Robust Query Performance Prediction for Dense Retrievers via Adaptive Disturbance Generation

Abbas Saleminezhad[1], Negar Arabzadeh[1], Radin Hamidi Rad[1], Soosan Beheshti[1], Ebrahim Bagheri[1],
Corresponding author: Abbas Saleminezhad[1]

[1*]Electrical and Computer Engineering Department, Toronto Metropolitan University, Toronto, On, Canada.

Contributing authors: abbas.saleminezhad@torontomu.ca; narabzad@torontomu.ca; radin@torontomu.ca; soosan@torontomu.ca; bagheri@torontomu.ca;

Abstract

This paper introduces `ADG-QPP` (Adaptive Disturbance Generation), an unsupervised Query Performance Prediction (QPP) method designed specifically for dense neural retrievers. The underlying foundation of `ADG-QPP` is to measure query performance based on its degree of robustness towards perturbations. Traditional QPP methods rely on predefined lexical perturbations on the query, which only apply to sparse retrieval methods and fail to maintain consistent performance across different datasets. In our work, we address these limitations by perturbing the query by injecting disturbance leveraged by the focal network-based measurements including node-based, edge-based, and cluster-based metrics, into its neural embedding representation. Rather than applying the same perturbation across all queries, our approach develops an instance-wise disturbance for each query that is then used for its perturbation. Through extensive experiments on three benchmark datasets, we demonstrate that `ADG-QPP` outperforms state-of-the-art baselines in terms of Kendall τ, Spearman ρ, and Pearson's ρ correlations.
Keywords: Information Retrieval, Query Performance Prediction, Post-retrieval Query Performance Prediction, and Dense Neural Retrievers.

1 Introduction

The main objective of many Information Retrieval (IR) methods is to curate and present information for the users to satisfy their information needs. In practice, IR methods do not necessarily exhibit comparable performance when addressing users'

information needs, often expressed through user queries. In other words, a particular IR method may show strong performance on a certain subset of user queries while not being quite effective on others. In addition, IR methods do not necessarily show coherent performance with each other where the difficult queries for one IR method may in fact be easy for another method. As such, the IR community has been interested in predicting the performance of queries for a given IR method to estimate how well the IR method can satisfy the query. More formally, the Query Performance Prediction (QPP) task is aimed at estimating the effectiveness of the retrieved results for a given query without having explicit information about the relevance of information for the user.

Query performance prediction has several real-world applications that significantly enhance the effectiveness and efficiency of information retrieval systems. One key application is in adaptive retrieval strategies, where QPP can predict the difficulty of a query and dynamically adjust the retrieval process. For simpler queries, search engines can employ cost-effective and faster rankers, while more complex queries may trigger the use of sophisticated, resource-intensive rankers. This not only optimizes resource usage but also minimizes system latency. Additionally, QPP can improve user engagement by identifying difficult queries and prompting users with clarifying questions to better understand their intent [1]. This interaction can increase user satisfaction, as it demonstrates the system's intelligence and responsiveness. Another significant application is in federated search and metasearch engines, where QPP can guide the merging of results from multiple data sources by weighing them according to their estimated quality [2, 3]. Additionally, QPP is useful in content enhancement through missing content analysis, allowing system administrators to identify and address gaps in the document collection to better answer emerging user needs. These applications demonstrate QPP's role in making retrieval systems more responsive, efficient, and user-centric [4].

Given the significance of QPP methods in the context of retrieval methods, researchers have examined various ways through which query performance could be estimated using unsupervised signals including query characteristics, distribution of content in the corpus, the semantic association between the query and the corpus, as well as, supervised approaches that learn to regression models by various forms of fine-tuning large language models for this task. One QPP approaches that received early attention without notable success was the idea of exploiting the relation between *query robustness* and its retrieval effectiveness. The premise behind these models is that a query is considered robust if it does not show considerable performance variation when subjected to *slight perturbations*. In other words, if the lexical form of a query changes (e.g., changing a query such as '*who is president of the united states*' to '*president of the US*'), the extent to which the retrieved list changes due to the lexical change of the query (perturbations) is an indication of query robustness, meaning that more robust queries are essentially easier for the retrieval method to satisfy. While theoretically elegant, QPP methods that operate based on robustness to perturbations face two major Limitations (**L1**) Their first major limitation is that they resort to lexical changes to the surface form of the query, which while useful for sparse

retrievers, do not apply to more advanced dense neural retrievers that show less sensitivity to the lexical form of the query. (**L2**) Furthermore, even on sparse retrievers, these QPP methods [4–6] do not show consistent performance across different datasets given their sensitivity to the type of perturbation that is applied to the queries.

In this paper, we propose a QPP method, referred to as `ADG-QPP` (Adaptive Disturbance Generation), which builds on the foundations of earlier work that have considered query robustness to lexical perturbations as a means to estimate query performance. In particular, `ADG-QPP` is designed to address the two major limitations of earlier work (L1 and L2). Our proposed approach is designed specifically to estimate query performance for dense neural retrievers and therefore rather than focusing on introducing perturbations on the lexical form of the query, we pay special attention on injecting disturbance into the neural embedding representation of queries. This way, our approach applies perturbations at the deeper semantic layer of the query captured in its embedding representation rather than changing the query itself, which may or may not be semantically meaningful, addressing L1. In addition, in our approach, we propose ways through which customized disturbance will be determined on an instance-wise basis, and as such, perturbations generated by measurements based on focal network metrics introduced in 3.4.1, applied to each query will be specific to that query. These metrics include three main categories: (1) *Node-based* measures (Semantic Network Size, degree centrality, closeness centrality, and PageRank); (2) *Edge-based measures* (Connectivity Score, Query Connectivity Strength, Average Query Connectivity, and Rare Path Index); and, (3) *Cluster-based measures* (Centroid Cluster Weight and Inter-cluster Connectivity). Therefore, our approach makes it possible to maintain consistent QPP performance across different dataset, hence satisfying L2. The key contributions of our work can be enumerated as follows:

1. We introduce the `ADG-QPP` method for estimating query performance for dense neural retrievers. Our method operates based on query robustness measured through changes observed in retrieval performance of a query after the injection of noisy perturbations on the embedding representation of the query;
2. We provide a systematic framework on how a customized degree of disturbance can determined to be injected into each particular query to optimize QPP performance;
3. Through extensive experiments on multiple benchmark datasets, such as TREC DL 2019 and 2020, and TREC DL-Hard [7], we find that our proposed approach offers consistently better performance compared to the state of the art baselines on metrics such as Kendall τ and Spearman ρ, and Pearson Pearson's ρ Correlations.

2 Related work

Query performance prediction methods are broadly categorized into pre-retrieval and post-retrieval approaches [4]. While pre-retrieval methods rely on query features and corpus statistics before retrieval [5, 8, 9], post-retrieval methods leverage an additional piece of information from the retrieved documents to predict query performance. Post-retrieval QPP methods have shown higher effectiveness compared to pre-retrieval methods, i.e., showing higher correlation with the actual retrieval system performance [6, 10, 11]. This paper is focused on post-retrieval QPP.

Accurate prediction of query performance has proven beneficial in various applications [12–15]. For instance, QPP has been effective in adapting retrieval strategies on a per-query basis [16]. By predicting query difficulty, search engines can deploy cost-effective rankers for simpler queries and more complex, resource-intensive rankers for more challenging queries [17, 18]. This approach optimizes resource usage while minimizing system latency. Additionally, QPP can enhance user engagement by identifying difficult queries and interacting with users to request further clarification of their intent i.e., by asking users clarifying questions [19, 20]. Previous research has shown that such interactions increase user satisfaction, as users perceive it as a sign of system intelligence [21].

Traditionally, post-retrieval QPP methods relied on term statistics from the index to predict the performance of traditional sparse bag-of-words high-dimensional retrievers, such as BM25 [22]. These methods were based on several ideas, including the association between the query and the retrieved documents [23], the query and the corpus, the retrieved documents and the corpus [23], as well as the inter-association between the retrieved documents [24]. For instance, some methods evaluated how the language model induced from the retrieved documents differed from that of the collection [25, 26]. Another group of traditional QPP methods focused on the retrieval scores among the top retrieved documents [6, 10, 11, 27, 28]. The idea was that a high variance in scores indicate that relevant and non-relevant documents were easier to distinguish. Conversely, a low standard deviation suggests difficulty in separating relevant from non-relevant documents, indicating a harder query to satisfy. More recently, with the advent of neural models, several QPP methods have used neural embeddings to tackle the task [5, 29–31]. With the availability of large-scale benchmarks and datasets, supervised neural-based models, such as NeuralQPP [29], NQA-QPP [32], BERT-QPP [33] and qpp-BERT-pl [34] have emerged and shown strong performance. These neural-based models have shown to outperform traditional term frequency-based QPP methods in several benchmarks. We provide a more detailed description of such methods in Section 4.1.

