), we used the Jaccard coefficient as the measure of set similarity:

$$\text{Sim}_{\text{click}}(q_1, q_2) = \frac{\text{U}(q_1) \cap \text{U}(q_2)}{\text{U}(q_1) \cup \text{U}(q_2)}, \tag{2}$$

where q_1 and q_2 denote the search queries, and $\text{U}(q_i)$ denotes the set of URLs associated with q_i. Query pairs exceeding a similarity threshold θ were selected as positive pairs.

3.3 Session Data

Session data comprise sequences of queries submitted by a single user within a specific time frame t. Queries within the same session are assumed to have similar search intent, as they may involve reformulating queries, adding further information to previous queries, or searching for different aspects of the same topic (Huang et al., 2003).

Simply mining query pairs that co-occurred within a session can introduce noise, as users may also search with different intents within a session, such as aimless web surfing or addressing multiple informational needs. To address this, we aggregated the co-occurrence frequencies of adjacent queries from each session across multiple sessions (Fonseca et al., 2005), assuming that query pairs with similar search intent are more prevalent than those with different search intents. Since high-frequency queries, such as “YouTube”, can bias simple co-occurrence frequencies, we used the Jaccard coefficient that accounts for individual query frequencies (Huang et al., 2003):

$$\text{Sim}_{\text{session}}(q_1, q_2) = \frac{c(q_1, q_2)}{f(q_1) + f(q_2) - c(q_1, q_2)}, \tag{3}$$

where $c(q_1, q_2)$ denotes the co-occurrence frequency of q_1 and $q2$, and $f(q_i)$ the frequency of query q_i. This measure ensures that even if two queries frequently co-occur, they receive a low similarity score if one of them is popular across different contexts. Query pairs with similarity above a threshold ϕ were selected as positive pairs.

[1] We also experimented with hard negative sampling, but it did not yield improved results (see Appendix A).

Query 1	Query 2
Click	
ユニバーサルスタジオジャパン ホテル 安い (Universal Studios Japan Hotel Cheap)	大阪 USJ 格安ホテル (Osaka USJ Budget Hotel)
海外旅行 クレカ (Overseas Travel CC)	海外に強い クレジットカード (Credit Card Good for Overseas)
最も長い 蛇 (Longest Snake)	10m 蛇 (10m Snake)
Session	
TDLテリヤキチキン (TDL Teriyaki Chicken)	ディズニーランド 照り焼きチキン レシピ (Tokyo Disneyland Teriyaki Chicken Recipe)
コンビニ大根サラダ (Convenience Store Radish Salad)	コンビニ 大根サラダ アレンジ (Convenience Store Radish Salad Variations)
アメリカ 80万 旅行 (USA 800,000 Yen Trip)	アメリカ 1週間 旅費 (USA One Week Travel Cost)

Table 1: Examples of positive query pairs in UBIQUE.

Task	#Samples	#Associated
Query-Synonym Retrieval	5,000	1
Query Suggestion	951	8.2
Query Classification	1,456	N/A
Short-Text Reranking	4,667	25.5

Table 2: Statistics of the QU benchmark. #Samples denotes the size of the dataset, and #Associated denotes the average number of associated items per source query. The associated items were created based on human annotations.

Examples of the constructed query pairs are presented in Table 1.

4 Experiment

We evaluated UBIQUE on four real-world QU tasks using Japanese search query logs.

4.1 Evaluation

A multifaceted evaluation across various QU tasks is essential to assess the effectiveness of universal embeddings, as performance on one task may not correlate with performance on others (Muennighoff et al., 2023). Due to privacy and proprietary restrictions, comprehensive benchmarks covering multiple QU tasks are not publicly available. Therefore, we constructed a QU benchmark comprising the following four distinct tasks, including one with a public dataset.

Query-Synonym Retrieval (QR) This task retrieves queries that express the same intent despite lexical differences (Li and Xu, 2014). For each source query, retrieval was conducted by calculating cosine similarity against all other queries in the test set, excluding the source query itself. Mean Reciprocal Rank (MRR) was used as the evaluation metric.