While most proposed methods are tailored to the characteristics of sparse retrievers, such as QL (Query Likelihood) and BM25 which are both high-dimensional bag-of-word based retrievers [35, 36] , little attention has been given to QPP for dense retrievers [37, 38]. Dense retrievers are supervised and, due to limited training data, sharing the data between ranker and QPP model training is challenging. Training dense retrievers and QPP models on the same data can lead to overfitting, causing QPP to overestimate performance. Therefore, we propose an unsupervised QPP method specifically designed for dense retrievers to avoid data overlap between QPP and ranking tasks. Our method, referred to as ADG-QPP, leverages the advantages of neural embeddings by using neural embedding representations of queries and retrieved documents for QPP. In this work, we build on the idea of disturbance injection through the retrieval channel, previously used in [6, 39] where the authors propose generating a noisy query based on the retrieved documents, similar to the idea of using pseudo relevance feedback. Among perturbation-based methods, Dense-QPP seeks to estimate robustness by adding a constant disturbance to all query representations.

However, the randomness of the added noise causes fluctuations in performance, making the results inconsistent. To address this, the authors proposed to inject the noise multiple times (30 as described in their paper) and calculate the average of predicted performance as the final prediction value. While this approach reduces variability and tends to increase the stability of the injected noise, it remains unstable and computationally demanding. This exposes a significant limitation in Dense-QPP, as it relies on multiple trials to achieve stability rather than effectively managing the underlying variability. By comparing results retrieved from the original query against those from its perturbed version, we assess the robustness of the query. A robust query is less affected by disturbance, implying it can handle perturbations without significant loss of retrieval effectiveness. In other words, the difference between the retrieved results of the original query and the perturbed query serves as a signal for query robustness or performance estimation.

While previous works have demonstrated the efficacy of perturbations for QPP , they were not generally stable and consistent across different benchmark datasets. This was partly because the perturbations happened at the lexical form of the query. Additionally, the perturbations were designed specifically for sparse retrievers, which are not applicable for dense retrievers. Our method aims to improve and expand upon the idea of query perturbation by generating *adaptive perturbations to queries specifically for dense retrievers*. We utilize the geometric properties of neural embedding of queries and their retrieved documents to generate adaptive disturbance on a per query basis, achieving better and more consistent results [5, 40].

3 Methodology

3.1 Problem Definition

Within the context of information retrieval, the Query Performance Prediction (QPP) task is defined as estimating the effectiveness of a retrieval method R in satisfying the information need behind a given query q without having access to relevance judgements. In practice, a retrieval method R would retrieve a ranked list of documents D_q for a given query q, denoted as $D_q \leftarrow R(q, C)$. Here, C represents the corpus of documents from which documents are retrieved and ranked, and $R(\cdot)$ is a function that takes the corpus C and a query q and produces a ranked list of documents D_q. The documents in D_q are rank-ordered based on their relevance score to the query.

By having relevance judgment annotations of documents in C for a given q, we measure the effectiveness of $R(q, C)$ by its ability to rank-order all relevant documents at the top of the ranked list. The retrieved documents D_q are assessed with an evaluation function $\mu(D_q|q, C)$. Some of the common μ functions are Mean Average Precision (MAP), Mean Reciprocal Rank (MRR), and normalized Discounted Cumulative Gain (nDCG). Finally, the quality of the retrieval set D_q is predicted as $\hat{\mu}(D_q|q, C)$ through a QPP method $\phi(D_q, q, C)$ through comparison with the actual performance $\mu(D_q|q, C)$. The predicted effectiveness of the retrieved results D_q for a given query q is evaluated by how accurate $\hat{\mu}(q, D_q)$ can predict $\mu(D_q, q)$ on a set of queries.

3.2 Approach Rationale

The foundation of our proposed QPP method is based on the relationship between the *robustness of a query to possible perturbations* and the subsequent performance of the information retrieval method on such a query. In simpler terms, a query is considered more robust if random perturbations do not significantly alter the retrieved document list. In other words, changes in the query do not lead to substantial changes in retrieval. Such a robust query could be seen as being resistant to changes in performance. Researchers have argued that, in the context of sparse retrievers, robust queries are those that have high discriminative power in identifying relevant documents and therefore are not impacted by perturbations. Conversely, less robust queries are sensitive to perturbations, resulting in notable changes to the retrieved document list when the query is altered. In practice, more robust queries are considered to be easier from a QPP point of view, as regardless of how the user formulates the query, it is highly likely that a similar set of relevant documents are retrieved for the query. On the other hand, less robust queries can be seen as being more difficult as small changes in query formulation by the user will lead to considerable changes in the results.

Let us consider the process of perturbing a query as a *noisy communication channel* through which a query acts as a signal passing through, potentially becoming noisier. One would be able to assess the robustness of the query before and after passing through the noisy channel by comparing results retrieved from the original query against those from its perturbed version, essentially testing the impact of disturbance on the query's effectiveness. A robust query is less affected by such disturbance, implying it can handle perturbations without significant loss of retrieval effectiveness. In contrast, a less robust query exhibits greater variance in its results when subjected to similar disturbances, indicating susceptibility to query formulation alterations. For sparse retrieval methods, the noisy channel can apply alterations on the surface form of the query, e.g., changing 'what the oldest you can have a baby' to 'what the *youngest* you can have a baby'. In contrast, given dense neural retrieval methods operate on the query based on its dense representation, it is possible inject disturbance into the embedding space rather than relying solely on lexical manipulations [6, 41]. Embedding space allows for more nuanced, fine-grained adjustments to query representations, offering a richer space for testing robustness through noisy perturbations.

Our work in this paper focuses on predicting query performance for *dense retrievers* by offering a systematic approach to inject disturbance into query representations such that query robustness can be measured. The fundamental premise of our approach is to test the resilience of queries to disturbances in the embedding space represented as dense vectors. By deliberately injecting disturbance into these embeddings and analyzing the impact on the retrieval results, it may be possible to measure the robustness of queries. Our proposed method leverages the continuous nature of embedding spaces, which allows for subtle and complex perturbations, unlike discrete changes typical in lexical spaces. By assessing how well queries withstand such perturbations, our proposed method aims to predict query effectiveness.

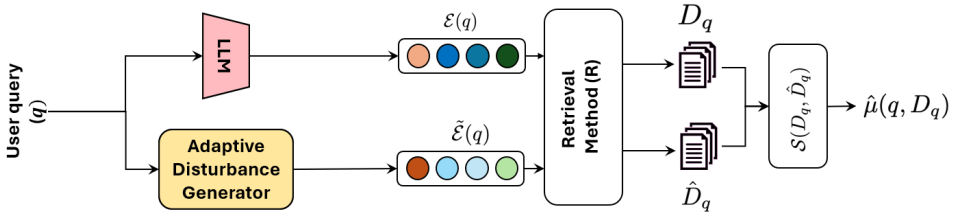

Fig. 1: The overview of our proposed approach. We note that the *adaptive disturbance generator* part is shown in Figure 2 .

3.3 Proposed Approach

Given a query q, a function $\mathcal{E}(.)$ maps queries into an embedding space as a dense p-dimensional vector where p. To retrieve the top-k relevant documents from a corpus C relevant to q, we employ a dense retriever R_d, such that $D_q^k = R_d(\mathcal{E}(q), C)$.

We introduce perturbations into the vector representation of the query $\mathcal{E}(q)$, by constructing a disturbance-injected query representation as follows:

$$\tilde{\mathcal{E}}(q) = \mathcal{E}(q) + \mathcal{N} \tag{1}$$

where $\mathcal{N}$ represents the injected disturbance. We conceptualize query perturbations by introducing a non-specific disturbance represented as $\mathcal{N}$, encompassing a range of potential noise models that can be injected into a dense query representation. This non-specific disturbance $\mathcal{N}$ is offered as a way to capture real-world variability in query representations while maintaining the dimensional bounds of both $\mathcal{E}(q)$ and $\tilde{\mathcal{E}}(q)$.

As illustrated in Figure 1, our approach involves first mapping each query into a dense vector representation based on a large language model (LLM). This embedding is then subjected to perturbations through an Adaptive Disturbance Generator shown in Figure 2, which considers the semantic relationships among queries or retrieved documents. The figure shows the process of generating the perturbed query embeddings $\tilde{\mathcal{E}}(q)$ by injecting context-sensitive disturbance in a Query-based Focal Network (QFN) which will be defined in Section 3.4.1. Given the original query and the disturbed query representations, we examine the retrieval outcomes for both of the representations. We define a performance metric QPP_{ADG} as a function that captures the association between the retrieved document lists from the original and perturbed queries. This association is a potential indicator of the stability of query performance when subjected to disturbances. The formal representation of QPP_{ADG} is defined as:

$$QPP_{ADG}(q, R_d, C) = \mathcal{S}(D_q^k, \tilde{D}_q^k) \tag{2}$$

where D_q^k is the corresponding list of documents for the original query q and the list retrieved post-perturbation is denoted by $\tilde{D}_q^k = R_d(\tilde{\mathcal{E}}(q), C)$. Furthermore, the function $\mathcal{S}$ is quantifies the differences between the retrieved lists of results for the original query $\mathcal{E}(q)$ and the perturbed query $\tilde{\mathcal{E}}(q)$.

The quantification of differences based on $\mathcal{S}$ needs to correlate with actual query performance. In particular, in document retrieval, items ranked lower are less likely to be seen by the users and are therefore less important than items higher in the list. Thus, $\mathcal{S}$ will need to show sensitivity to not only the presence of items on the two lists but also their relative ranking in the two lists. The Ranked Bias Overlap (RBO) measure [42] has shown to be able satisfy such conditions in that it captures both item similarity as well as item rank when comparing two lists and therefore we propose

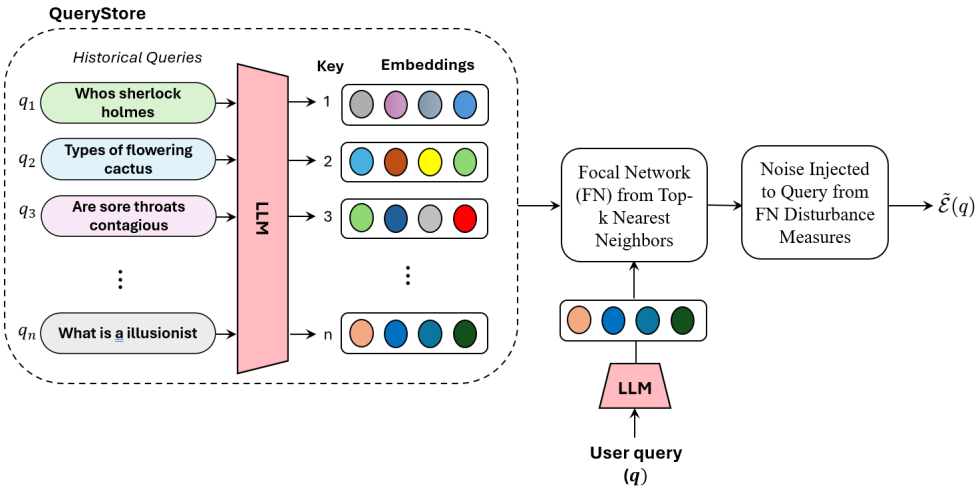

Fig. 2: Overview of the *adaptive disturbance generator* method depicting a Query-based Focal Network (QFN).

that it can be appropriate for operationalizing $\mathcal{S}$. More formally, the overlap O_d at depth d between D_q^k and $\tilde{D}_q^k$ is the number of items common between the two lists in their top d positions. The weighted overlap at depth is $W_d = \frac{O_d}{d}$. On this basis, $\mathcal{S}(D_q^k, \tilde{D}_q^k)$ can be defined based on W_d with a decay factor p such that importance of items exponentially decreases as the depth increases:

$$\mathcal{S}(D_q^k, \tilde{D}_q^k) = (1-p)\sum_{d=1}^{L} p^{d-1} W_d \tag{3}$$

where L is the maximum length $\mathcal{S}$ will be calculated. In the following sections, we will elaborate on the characteristics appropriate for the disturbance $\mathcal{N}$ in Equation 1.

3.4 Adaptive Disturbance Generation for Query Perturbation

To operationalize the disturbance $\mathcal{N}$ in Equation 1, a clean approach would be to inject an Additive White Gaussian Noise (AWGN) into the representations of each query to produce query perturbations. Such an approach offers several advantages: (i) AWGN has a uniform power spectral density across frequencies [43, 44], ensuring that embedding vector elements receive noise uniformly across different frequencies; and (ii) AWGN follows a Gaussian distribution, which is desirable as it accurately models real-world noisy perturbations [45]. By uniformly applying AWGN to all queries, it would be possible to assess how the retrieval method handles noise and identify which queries are more robust and which are more sensitive to noisy disturbances. Based on AWGN, the perturbation to individual query representations $\mathcal{E}(q)$ can be defined as:

$$\tilde{\mathcal{E}}(q) = \mathcal{E}(q) + \mathcal{N}(0, \sigma^2) \tag{4}$$

where $\mathcal{N}(0, \sigma^2)$ represents the Gaussian noise distribution with a mean of 0 and a variance of σ^2, which is added to each query representation. The variance σ^2 is a critical parameter in this equation, determining the intensity of the noise being introduced. It influences the degree of perturbation each query representation receives where a higher variance results in a more significant deviation from the original query.

However, a significant drawback of injecting disturbances into queries, as described in Equation 4, is the assumption of homogeneity among queries. This assumption does not account for the diverse nature of queries in practical scenarios. Some queries might be naturally more robust due to their clarity, specificity, or corpus-related properties, while others might be inherently more ambiguous and sensitive to specific disturbances. Consequently, applying the same level of disturbance to all queries might not produce an ideal disturbance for creating useful query variants for the QPP tasks. While the uniform AWGN offers a simple and clean way to generate perturbations, it may not capture the individualized response of the retrieval system to each unique query. Injecting instance-wise *personalized* disturbance enables a fine-tuned application of disturbance, aligning more closely with the semantic features and complexities of each query, thereby yielding a more discerning assessment of the retrieval system's robustness. For this reason, we additionally propose to personalize disturbance to the specific representation of each query. This *Adaptive Disturbance Generation* method utilizes network-based metrics to inform the level of perturbations applied to each representation on a per query basis, enabling a contextualized evaluation of query robustness. This *adaptive disturbance generation* method still incorporates Gaussian noise but adjusts the disturbance level according to the structural and semantic context of each query within the embedding space. By adjusting the disturbance level on an instance-wise basis, the method adjusts the injected disturbance based on each query's specific representation.

Previous studies have shown that the geometric properties of neural representations can be useful for various tasks such as specificity quantification and analogical reasoning [46, 47]. Similarly, we explore personalized disturbance to be injected in each query representation based on the context of the query representation in the embedding space. The idea is that the context surrounding each query in the embedding space is a potential indicator for the sensitivity of the query itself. For example, a query surrounded by diverse documents in the embedding space may experience a significant performance drift with a small amount of disturbance, whereas a query surrounded by documents with highly similar content may remain stable despite the injected disturbance. Therefore, the representation of queries in the embedding space and the documents that surround them can potentially be used as a measure for personalizing the disturbance to be injected into the query.

3.4.1 Focal Networks

To explore the surrounding space of a query within the embedding space, we define the concept of a *focal network*. A focal network for a query q, represented as $\zeta(q_i) = (\mathbb{V}_\zeta, \mathbb{E}_\zeta, \gamma)$, is a weighted undirected graph where $\mathbb{V}_\zeta$ includes the graph vertices, and $\mathbb{E}_\zeta = \{e_{i,j} : i, j \in \mathbb{V}_\zeta\}$ includes the edges between every pair of nodes i and j. The function $\gamma : \mathbb{E}_\zeta \rightarrow [-1, 1]$ determines the edge weights as the semantic relatedness between the two nodes e.g., cosine similarity of their embedded dense vector representations. Based on the types of nodes participating in constructing the focal network, we propose to build (i) a Query-based Focal Network `(QFN)`, and (ii) a Document-based Focal Network `(DFN)`.

Query Store: The QueryStore is defined as $\mathbb{Q} = \{q_1, q_2, \ldots, q_n\}$ where each q_i is a query previously submitted to the same search engine by the users. Upon receiving an input query, the QueryStore mechanism computes the similarity between this query and the archived queries using a predefined distance metric. Given the LLM decoder, top K similar queries are identified by mapping the representation of the main query to its similar vector representations in the Q set.

Query-based Focal Network (`QFN`): A Query-based Focal Network ζ_{QFN} is a focal network where the set of nodes $\mathbb{V}_\zeta$ consists of the main query q and a set of similar queries to q noted as $\mathbb{Q}_q$. Here, $\mathbb{Q}_q$ is obtained from a *QueryStore* $\mathbb{Q}$, which consists of a set of previously submitted queries to the same search engine.

Now, to build the `QFN` for query q, i.e., $\zeta_{QFN}(q)$, we obtain the set of vertices $\mathbb{V}_\zeta^{QFN}$ including the main query as well as most similar queries, i.e., $\mathbb{V}_\zeta^{QFN} = \{q \cup \mathbb{Q}_q\}$. The edges in `QFN` represent the similarity between all vertices in $\mathbb{V}_\zeta^{QFN}$.