Query Suggestion (QS) This task aims to retrieve contextually related queries that users may consider next. Related queries are sourced from related search keywords in our search system, curated by human evaluators for quality assurance. For evaluation, we retrieved the top ten queries from the full set of related queries, ranked by cosine similarity to the source query. We computed Normalized Discounted Cumulative Gain (NDCG)@10 by assigning a gain value of 1.0 to related queries and 0.0 to all others for each source query.

Query Classification (QC) This task involves categorizing geolocation-related queries into four classes: landmarks, chain stores, addresses, and station names. We trained a linear classifier on the embeddings and evaluated its performance using five-fold cross-validation following Conneau and Kiela (2018) and reported the average of macro F1 score.

Short-Text Reranking (SR) This task re-ranks product names linked to user queries using the publicly available ESCI dataset (Reddy et al., 2022). Each query corresponds to multiple products with graded relevance labels: Exact, Substitute, Complement, and Irrelevant. We assigned gain values of 1.0, 0.1, 0.01, and 0.0 to these labels, respectively, for computing NDCG. We ranked all the product names by cosine similarity to the source query.

Statistics of the QU benchmark are shown in Table 2.

Model	Params	QR	QS	QC	SR	Avg.
General						
SOTA						
Sup. SimCSE$_{\text{large}}$	337M	40.9	81.3	85.4	88.4	74.0
Ruri$_{\text{large}}$	337M	67.7	86.3	**88.0**	90.5	83.1
mE5$_{\text{large}}$	560M	63.1	87.3	82.4	91.1	81.0
Sarashina$_{\text{1.1b}}$	1.2B	$\underline{73.9}$	89.2	84.5	$\underline{91.3}$	84.7
OpenAI$_{\text{3-large}}$	-	65.9	89.9	80.8	**91.4**	82.0
Similar Scale						
DistilBERT	68M	20.3	79.7	83.4	87.3	67.7
Ruri$_{\text{small}}$	68M	54.5	87.6	84.1	90.8	79.3
mE5$_{\text{small}}$	118M	59.5	87.7	71.5	90.8	77.4
Search Logs						
Unsupervised						
fastText	-	22.9	84.8	82.5	87.7	69.5
Unsup. SimCSE	68M	28.5	84.8	83.2	88.2	71.2
Ours						
UBIQUE$_{\text{click}}$	68M	**91.4**	$\underline{91.2}$	85.8	90.5	**89.7**
UBIQUE$_{\text{session}}$	68M	71.9	**92.6**	$\underline{86.9}$	90.3	$\underline{85.4}$

Table 3: Performance comparison of models on QU benchmark. Metrics: QR (MRR), QS (NDCG@10), QC (F1), SR (NDCG). Avg. is the macro average across tasks. Bold: best, Underline: second best.

4.2 Training Details

The training data, sourced from user logs of Yahoo! JAPAN Search[2] in April 2024, includes 50 million query pairs. For clickthrough data, we set θ to 0.4, while for session data, we set ϕ to 0.2 and t to 300 seconds.[3] Queries containing predefined adult terms were excluded, as such queries often trigger diverse URL clicks or shifts in intent within a short time frame, resulting in the generation of irrelevant query pairs.

We used Japanese DistilBERT (Koga et al., 2023) as the base model, a lightweight model well-suited for practical deployment. The `[CLS]` representation was used as the query embedding. The batch size was set to 1,024, with a maximum sequence length of 16[4]. The learning rate was 2e-4, using linear decay and a warmup for the initial 1% of steps, with the AdamW optimizer. Training was conducted over 5 epochs, and we selected the best checkpoint on the basis of evaluations conducted every 4,000 steps. We implemented our code using Transformers (Wolf et al., 2020) and ran the training on four NVIDIA V100 GPUs, which took 16 hours. To leverage a large number of in-batch negatives crucial for model performance (Wang et al., 2024a), we used DeepSpeed ZeRO-2 (Rajbhandari et al., 2020) to reduce memory usage and scale up batch size (see Appendix B for details).