Document-based Focal Network (`DFN`): In contrast to `QFN` where a query is contextualized with similar queries received from the users in the past, the document-based focal network `DFN` contextualizes a query in the context of its top-K retrieved documents D_q. The `DFN` network contextualizes the query with the documents that surround it in the embedding space. More precisely, $\mathbb{V}_\zeta^{DFN} = \{q \cup \{d \in D_q\}\}$ consists of the query itself and the set of top-k retrieved documents for this query. Similar to `QFN`, the edges in `DFN` represent the similarity between all vertices in $\mathbb{V}_\zeta^{DFN}$.

In both `QFN` and `DFN` networks, an edge between any two nodes i and j is weighted by the semantic similarity between the two nodes, which can be measured by any similarity-based metric between the representation of the two nodes. Without loss of generality, in this paper, we adopt the cosine similarity measure for this purpose. We define the pruning constant ϵ as the threshold for edge weights. Both networks $\zeta(q)$ can be pruned and sparsified to $\zeta_\epsilon(q)$ by removing edges with weights below ϵ. More specifically, we aim to construct a focal network for q_i, where q_i serves as the focal node and forges connections directly to other queries and documents only if the semantic relatedness between the focal node and its neighbors surpasses a specified threshold. In essence, two nodes within the ϵ-neighborhood of the query q_i, exhibiting semantic relatedness to q_i greater than ϵ, are interconnected within the focal network. We propose that the properties of the `QFN` and `DFN` networks that capture the context surrounding each query would allow us to to derive context-specific disturbances, i.e., σ in Equation 4. Variance σ is determined directly by measuring focal network metrics.

For the sake of clarity, we visualize two sample `QFN` networks in Figure 3 representing two queries, namely '*access, how to go to most recent record*' (on the left), and '*what important job do the lysosomes have*' (on the right). In each `QFN`, the query is centered, surrounded by its top-6 most similar queries. In this figure, any edges with weights below $\epsilon = 0.5$ have been pruned. In the query on the left, we demonstrate a poorly performing query, which cannot be satisfactorily addressed by the retrieval method. As seen in the figure, the `QFN` network for this query is sparsely connected, potentially pointing to the fact that any perturbations to they query can lead to higher changes in changes in retrieval. Conversely, the query on the right side depicts the `QFN` network for a well-performing query. This network is characterized by a dense and well-connected structure, indicating that the query will be robust to possible

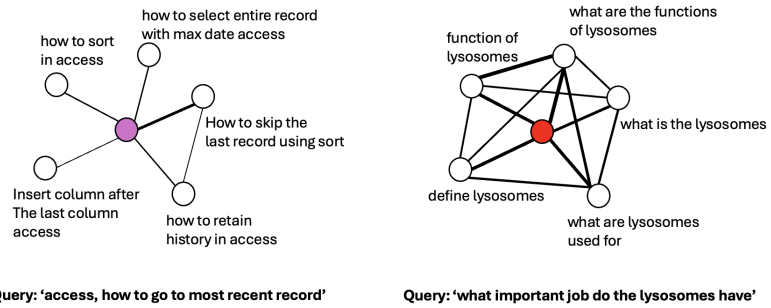

Fig. 3: Two sample `QFN` networks for two queries that have differing levels of difficulty. The left network illustrates a poorly-performing query with high sparsity, while the right network displays a well-performing query with a dense structure.

noisy perturbations. The contrast between the two networks, i.e., high sparsity for the poorly performing query and the high density for the well-performing query, points to the potential utility of the `QFN` and `DFN` networks for deriving context-sensitive disturbances. By tailoring the level of disturbance injected to the query based on its context in the focal networks, our proposed approach aims to capture the potential variabilities that queries may encounter in the real-world and hence more accurately estimate query performance in practice.

In the following, we elaborate on the potential host of disturbance measures that can be derived from the focal networks.

3.5 Network-Based Disturbance Measures

Our objective is to utilize focal network characteristics to generate contextualize disturbances, i.e., σ in Equation 4. We define three different forms of disturbances based on *node-based, edge-based*, and *cluster-based* of the focal networks. The set of all metrics used for disturbances are introduced in Table 1.

Node-based Disturbances. We postulate that the position of a node within the network is a strong sign of its resilience to disturbance and hence its robustness. For instance, if a query exhibits popularity within the focal network, this can indicate that the query may serve as a bridge between various nodes in the network; implying that minor perturbations would not significantly disrupt its connections with other nodes in the network. High robustness in this context means that the query can withstand variations while maintaining its role in the network. As such, node-based measures can provide insight into the structural importance and connectivity of the query node within the focal network, which may be relevant factors in determining the appropriate disturbance for assessing query robustness. Table 1 describes node-based measures including *Semantic Network Size*, *degree centrality*, *closeness centrality*, and *PageRank*.

Edge-based Disturbances. We suggest that focal networks with stronger connections are more resilient to disturbance; therefore, representing queries that are more robust to perturbations. Conversely, focal network with weaker edge connections can be an indication of a less robust query that can be susceptible to high variation in the face of small perturbations. Given edge-based network metrics focus on evaluating the resilience and consistency of connections (edges) between nodes, they have the potential to assess the structure and strength of connections within a focal network. For instance, a query that is not only highly connected in the focal network but is

Table 1: Network-based disturbance measures.

	Metric Name	Formula	Description
Node-based	Semantic Network Size (SNS)	$\lvert V_\zeta \rvert$	A large network size surrounding a particular query node suggests that the query is embedded in a dense semantic space, potentially enhancing its robustness due to multiple relational pathways.
	Degree Centrality (DC)	$\lvert\{e_{i,j} \in E_\zeta\}\rvert$	Extent to which a query connects to other nodes, identifying whether the node is popular in the focal network. Popular queries are likely to be robust to disturbance.
	Closeness Centrality (CC)	$\left[\sum_{j \in V_\zeta, j \neq i} d(q,j)\right]^{-1}$	Highlights how quickly it is possible to move from the query node to others in the focal network. Queries with high connectivity maintain short paths even when perturbed and remain robust.
	PageRank (PR)	$\frac{1-d}{\lvert \mathbb{V}_\zeta \rvert} + d \sum_{j \in \mathbb{V}_\zeta} \frac{PR(j)}{\deg(j)}$	A high PR measures how well a query is not only connected to many other nodes but also connected to other highly connected ones. A high PageRank may be a sign of being robustness to disturbance.
Edge-based	Connectivity Score (CS)	$\lvert E_\zeta(q) \rvert$	A high edge count indicates that the query is well-connected, suggesting that it will be less sensitive to noisy perturbations.
	Query Connectivity Strength (QCS)	$\sum_{e_{q,j} \in E_\zeta} \gamma(e_{q,j})$	A high connectivity strength value indicates numerous relevant connections within the network, which can be a sign of the robustness of query.
	Average Query Connectivity (AQC)	$\frac{1}{\lvert E_\zeta(q) \rvert} \sum_{e_{q,j} \in E_\zeta} \gamma(e_{q,j})$	This metric computes the average strength of connections for the query node where high average strengths point to more resilient queries against disturbance.
	Rare Path Index (RPI)	$\frac{1}{\lvert E_\zeta(q) \rvert} \sum_{e_{q,j} \in E_\zeta} \frac{1}{\gamma(e_{q,j})}$	A High RPI suggests that specific connections remain stable and relevant when query is altered.
Cluster-based	Inter-Cluster Connectivity (ICC)	$\frac{1}{\binom{K}{2}} \sum_{I \neq j} \max_{i,j} \gamma(e_{i,j})$	The strength of connections between different clusters reflects the overall interconnectedness of the network pointing to resilience against disturbance.
	Centroid Cluster Weight (CCW)	$\frac{\sum_{(u,v) \in E_{C_k}} w(u,v)}{\lvert E_{C_k} \rvert}$	measures strength within the most cohesive cluster, a sign of robustness against disturbances in at least one aspect of the query.

also connected to other highly connected nodes can point to a query that is highly resilient to noisy perturbations, and hence robust in retrieval performance. Table 1 offers the motivation behind edge-based measures such as *Connectivity Score*, *Query Connectivity Strength*, *Average Query Connectivity* , and *Rare Path Index*.

Cluster-based Disturbances. In this class of disturbances, we are motivated by earlier empirical research that suggests that semantic tightness or diversity of retrieval results of a query can be an indication of query performance [5, 48]. Low variance in clusters surrounding the query can suggest that the query is tightly related to others, indicating a high degree of robustness. Conversely, high variance within surrounding clusters might signal a broader, more eclectic set of nearest queries or documents, which may indicate susceptibility to semantic shifts or varying interpretations of the query. Understanding the overall interconnectedness of the focal networks may allow us to gauge the potential differing semantic interpretations of the query. To this end and within the focal network $\zeta(q)$, we apply clustering to obtain K clusters from the nodes in $\mathbb{V}_\zeta$. The center of each cluster C_k is the centroid, calculated as the average of

the embeddings of the nodes that make up the cluster. A high cohesion within clusters indicates that all nodes in the cluster are semantically related, suggesting strong group specificity and robustness to semantic drift. Additionally, well-distinguishable clusters where nodes are well matched to their own cluster and distinct from other clusters indicate a well-structured semantic network. Such a network suggests that the main query is robust, as the clusters are distinct enough such that noisy perturbations cannot cause significant query drift. Table 1 provides an overview of cluster-based measures such as *Centroid Cluster Weight* and *Inter-cluster Connectivity*.

4 Experimental Setup

Datasets. We evaluate our proposed approach on the widely adopted MS MARCO (Microsoft Machine Reading Comprehension) V1 passage collection, which includes 8.8 million passages over 500,000 queries, each associated with at least one relevance-judged document. This dataset has been extensively used for large-scale training of downstream tasks, such as QPP. We adopted three query sets associated with the MS MARCO V1 passage collection from the TREC Deep Learning Track datasets from 2019 (DL-2019) and 2020 (DL-2020), as well as the Deep Learning Hard (DL-Hard) query set, which includes more challenging queries with a higher number of labels per query. These datasets were chosen due to their extensive human-labeled relevance judgments per query, providing a robust basis for evaluating the performance of our approach. The actual performance on these three datasets is reported by the official evaluation metric on these query sets, i.e., nDCG@10.

Query Store. We integrated 808,731 queries from the MS MARCO V1 passage collection using the faiss library [49] to efficiently index and retrieve similar queries. The indexing method employed was Exact Search for L2 (IndexFlatL2), which stores vectors as fixed-size codes in an array. During search, all indexed vectors were decoded and compared to query vectors. However, for improved search speed, we utilized IndexPQ, which compares vectors in the compressed domain without decoding, facilitating faster query retrieval.

Evaluation Metrics. The evaluation of a query performance predictor is typically performed by measuring the correlation between the predicted and actual query performance on a set of queries. Essentially, given two lists of queries performances—the actual performance and the predicted performance—the correlation between these lists quantifies the quality of the prediction. As widely used for QPP evaluation [4], we report common linear and rank-based correlation metrics, including Pearson's ρ linear correlation, as well as Kendall's τ and Spearman's ρ rank-based correlations. Higher correlation values indicate more accurate predictive performance.

Retrievers. In order to show the generalizability of our approach, we report the performance of our work on the QPP task for two different state of the art retrievers, namely S-BERT [50] and ANCE [51].

Injected Disturbance. White Gaussian Noise was introduced by setting $\mu = 0$ and selecting σ values through the metrics introduced in 3.5. The range for σ is normalized to (0,1]. We performed an element-wise addition of the disturbance to the embedded query vector and retrieved thsection 3.5. e original and perturbed query from the embedded document index using Faiss library [49].

Pruning. For building the focal networks, the ϵ threshold defines the minimum similarity required to establish a connection between nodes in the network, with edges having lower weights than ϵ being pruned from the network. We analyze the impact of ϵ on the performance of our proposed approach.
Ranked Bias Overlap (RBO). To measure the differences between the two retrieved ranked lists of documents for the original query and the perturbed query, we first retrieved the top 1,000 documents for both queries. We then calculate the similarity between the two retrieved lists using RBO. As suggested in previous studies [37, 52], we set p to 0.95.
Codebase. We note that for reproducibility purposes, our code and data is publicly available at https://anonymous.4open.science/r/ADG-QPP-6529 .

4.1 Baselines

We compare our proposed method against several state-of-the-art supervised and unsupervised post-retrieval QPP methods:
Unsupervised QPP Baselines. Unsupervised term-statistics QPP baselines we consider in this paper include: (1) *Weighted Information Gain (WIG)* [6]: Calculates the mean divergence of the retrieval score of the top-ranked documents from the corpus to predict query performance. (2) *Clarity* [26]: Measures the coherence of term distribution in the retrieved list of documents with respect to the corpus by comparing the language models induced from the retrieved documents and the entire corpus. (3) *Query Feedback (QF)* [6]: This disturbance-based QPP method measures the overlap between the top retrieved documents from the original query and a disturbance-injected query created using pseudo-relevance feedback. (4) *Normalized Query Commitment (NQC)* [27]: This measure calculates the standard deviation of the rank scores of the retrieved documents normalized by the corpus score. The idea behind score-based metrics is that higher variance among the score of top-retrieved documents indicates easier distinguishability of relevant and non relevant documents and thus an easier query to satisfy. (5) *Score Magnitude and Variance (SMV)* [10]: Uses both the magnitude and variance of the scores of the retrieved documents to estimate query performance. (6) *Utility Estimation Framework based on NQC (UEF_{NQC})* [53]: In this approach, QPP is modeled as predicting the utility a user can derive from the retrieved result list. This prediction is based on estimating the similarity of the results to the ideal scenario, where all relevant documents are positioned at the highest ranks and all non-relevant documents are positioned below the relevant ones. (7) *Pairwise Rank Preference (QPP-PRP)* [54]: This method is designed for QPP on dense retrievers and estimates the consistency of the observed ranking with the predicted pairwise preferences. A higher agreement indicates better query performance.
AWGN (Dense-QPP). In this baseline, a constant perturbation is added across all the query representations. In their paper, the authors proposed to do the inference with N different noisy queries where N is suggested to be 30 to overcome the fluctuations in performance outcomes by averaging the predicted outputs. The need for multiple runs to achieve stable results bring up inefficiency challenges. In order to make this a fair comparison, we report the results with one run in Table 3.

Supervised QPP Baselines. The supervised QPP methods employed in this paper include: (1) *Neural-QPP* [29]: This was the first supervised neural-based QPP method which uses existing unsupervised QPP methods as signals for weakly-supervised learning to tackle QPP task. (2) *NQA-QPP* [32]: This method works by integrating three key components to predict the performance of non-factoid QA systems. The three components are i) the score-based component analyzes the distribution of retrieval scores assigned to the top-ranked answers, ii) the representation of the query iii) interactions of the representations of the question-answer component using BERT. (3) *BERT-QPP* [33]: This method utilizes fine-tuned BERT models to directly estimate the performance based on the quality of the retrieved documents. (4) *qppBERT-PL* [34]: It addresses QPP using BERT by leveraging both pointwise and listwise approaches. The method partitions the top-k retrieved documents into smaller chunks and later encodes the interactions between the query and different chunks with an LSTM to account for sequential information. (5) *Deep-QPP* [31]: This interaction-based model focuses on the semantic interactions between query terms and terms in the top-retrieved documents. The model employs a series of 2D convolutional layers to extract features from these interactions, followed by a feed-forward layer to predict the relative specificity.

5 Findings

5.1 Experimental Results

We compare the performance of the network-based metrics introduced in Table 1 on both Query Focal Network (QFN) and Document Focal Network (DFN) in Table 2 top and bottom sections, respectively. Additionally, we dig deeper into the performance of different groups of disturbance metrics, including Node-based, Edge-based, and Cluster-based metrics, across the three datasets: DL-Hard, DL-2019, and DL-2020.

5.1.1 Best-Performing Disturbance Metrics on `QFN`

For each of the three groups of network-based metrics, we select the best-performing one as the representative of that group. When building the focal network based solely on queries (`QFN`), we observe that among the node-based metrics, DC performs the best. This means that when injecting disturbance to query representations relative to their Degree Centrality across `QFN`, the RBO between the retrieved ranked list of the original query and the perturbed query shows the highest correlation across all the different node-based metrics. Similarly, for edge-based metrics, we observe that CS, QCS, and AQC all show competitive performance when derived from the Query Focal Network. For example, while AQC achieves the highest Pearson-ρ correlation of 0.709 on DL-2019, indicating an extremely high correlation for the QPP task, CS enjoys the highest correlation on DL-2020. Among cluster-based metrics, as reported in the last rows of Table 2, ICC performs better on DL-Hard by a wide margin, while CCW outperforms well on DL-2020. On DL-2019, their performance are similar and do not show statistically significant differences.

5.1.2 Best-Performing Disturbance Metrics on `DFN`

We compare the performance of different disturbance metrics obtained from `DFN` in the bottom part of Table 2. Among Node-based metrics applied to `DFN`, SNS and CC show

Table 2: Performance comparison between different variations of `ADG-QPP` on Query Focal Network (`QFN`, upper table) and Document Focal Network (`DFN`, bottom table) in terms of the Pearson-ρ (P-ρ), Kendall-τ (K-τ) and Spearman-ρ (S-ρ). All correlations are statistically significant at $\alpha = 0.5$. The highest value in each column of the subgroup is in bold.