4.3 Baselines

We compared UBIQUE$_{\text{click}}$ and UBIQUE$_{\text{session}}$ with SOTA general domain text embedding models and unsupervised models trained on search queries.

We used five SOTA models: Japanese Sup. SimCSE$_{\text{large}}$ (Tsukagoshi et al., 2023), Ruri$_{\text{large}}$ (Tsukagoshi and Sasano, 2024), mE5$_{\text{large}}$ (Wang et al., 2024b), Sarashina$_{\text{1.1b}}$ (SB Intuitions, 2024), and the commercial model OpenAI$_{\text{3-large}}$ (OpenAI, 2024). We also used Japanese DistilBERT (UBIQUE's base model), Ruri$_{\text{small}}$, and mE5$_{\text{small}}$ as similar-scale models for fair comparison.

For unsupervised models, we used fastText (Bojanowski et al., 2017) and Unsup. SimCSE, both trained on 50 million queries. For fastText, we tokenized queries with MeCab (Kudo, 2006) and trained a 300-dimensional vector model using Skip-gram, with default hyperparameters. For Unsup. SimCSE, we used Japanese DistilBERT as the base model, with a learning rate of 3e-5, dropout rate of 0.2, and the same settings as our UBIQUE models for the remaining parameters (see Appendix C for details).

4.4 Results

Table 3 presents the evaluation results on the QU benchmark. UBIQUE$_{\text{click}}$ and UBIQUE$_{\text{session}}$ substantially outperformed all similar-scale models on most tasks and even surpassed the larger SOTA models on average. For instance, UBIQUE$_{\text{click}}$ achieved high scores on average, outperforming Ruri$_{\text{large}}$ by 6.6% in average performance (89.7% vs. 83.1%).[5] UBIQUE$_{\text{session}}$ also surpassed Ruri$_{\text{large}}$ with an average score of 2.3% and demonstrated exceptional strength in the QS task, achieving an NDCG@10 score of 92.6%, which is a 6.3% absolute improvement over the baseline's 86.3%. It is worth noting that these SOTA models are not solely based on contrastive learning but involve complex two-stage training pipelines using rerankers (Wang et al., 2024a; Li et al., 2023b). These results underscore the importance of constructing positive examples specialized for search queries.

[2] `https://search.yahoo.co.jp`

[3] Performance improved with higher thresholds for θ and ϕ, reaching a plateau at these values.

[4] This length covers 98.4% of the search queries.

[5] UBIQUE$_{\text{click}}$ even surpassed Ruri$_{\text{large}}$ on the dev set early in training, at just 2.5% of the total training steps.

Model	QR	QS	QC	SR
UBIQUE$_{\text{click}}$	**91.4**	**91.2**	**85.8**	**90.5**
w/o Jaccard	89.4	90.8	85.6	90.4
UBIQUE$_{\text{session}}$	**71.9**	**92.6**	86.9	**90.3**
w/o Jaccard	58.7	92.1	**87.1**	90.1

Table 4: Ablation study on the QU benchmark.

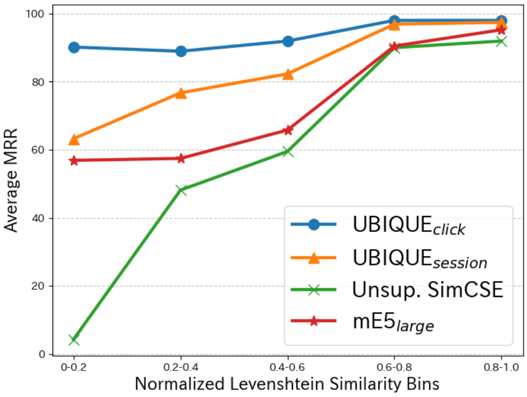


Figure 2: MRR scores on the QR task across different bins of normalized Levenshtein similarity.

UBIQUE$_{\text{click}}$ and UBIQUE$_{\text{session}}$ substantially outperformed unsupervised models trained on search queries. For example, Unsup. SimCSE achieved an MRR of 28.5% on the QR task, whereas UBIQUE$_{\text{click}}$ achieved 91.4%. This notable performance gap in QR task scores indicates that these unsupervised models struggle to capture semantic relationships between search queries with different appearances (see also §5.1), resulting in limited performance improvements.

To evaluate the effectiveness of the Jaccard coefficient in query pair selection, we conducted an ablation study. We trained UBIQUE$_{\text{click}}$ and UBIQUE$_{\text{session}}$ without applying Jaccard similarity thresholds (i.e., using query pairs that simply co-clicked on a single URL (Zhang et al., 2019) or just co-occurred in a session). As shown in Table 4, incorporating the Jaccard coefficient led to consistent performance improvements in both our models across most tasks. This suggests the importance of integrating a robust query-pair-mining approach based on the Jaccard coefficient to mitigate noise and irrelevant pairs.

5 Analysis

To understand the effectiveness of UBIQUE models, we conducted comparative analyses with representative baseline models.

5.1 Robustness to Lexical Variations

By leveraging user interactions for contrastive learning, UBIQUE$_{\text{click}}$ and UBIQUE$_{\text{session}}$ avoid reliance on appearances alone and capture the semantic meaning of search queries, which are often short and thus prone to lexical variations. To verify this property, we evaluated their performance on a query-synonym retrieval task across different edit distances.

As shown in Figure 2, we observed that all models achieved decent MRR scores for lexically similar pairs (e.g., “colour palette” and “color palette”). However, as the lexical difference increased (e.g., “purchase an automobile” and “buy car”), the scores of the baseline models, especially Unsup. SimCSE, decreased dramatically, whereas our models maintained their performance. These findings indicate that, while Unsup. SimCSE is highly sensitive to lexical variations, our models are robust against such variations and can appropriately capture the intent of queries. This robustness can be attributed to using user interactions as weak supervision, which enables the models to focus on semantic similarities rather than appearances.

5.2 Qualitative Analysis

To understand how our models improve query embeddings, we analyzed nearest neighbor queries for each model in the embedding space[6]. Representative nearest neighbor queries are shown in Table 5.

With mE5$_{\text{large}}$ and Unsup. SimCSE, the nearest neighbors often had similar appearances but different intents. For example, when given a query “ロス 旅費 (LA travel expenses)”, these baseline models retrieved “スイス 旅費 (Swiss travel expenses)” because they were affected by the lexical overlap “旅費 (Travel expenses)”, even though the destination differed. In contrast, our models succeeded in retrieving queries that share similar intents regardless of lexical differences, such as “ロサンゼルス 旅行 費用 (Cost of a trip to Los Angeles).” UBIQUE$_{\text{click}}$ tended to retrieve paraphrases of queries that more precisely matched the intent while UBIQUE$_{\text{session}}$ retrieved queries with broader or transitional intents, such as “ロス現地時間 (LA local time).” These observations align with the characteristics of each data source.

[6] Using Faiss (Douze et al., 2024), we conducted approximate nearest neighbor search on 10 million random queries.

Model	1st Query	2nd Query	3rd Query
Unsup. SimCSE	ケアンズ 旅費 (Cairns travel expenses)	スイス 旅費 (Switzerland travel expenses)	シンガポール 旅費 (Singapore travel expenses)
mE5$_{\text{large}}$	スイス 旅費 (Switzerland travel expenses)	ロサンゼルス 旅行 費用 (Cost of a trip to Los Angeles)	タイ 旅費 (Thailand travel expenses)
UBIQUE$_{\text{click}}$	旅行ロス (Trip to LA)	ロサンゼルス 旅行 費用 (Cost of a trip to Los Angeles)	ロサンゼルス物価 (Los Angeles cost of living)
UBIQUE$_{\text{session}}$	ロサンゼルス 旅行 費用 (Cost of a trip to Los Angeles)	ロス 羽田 (LA Haneda Airport)	ロス現地時間 (LA local time)

Table 5: Nearest neighbors in embedding space for "ロス 旅費 (LA travel expenses)" across models.