`QFN`

		DL-Hard			DL-2019			DL-2020		
	Metrics	P−ρ	K−τ	S -ρ	P−ρ	K−τ	S −ρ	P−ρ	K−τ	S−ρ
Node	SNS	0.429	0.276	0.405	0.554	0.339	0.508	0.225	0.158	0.226
	CC	0.322	0.185	0.289	0.468	0.293	0.455	0.180	0.150	0.217
	DC	0.469	0.319	0.449	0.684	**0.439**	**0.598**	0.401	0.298	0.424
	PR	0.289	0.195	0.306	0.355	0.299	0.431	0.245	0.165	0.236
Edge	CS	**0.479**	0.339	0.460	0.539	0.326	0.450	**0.470**	**0.299**	**0.427**
	QCS	0.450	**0.352**	**0.494**	0.535	0.343	0.497	0.396	0.246	0.353
	AQC	0.420	0.288	0.408	**0.709**	0.419	0.587	0.324	0.166	0.247
	RPI	0.248	0.155	0.229	0.459	0.279	0.396	0.202	0.134	0.193
Cluster	ICC	0.466	0.324	0.456	0.589	0.357	0.503	0.307	0.196	0.272
	CCW	0.253	0.183	0.262	0.590	0.366	0.516	0.378	0.245	0.349

`DFN`

Node	SNS	0.452	0.344	0.479	**0.690**	0.419	0.573	0.240	0.186	0.283
	CC	0.389	0.278	0.393	0.685	0.401	0.572	0.367	**0.322**	**0.468**
	DC	0.513	0.362	0.510	0.611	0.419	0.579	0.221	0.158	0.252
	PR	0.346	0.225	0.328	0.239	0.182	0.245	0.267	0.169	0.272
Edge	CS	0.453	0.304	0.427	0.662	**0.425**	**0.605**	0.358	0.229	0.349
	QCS	0.467	0.357	0.520	0.587	0.375	0.498	0.213	0.144	0.207
	AQC	0.527	**0.405**	**0.556**	0.617	0.361	0.524	0.324	0.150	0.222
	RPI	0.417	0.294	0.424	0.417	0.304	0.434	0.292	0.245	0.366
Cluster	ICC	**0.564**	0.402	**0.556**	0.574	0.348	0.508	0.287	0.148	0.209
	CCW	0.476	0.309	0.451	0.613	0.383	0.519	**0.405**	0.210	0.295

very competitive performance. While both metrics perform similarly on DL-2019, CC outperforms SNS on DL-2020. Comparing the same metrics from `QFN` and `DFN`, we observe that including documents in the focal network is extremely beneficial for some metrics. For instance, comparing SNS in `QFN` versus `DFN`, SNS achieves a Pearson-ρ of 0.554 on DL-2019, which increased to 0.690 when documents were included in the focal network. However, adding documents does not always have the same level of impact. For most edge-based metrics, including documents did not help. We hypothesize that metrics capturing the centrality of queries within the focal network, such as centrality-based metrics, benefit more from the inclusion of documents. Overall, the metrics that show strong performance are consistent whether applied to `QFN` or `DFN`. For instance, ICC from the cluster-based group outperforms CCW in both `QFN` and `DFN`.

5.1.3 Comparison between QFN and DFN

In this section, we compare the results between two Networks QFN and DFN. The performance of edge-based metrics varies more between `QFN` and `DFN`. In `QFN`, metrics like CS and AQC show strong results, with CS achieving the highest Pearson-ρ of 0.470 on DL-2020. However, in `DFN`, while these metrics still perform well, their effectiveness is somewhat reduced, particularly for QCS, which shows a decline in performance on DL-2020 compared to its performance in `QFN`. This suggests that edge-based metrics, while useful, may be more sensitive to the network configuration, with their performance being more variable when documents are included in the network. In the cluster-based metrics, ICC stands out, particularly in `DFN`, where it achieves a

Table 3: Performance comparison between our best-performed proposed approach and SOTA baselines when predicting the performance of S-BERT dense retriever. All correlations are statistically significant at $\alpha = 0.5$ except the *italic* ones. The highest value in each column is in bold.

	DL-Hard			DL-2019			DL-2020		
	$P-\rho$	$K-\tau$	$S-\rho$	$P-\rho$	$K-\tau$	$S-\rho$	$P-\rho$	$K-\tau$	$S-\rho$
Clarity	0.232	0.110	0.162	0.217	0.111	0.151	0.196	0.137	0.188
QF	0.044	0.051	0.060	0.071	0.022	0.043	0.148	0.029	0.052
NQC	0.418	0.276	0.381	0.560	0.419	0.598	0.285	0.194	0.289
WIG	0.093	0.072	0.105	0.139	0.071	0.116	0.153	0.032	0.051
$n(\sigma_x)$	0.400	0.259	0.369	0.501	0.361	0.532	0.242	0.158	0.232
SMV	0.396	0.314	0.438	0.577	0.428	0.600	0.360	0.246	0.357
UEF	0.441	0.298	0.412	0.607	0.428	**0.601**	0.336	0.228	0.329
NeuralQPP	0.232	0.080	0.103	0.209	0.057	0.057	0.152	0.015	0.003
Pclarity_NQC	0.088	0.053	0.083	0.428	0.314	0.451	0.084	0.202	0.292
NQAQPP	0.113	0.240	0.359	0.269	0.129	0.160	0.221	0.159	0.234
BERTQPP	0.435	0.181	0.256	0.334	0.143	0.194	0.378	0.273	0.411
qppBERT-PL	0.405	0.171	0.225	0.299	0.131	0.183	0.344	0.224	0.335
Deep-QPP	0.096	*0.049*	*0.065*	0.139	0.103	0.106	0.262	0.197	0.291
QPP-PRP	0.181	0.099	0.144	0.203	0.204	0.281	0.181	0.143	0.219
AWGN (Dense-QPP)	0.371	0.254	0.384	0.572	0.414	0.574	0.331	0.199	0.318
Our Approach	**0.469**	**0.319**	**0.449**	**0.684**	**0.439**	0.598	**0.401**	**0.298**	**0.424**

Pearson-ρ of 0.564 on DL-Hard, outperforming other metrics within its group. While CCW shows some promise, especially on DL-2020, it fails to match the consistent and strong performance of ICC. Interestingly, in `DFN`, ICC appears to be better suited to networks with richer document-query interactions. When comparing node-based metrics across `QFN` and `DFN`, we observe distinct patterns in their effectiveness. Degree Centrality (DC) emerges as the most consistent performer in `QFN`, with particularly strong results across all datasets. For instance, DC achieves a Pearson-ρ of 0.684 on DL-2019 and 0.469 on DL-Hard in `QFN`, significantly outperforming other node-based metrics like SNS and CC. In contrast, while DC also performs well in `DFN`, its effectiveness slightly diminishes compared to its performance in `QFN`, with a Pearson-ρ of 0.611 on DL-2019. This suggests that DC is more robust and stable when applied within `QFN`, where the focus is solely on query interactions without the inclusion of documents. While other metrics show variability in their performance when moving from `QFN` to `DFN`, DC maintains a high level of performance across all datasets in `QFN` marking it as the chosen metric for the upcoming experiments.

5.1.4 Comparison with Baselines

In Table 3, we compare our best-performing variation from Table 2, i.e., the DC metric from the node-based group on `QFN`, with various state-of-the-art unsupervised and supervised QPP baselines. Based on the results, we make several observations:
(1) We first draw attention to comparing the performance of the last two rows, i.e., injecting uniform noise to all queries (AWGN) versus the adaptive disturbance, which injects different disturbance per query based on its focal network. Comparing AWGN with the best-performing adaptive disturbance method shows that injecting personalized disturbance per query leads to higher performance than uniformly injecting the

Table 4: Performance comparison between `ADG-QPP` and SOTA baselines when predicting the performance of ANCE dense retriever. All correlations are statistically significant at $\alpha = 0.5$ except the *italic* ones. The highest value in each column is in bold.