6 Conclusion and Future Work

We proposed UBIQUE, a simple yet effective approach to address the challenges of learning universal search query embeddings by harnessing user behavior data through contrastive learning. UBIQUE constructs positive query pairs from clickthrough and session data, enabling the model to align embeddings based on user intent rather than surface-level similarities. The empirical results on four practical QU tasks demonstrated that UBIQUE models outperformed strong baselines, particularly in their robustness to lexical variations in search queries.

While our study focused on a Japanese search system, we recognize that search styles can vary across languages (Chu et al., 2012). Since UBIQUE is theoretically applicable to other languages, evaluating its effectiveness in diverse linguistic contexts is an exciting future direction. Although we constructed our models separately using clickthrough and session data, combining these data sources may lead to further performance improvements. Incorporating additional information from search results, such as titles and documents, could further enhance UBIQUE, provided the potential increase in inference latency is acceptable.

7 Ethics Statement

Throughout UBIQUE's training data generation process (§3) and the creation of evaluation datasets (§4.1), all user information was rigorously anonymized to ensure that neither researchers nor reviewers could identify individual users. Specifically, user IDs were replaced with hashed strings, guaranteeing that personal identities remain undisclosed. Additionally, all annotation tasks were conducted by internal senior reviewers who had access only to the queries themselves, without any user information.

In our qualitative evaluation (§5.2), we included only queries that appeared at least ten times in the logs to further protect user privacy.

Acknowledgments

We would like to thank Dr. Satoshi Akasaki for his valuable comments and suggestions on an earlier version of this manuscript.

References

Doug Beeferman and Adam Berger. 2000. Agglomerative clustering of a search engine query log. In *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '00, page 407–416.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Huanhuan Cao, Derek Hao Hu, Dou Shen, Daxin Jiang, Jian-Tao Sun, Enhong Chen, and Qiang Yang. 2009. Context-aware query classification. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '09, page 3–10.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A Simple Framework for Contrastive Learning of Visual Representations. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607.

Ting Chen, Yizhou Sun, Yue Shi, and Liangjie Hong. 2017. On Sampling Strategies for Neural Network-based Collaborative Filtering. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '17, page 767–776.

Peng Chu, Eszter Jozsa, Anita Komlodi, and Karoly Hercegfi. 2012. An exploratory study on search behavior in different languages. In *Proceedings of the 4th Information Interaction in Context Symposium*, IIIX '12, page 318–321.

Alexis Conneau and Douwe Kiela. 2018. SentEval: An Evaluation Toolkit for Universal Sentence Representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Bruce Croft, Donald Metzler, and Trevor Strohman. 2009. *Search Engines: Information Retrieval in Practice*, 1st edition.

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2024. The Faiss library. *Preprint*, arXiv:2401.08281.

Bruno M. Fonseca, Paulo Golgher, Bruno Pôssas, Berthier Ribeiro-Neto, and Nivio Ziviani. 2005. Concept-based interactive query expansion. CIKM '05, page 696–703.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple Contrastive Learning of Sentence Embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910.

R. Hadsell, S. Chopra, and Y. LeCun. 2006. Dimensionality Reduction by Learning an Invariant Mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742.

Homa Baradaran Hashemi. 2016. Query intent detection using convolutional neural networks.

Chien-Kang Huang, Lee-Feng Chien, and Yen-Jen Oyang. 2003. Relevant term suggestion in interactive web search based on contextual information in query session logs. *J. Am. Soc. Inf. Sci. Technol.*, 54(7):638–649.