	DL-Hard			DL-2019			DL-2020		
	$P-\rho$	$K-\tau$	$S-\rho$	$P-\rho$	$K-\tau$	$S-\rho$	$P-\rho$	$K-\tau$	$S-\rho$
Clarity	0.221	0.230	0.331	0.353	0.237	0.344	0.281	0.215	0.320
QF	0.155	0.118	0.165	0.129	0.098	0.148	0.283	0.257	0.361
NQC	0.235	0.300	0.424	0.504	0.335	0.446	0.442	0.328	0.449
WIG	0.166	0.133	0.189	0.159	0.120	0.186	0.230	0.195	0.275
$n(\sigma_x)$	0.242	0.197	0.275	0.361	0.233	0.347	0.199	0.181	0.262
SMV	0.174	0.290	0.438	0.518	0.337	0.447	0.417	0.328	0.456
UEF	0.229	0.310	**0.458**	0.520	0.350	0.453	0.458	0.348	0.497
NeuralQPP	0.142	0.063	0.099	0.047	0.004	0.035	0.220	0.087	0.120
Pclarity_NQC	0.157	0.172	0.112	0.383	0.247	0.402	0.209	0.308	0.174
NQAQPP	0.334	0.264	0.360	0.115	0.140	0.192	0.147	0.152	0.204
BERTQPP	0.213	0.143	0.049	0.144	0.165	0.232	0.362	0.268	0.381
qppBERT-PL	0.303	0.254	0.342	0.229	0.189	0.247	0.313	0.205	0.280
Deep-QPP	0.154	0.131	0.175	0.183	0.195	0.264	0.220	0.127	0.194
QPP-PRP	0.016	0.004	0.033	0.096	0.086	0.107	0.201	0.170	0.278
AWGN	0.315	0.310	0.456	0.528	0.363	0.462	0.443	0.332	0.483
Our Approach	**0.432**	**0.316**	0.423	**0.535**	**0.368**	**0.466**	**0.543**	**0.361**	**0.521**

same disturbance to all queries. This emphasizes that to test the robustness of different queries fairly and in the same context, different levels of disturbance are needed.

(2) Our proposed approach outperforms all the baselines on all datasets except on DL-2019 in terms of Spearman-ρ. While our approach achieved a Spearman-ρ of 0.598, UEF obtained 0.601. We conducted paired t-test significance test to determine whether this difference is statistically significant. The results show that this difference is not statistically significant. As such, we can conclude that our proposed approach is capable of achieving the best performance across all datasets and all evaluation metrics.

(3) Some baselines show competitive performance on specific datasets, but our approach demonstrates the most robust performance across all three datasets and metrics. For example, BERT-QPP achieved a competitive Pearson-ρ of 0.435 on DL-Hard, compared to our method's 0.469. However, BERT-QPP's Kendall-τ was much lower than ours (0.181 vs. 0.319). The most competitive baseline, UEF, shows similar performance on DL-Hard and DL-2019. However, on DL-2020, the margin is significant, and our method outperformed UEF by a margin of 0.1 in terms of Spearman-ρ.

(4) Among the unsupervised QPP baselines, those that rely on the distribution of retrieval scores, such as NQC, SMV, and $n(\sigma_x)$, exhibit better performance compared to methods, such as Clarity and QF, that sometimes fail to achieve statistically significant correlations with actual query performance. This observation aligns with previous studies [38]. Within the supervised QPP baselines, NeuralQPP shows extremely low correlation values, likely due to its reliance on weak signals from unsupervised methods, which are not robust indicators for QPP in the context of dense retrievers. Additionally, NeuralQPP demands large training data volumes and demonstrates poor performance with limited data availability. Similarly, while Deep-QPP has shown effectiveness with sparse retrievers, it fails to maintain consistent performance with dense

retrievers. On the other hand, methods like NQAQPP, BERTQPP, and qppBERT-PL display higher degrees of correlation with actual query performance, although still showing lower performance than our proposed approach.
(5) Previous studies have shown that neural-based methods generally show reliable performance when predicting the performance of sparse retrievers such as BM25. However, they do not perform well when predicting the performance of dense retrievers, as also observed in [38, 54]. This could be due to the inability to train these methods from scratch to learn the performance of dense retrievers due to overlapping data between the ranker and the QPP method, resulting in a performance drop in a zero-shot setting. Therefore, using an unsupervised neural-based method leverages the semantic space without the need for additional training.
(6) Other than our method, the only other unsupervised neural-based method, QPP-PRP, shows lower performance compared to our method. For example, on DL-Hard, we achieved Pearson-ρ and Kendall-τ of 0.469 and 0.319, respectively, while QPP-PRP only achieved 0.181 and 0.099, respectively.

5.2 Robustness Analysis

5.2.1 The Impact of Dense Retrievers

Here, we study the generalizability of our proposed method in predicting the performance of different dense retrievers. In Table 4, we have shown that our proposed method outperforms other state-of-the-art baselines when predicting the performance of S-BERT [50] as a state-of-the-art dense retriever. Here, we aim to determine if it can also predict the performance of another dense retriever effectively. To do so, we consider Approximate Nearest Neighbor Negative Contrastive Estimation (ANCE) [51], which has shown high performance on various downstream NLP and IR tasks.

The results in Table 4 indicate that our proposed approach achieves high correlation not only on S-BERT but also on ANCE. This emphasizes the generalizability of our approach, showing that our method is not limited to a single dense retriever, thus demonstrating robustness and generalizability. Our proposed method not only outperforms all the baselines in predicting the performance of ANCE but also achieves a high and meaningful range of correlations, which are reliable and significant. On all three datasets and in terms of all three evaluation metrics, our proposed approach outperformed all the baselines except on DL-Hard in terms of Spearman-ρ. UEF, which has also shown reliable performance in predicting the performance of S-BERT, exhibits a high correlation for predicting ANCE. In addition, it is noteworthy that although adding disturbance uniformly to all queries shows a significant correlation with the actual performance, leveraging the adaptive disturbance method can still improve it and outperforms AWGN. In summary, while the choice of the dense retriever has been shown to impact the performance of QPP methods, our method consistently shows the highest and most stable performance across all datasets for both dense retrievers.

5.2.2 The Impact of Focal Network Pruning

As highlighted in Section 4, when building the focal networks, we prune edges with weights below a certain threshold ϵ. The threshold ϵ represents the minimum similarity required to establish a connection between nodes (queries or documents) in the focal network. Here, we analyze the impact of ϵ on the performance of `ADG-QPP`. Figure

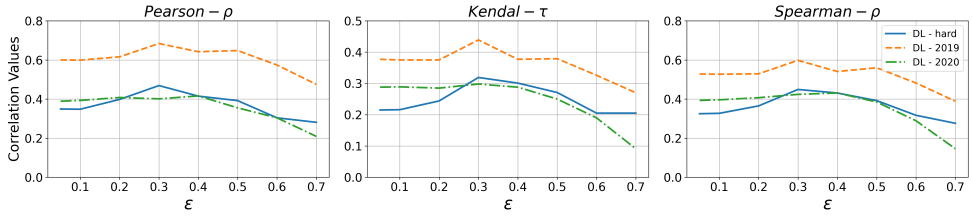

Fig. 4: Impact of epsilon on the performance of `ADG-QPP`. The Epsilon (ϵ) values (X-axis) vs performance (Kendall's τ and Pearson's ρ) of `ADG-QPP` (Y-axis).

4 presents the Pearson-ρ, Kendall-τ and Spearman-ρ correlations from left to right across different pruning thresholds (ϵ values) on the three datasets of our experiments.

Higher ϵ results in a sparser focal network, while $\epsilon = 0$ means no edges are pruned. As depicted in these plots, by pruning more edges, the performance of `ADG-QPP` increases. However, after a certain threshold, the performance starts to drop. We hypothesize that this decline occurs because the focal networks become too sparse, leaving insufficient meaningful edges. For instance, on all three datasets, we observe that all three correlations improve with increasing ϵ values to 0.3, beyond which the performance starts to decline. Choosing an appropriate ϵ value is critical for constructing a robust semantic network. Too low an ϵ value may result in a sparse network, missing important semantic relationships, while too high an ϵ value may introduce redundancy by linking semantically distant nodes. The suitable ϵ value balances the inclusion of meaningful semantic relationships without introducing excessive disturbance and redundancy. Having said that and based on these results, we selected an ϵ value of 0.3 for our experiments, as it achieved a consistent and robust correlation with the actual performance across all datasets.

6 Concluding Remarks

In this paper, we have introduced a query performance predictor designed specifically for dense retrieval methods. Our proposed approach is premised on the foundation that queries that exhibit robustness towards noisy perturbations are likely to be easier to be addressed by retrieval methods. On this basis, we propose how customized disturbance can be generated and injected into the embedding representation on a per query basis, which can then be used for retrieval and the estimation of the performance of the query. We have shown through extensive experiments on three widely used datasetsthat our proposed approach is able to exhibit consistently better performance for estimating the retrieval effectiveness of queries on two different neural retrieval methods.

References

[1] Roitman, H.: Ictir tutorial: Modern query performance prediction: Theory and practice. ICTIR '20. Association for Computing Machinery, New York, NY, USA (2020)

[2] Hashemi, H., Zamani, H., Croft, W.B.: Performance prediction for non-factoid question answering. ICTIR '19. Association for Computing Machinery, New York, NY, USA (2019)

[3] Roitman, H., Mass, Y., Feigenblat, G., Shraga, R.: Query performance prediction for multifield document retrieval. ICTIR '20. Association for Computing Machinery, New York, NY, USA (2020)

[4] Carmel, D., Yom-Tov, E.: Estimating the query difficulty for information retrieval. In: SIGIR (2010)
[5] Arabzadeh, N.e.a.: Neural embedding-based specificity metrics for pre-retrieval query performance prediction. IP&M Journal **57**(4) (2020)
[6] Zhou, Y., Croft, W.B.: Query performance prediction in web search environments. In: SIGIR, pp. 543–550 (2007)
[7] Mackie, I., Dalton, J., Yates, A.: How deep is your learning: the dl-hard annotated deep learning dataset. In: SIGIR (2021)
[8] Zhao, Y., Scholer, F., Tsegay, Y.: Effective pre-retrieval query performance prediction using similarity and variability evidence. In: ECIR (2008)
[9] Hauff, C., Hiemstra, D., Jong, F.: A survey of pre-retrieval query performance predictors. In: CIKM (2008)
[10] Tao, Y., Wu, S.: Query performance prediction by considering score magnitude and variance together. In: CIKM, pp. 1891–1894 (2014)
[11] Pérez-Iglesias, J., Araujo, L.: Standard deviation as a query hardness estimator. In: SPIRE, pp. 207–212 (2010). Springer
[12] Sarnikar, S., Zhang, Z., Zhao, J.L.: Query-performance prediction for effective query routing in domain-specific repositories. Journal of the Association for Information Science and Technology **65**(8), 1597–1614 (2014)
[13] Pal, D., Ganguly, D.: Effective query formulation in conversation contextualization: A query specificity-based approach. In: ICTIR, pp. 177–183 (2021)
[14] Deveaud, R., Mothe, J., Ullah, M.Z., Nie, J.-Y.: Learning to adaptively rank document retrieval system configurations. ACM TOIS **37**(1), 1–41 (2018)
[15] Deveaud, R., Mothe, J., Nie, J.-Y.: Learning to rank system configurations. In: CIKM, pp. 2001–2004 (2016)
[16] Raiber, F., Kurland, O.: Query-performance prediction: setting the expectations straight. In: SIGIR (2014)
[17] Ganguly, D., Yilmaz, E.: Query-specific variable depth pooling via query performance prediction. In: SIGIR, pp. 2303–2307 (2023)
[18] Tonellotto, N., Macdonald, C., Ounis, I.: Efficient and effective retrieval using selective pruning. In: WSDM, pp. 63–72 (2013)
[19] Arabzadeh, N., Seifikar, M., Clarke, C.L.: Unsupervised question clarity prediction through retrieved item coherency. In: CIKM, pp. 3811–3816 (2022)
[20] Roitman, H., Erera, S., Feigenblat, G.: A study of query performance prediction for answer quality determination. In: ICTIR, pp. 43–46 (2019)
[21] Zamani, H., Dumais, S., Craswell, N., Bennett, P., Lueck, G.: Generating clarifying questions for information retrieval. In: WWW (2020)
[22] Jones, K.S., Walker, S., Robertson, S.E.: A probabilistic model of information retrieval: development and comparative experiments: Part 2. IPM (2000)
[23] Cronen-Townsend, S., Zhou, Y., Croft, W.B.: Predicting query performance. In: SIGIR, pp. 299–306 (2002)
[24] Arabzadeh, N., Bigdeli, A., Zihayat, M., Bagheri, E.: Query performance prediction through retrieval coherency. In: ECIR (2021). Springer

[25] Lavrenko, V., Croft, W.B.: Relevance based language models. In: SIGIR. SIGIR '01. ACM, New York, NY, USA (2001)
[26] Cronen-Townsend, S., Zhou, Y., Croft, W.B.: Predicting query performance. In: SIGIR (2002)
[27] Shtok, A., Kurland, O., Carmel, D., Raiber, F., Markovits, G.: Predicting query performance by query-drift estimation. TOIS **30**(2), 1–35 (2012)
[28] Cummins, R., Jose, J., O'Riordan, C.: Improved query performance prediction using standard deviation. In: SIGIR, pp. 1089–1090 (2011)
[29] Zamani, H., Croft, W.B., Culpepper, J.S.: Neural query performance prediction using weak supervision from multiple signals. In: SIGIR, pp. 105–114 (2018)
[30] Arabzadeh, N., Zarrinkalam, F., Jovanovic, J., Bagheri, E.: Neural embedding-based metrics for pre-retrieval query performance prediction. In: ECIR (2020)
[31] Datta, S., Ganguly, D., Greene, D., Mitra, M.: Deep-qpp: A pairwise interaction-based deep learning model for supervised query performance prediction. In: WSDM (2022)
[32] Hashemi, H., Zamani, H., Croft, W.B.: Performance prediction for non-factoid question answering. In: ICTIR, pp. 55–58 (2019)
[33] Arabzadeh, N., Khodabakhsh, M., Bagheri, E.: Bert-qpp: Contextualized pre-trained transformers for query performance prediction. In: CIKM (2021)
[34] Datta, S., MacAvaney, S., Ganguly, D., Greene, D.: A'pointwise-query, listwise-document'based query performance prediction approach. In: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 2148–2153 (2022)
[35] Zhai, C.: Statistical Language Models for Information Retrieval vol. 1, (2008). https://doi.org/10.3115/1614181.1614183
[36] Ponte, J.M., Croft, W.B.: A language modeling approach to information retrieval. In: ACM SIGIR Forum, vol. 51, pp. 202–208 (2017). ACM New York, NY, USA
[37] Arabzadeh, N., Hamidi Rad, R., Khodabakhsh, M., Bagheri, E.: Noisy perturbations for estimating query difficulty in dense retrievers. In: ICIKM (2023)
[38] Faggioli, G., Formal, T., Marchesin, S., Clinchant, S., Ferro, N., Piwowarski, B.: Query performance prediction for neural ir: Are we there yet? In: ECIR (2023)
[39] Zhou, Y., Croft, W.B.: Ranking robustness: a novel framework to predict query performance. In: CIKM (2006)
[40] Arabzadeh, N., Zarrinkalam, F., Jovanovic, J., Bagheri, E.: Geometric estimation of specificity within embedding spaces. In: CIKM, pp. 2109–2112 (2019)
[41] Datta, S., Ganguly, D., Mitra, M., Greene, D.: A relative information gain-based query performance prediction framework with generated query variants. TOIS **41**(2), 1–31 (2022)
[42] Webber, W., Moffat, A., Zobel, J.: A similarity measure for indefinite rankings. ACM Transactions on Information Systems (TOIS) **28**(4), 1–38 (2010)
[43] Lyon, A.: Why are normal distributions normal? The British Journal for the Philosophy of Science (2014)
[44] Casella, G., Berger, R.: Statistical Inference, (2024)

[45] Carlson, A.B.: communication systems: an introduction to signal noise in electrical communication (2002)
[46] Sorscher, B., Ganguli, S., Sompolinsky, H.: Neural representational geometry underlies few-shot concept learning. PNAS (2022)
[47] Faggioli, G., Ferro, N., Muntean, C.I., Perego, R., Tonellotto, N.: A geometric framework for query performance prediction in conversational search (2023)
[48] Roy, D., Ganguly, D., Mitra, M., Jones, G.J.: Estimating gaussian mixture models in the local neighbourhood of embedded word vectors for query performance prediction. IPM (2019)
[49] Johnson, J., Douze, M., Jégou, H.: Billion-scale similarity search with gpus. IEEE Transactions on Big Data **7**(3), 535–547 (2019)
[50] Reimers, N., Gurevych, I.: Sentence-bert: Sentence embeddings using siamese bert-networks. ArXiv (2019)
[51] Xiong, L., Xiong, C., Li, Y., Tang, K.-F., Liu, J., Bennett, P., Ahmed, J., Overwijk, A.: Approximate nearest neighbor negative contrastive learning for dense text retrieval. arXiv preprint arXiv:2007.00808 (2020)
[52] Clarke, C.L., Vtyurina, A., Smucker, M.D.: Assessing top-preferences. ACM Transactions on Information Systems (TOIS) **39**(3), 1–21 (2021)
[53] Shtok, A., Kurland, O., Carmel, D.: Using statistical decision theory and relevance models for query-performance prediction. In: SIGIR, pp. 259–266 (2010)
[54] Singh, A., Ganguly, D., Datta, S.: Unsupervised query performance prediction for neural models utilising pairwise rank preferences. def (2023)