Yupin Huang, Jiri Gesi, Xinyu Hong, Han Cheng, Kai Zhong, Vivek Mittal, Qingjun Cui, and Vamsi Salaka. 2023. Behavior-driven query similarity prediction based on pre-trained language models for e-commerce search. In *SIGIR 2023 Workshop on eCommerce*.

Haoming Jiang, Tianyu Cao, Zheng Li, Chen Luo, Xianfeng Tang, Qingyu Yin, Danqing Zhang, Rahul Goutam, and Bing Yin. 2022. Short Text Pre-training with Extended Token Classification for E-commerce Query Understanding. *Preprint*, arXiv:2210.03915.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781.

Kobayashi Koga, Shengzhe Li, Akifumi Nakamachi, and Toshinori Sato. 2023. LINE DistilBERT Japanese.

Taku Kudo. 2006. MeCab: Yet Another Part-of-Speech and Morphological Analyzer.

Tessa Lau and Eric Horvitz. 1999. Patterns of Search: Analyzing and Modeling Web Query Refinement. In *Proceedings of the Seventh International Conference on User Modeling, Banff, Canada, June 1999*, pages 119–128.

Hang Li and Jun Xu. 2014. Semantic Matching in Search. *Found. Trends Inf. Retr.*, 7(5):343–469.

Juanhui Li, Wei Zeng, Suqi Cheng, Yao Ma, Jiliang Tang, Shuaiqiang Wang, and Dawei Yin. 2023a. Graph Enhanced BERT for Query Understanding. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, page 3315–3319.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023b. Towards General Text Embeddings with Multi-stage Contrastive Learning. *Preprint*, arXiv:2308.03281.

Fangyu Liu, Ivan Vulić, Anna Korhonen, and Nigel Collier. 2021. Fast, Effective, and Self-Supervised: Transforming Masked Language Models into Universal Lexical and Sentence Encoders. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*.

Niklas Muennighoff, Nouamane Tazi, Loic Magne, and Nils Reimers. 2023. MTEB: Massive Text Embedding Benchmark. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2014–2037.

OpenAI. 2024. New embedding models and API updates.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. 2020. ZeRO: Memory Optimizations Toward Training Trillion Parameter Models. *Preprint*, arXiv:1910.02054.

Chandan K. Reddy, Lluís Màrquez, Fran Valero, Nikhil Rao, Hugo Zaragoza, Sambaran Bandyopadhyay, Arnab Biswas, Anlu Xing, and Karthik Subbian. 2022. Shopping Queries Dataset: A Large-Scale ESCI Benchmark for Improving Product Search. *Preprint*, arXiv:2206.06588.

SB Intuitions. 2024. Sarashina-embedding-v1-1b.

Ben Shneiderman, Don Byrd, and W. B Croft. 1997. Clarifying Search: A User-Interface Framework for Text Searches. Technical report.

Hayato Tsukagoshi and Ryohei Sasano. 2024. Ruri: Japanese General Text Embeddings. *Preprint*, arXiv:2409.07737.

Hayato Tsukagoshi, Ryohei Sasano, and Koichi Takeda. 2023. Japanese SimCSE Technical Report. *Preprint*, arXiv:2310.19349.

Bin Wang, C.-C. Jay Kuo, and Haizhou Li. 2022. Just Rank: Rethinking Evaluation with Word and Sentence Similarities. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6060–6077.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2024a. Text Embeddings by Weakly-Supervised Contrastive Pre-training. *Preprint*, arXiv:2212.03533.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024b. Multilingual E5 Text Embeddings: A Technical Report. *Preprint*, arXiv:2402.05672.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45.

Xing Wu, Chaochen Gao, Liangjun Zang, Jizhong Han, Zhongyuan Wang, and Songlin Hu. 2022. ESimCSE: Enhanced Sample Building Method for Contrastive Learning of Unsupervised Sentence Embedding. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3898–3907.

Dejiao Zhang, Shang-Wen Li, Wei Xiao, Henghui Zhu, Ramesh Nallapati, Andrew O. Arnold, and Bing Xiang. 2021. Pairwise Supervised Contrastive Learning of Sentence Representations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*.

Hongfei Zhang, Xia Song, Chenyan Xiong, Corby Rosset, Paul N. Bennett, Nick Craswell, and Saurabh Tiwary. 2019. Generic Intent Representation in Web Search. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, page 65–74.

A Limitations of Hard Negative Sampling

Model	**QR**	**QS**	**QC**	**SR**
UBIQUE$_{\text{click}}$	**91.4**	**91.2**	85.8	**90.6**
w/ hardnegatives	90.2	90.1	**86.2**	90.1

Table 6: Results of introducing hard negatives.

To capture more fine-grained information with our models, we aimed to incorporate hard negatives —negative examples that are challenging to distinguish from the anchor query. Following a prior study (Karpukhin et al., 2020), we selected hard negative queries that are lexically similar to the anchor query (i.e., with small edit distances) but have non-overlapping sets of clicked URLs. Specifically, we applied string matching using SimString [7] to a dataset of 10 million queries, treating the anchor query from clickthrough-based training pairs (§3.2) as the search string. To avoid false negatives due to missing click information, we ensured that all 10 million queries in this dataset were associated with click data. We empirically set the similarity range to 0.45–0.60 to avoid selecting queries that are too lexically similar as hard negatives. We then filtered out extracted queries with any overlapping clicked URLs, treating the remaining queries as hard negatives. Using these hard negatives, we constructed a triplet dataset (i.e., anchor, positive, hard negative) and conducted additional contrastive learning using UBIQUE$_{\text{click}}$.

Despite this effort, overall task performance slightly declined (see Table 6). While this model showed a slight improvement in distinguishing lexically similar negatives, it struggled overall to recognize semantically equivalent queries. This decline in performance may be attributed to the inherent difficulty of consistently using lexically similar queries as negatives, as surface features can also serve as cues for query representation. Future work will focus on refining the negative sampling strategy beyond simple edit-distance measures.

B Training Details

To construct the training data, we conducted deduplication to prevent overfitting and excluded query pairs included in the test set to prevent leakage. The learning rate was explored from {2e-4, 3e-4, 3e-5}, and we chose the best one, 2e-4, based on the dev set. For evaluation during training to select the best checkpoint, we used query-synonym retrieval, as the symmetric retrieval task exhibits a strong correlation with downstream tasks (Wang et al., 2022). We used a dev set consisting of 5,000 queries for evaluation.

We also tried using Ruri$_{\text{small}}$ as the base model for UBIQUE models. Ruri$_{\text{small}}$ was initialized with Japanese DistilBERT and further trained using contrastive learning with weak supervision on large-scale web data. While Ruri$_{\text{small}}$-based UBIQUE models' performance was relatively higher than

[7] https://www.chokkan.org/software/simstring/index.html.en

that of Japanese DistilBERT-based UBIQUE models in the initial stages of training, the final performance showed a negligible difference. This result underscores the importance of using user-behavior data rather than general web data for constructing query embedding models.

C Baseline Details

We used `[CLS]` pooling for Sup. SimCSE$_{\text{large}}$, mean pooling for DistilBERT, Ruri, and mE5, and last-token pooling for Sarashina$_{\text{1.1b}}$, with a maximum sequence length of 512 used across all models. For mE5 and Ruri, it is necessary to add a prefix to the input sentence, indicating whether it is a source text (query) or a target text (passage) to differentiate the embeddings. We added a query prefix to the source query across all tasks. For the target query, the prefix was added according to the task: a query prefix was used for the symmetric task QR, while a passage prefix was used for the asymmetric tasks QS and SR.

For fastText, we obtained query embeddings by applying mean pooling to the vectors of each token. In Unsup. SimCSE, we explored learning rates from {2e-4, 3e-4, 3e-5} and dropout rates from {0.05, 0.1, 0.2} on the dev set, choosing the best ones, 3e-5 and 0.2, respectively.