Information Retrieval Journal
https://doi.org/10.1007/s10791-022-09405-y

SPECIAL ISSUE ON ECIR 2021

CEQE to SQET: A study of contextualized embeddings for query expansion

Shahrzad Naseri[1] · **Jeffrey Dalton**[2] · **Andrew Yates**[3] · **James Allan**[1]

Received: 9 July 2021 / Accepted: 5 February 2022

Abstract
In this work, we study recent advances in context-sensitive language models for the task of query expansion. We study the behavior of existing and new approaches for lexical word-based expansion in both unsupervised and supervised contexts. For unsupervised models, we study the behavior of the Contextualized Embeddings for Query Expansion (CEQE) model. We introduce a new model, Supervised Contextualized Query Expansion with Transformers (SQET) that performs expansion as a supervised classification task and leverages context in pseudo-relevant results. We study the behavior of these expansion approaches for the tasks of ad-hoc document and passage retrieval. We conduct experiments combining expansion with probabilistic retrieval models as well as neural document ranking models. We evaluate expansion effectiveness on three standard TREC collections: Robust, Complex Answer Retrieval, and Deep Learning. We analyze the results of extrinsic retrieval effectiveness, intrinsic ability to rank expansion terms, and perform a qualitative analysis of the differences between the methods. We find out CEQE statically significantly outperforms static embeddings across all three datasets for Recall@1000. Moreover, CEQE outperforms static embedding-based expansion methods on multiple collections (by up to 18% on Robust and 31% on Deep Learning on average precision) and also improves over proven probabilistic pseudo-relevance feedback (PRF) models. SQET outperforms CEQE by 6% in P@20 on the intrinsic term ranking evaluation and is approximately as effective in retrieval performance. Models incorporating neural and CEQE-based expansion score achieves gains of up to 5% in P@20 and 2% in AP on Robust over the state-of-the-art transformer-based re-ranking model, Birch.

Keywords Query expansion · Contextualized language models · Embeddings

This is an extension of CEQE: Contextualized Embeddings for Query Expansion, published in ECIR 2021. This work has several key differences and extensions over previous work. First, it adds additional experimental results of CEQE on a third TREC dataset, Complex Answer Retrieval (CAR). These experiments include unsupervised retrieval and intrinsic evaluation results. Second, it proposes a new and previously unpublished contextual expansion model, SQET that is a discriminatively trained supervised model that classifies expansion terms. The behavior of CEQE and SQET are compared on the Robust test collection for both extrinsic retrieval effectiveness and intrinsic ability to rank expansion terms. Finally, a qualitative comparison of CEQE and SQET terms as well as a discussion of per-layer CEQE behavior is provided. We estimate 30–50% of this work is new or significantly updated over the original paper.

Extended author information available on the last page of the article

Published online: 22 March 2022

1 Introduction

Recently there is a significant shift in text processing from high-dimensional word-based representations to ones based on continuous low-dimensional vectors. However, fundamentally both are static – each word has a *context-independent* or static representation. The fundamental challenge of polysemy remains. Recent language models aim to address this, namely ELMo (Peters et al. 2018) and BERT (Devlin et al. 2019), by creating *context-dependent* representations that depend on the surrounding context. The power of contextualized models comes from this ability to disambiguate and generate distinctive representations for terms with the same lexical form. Contextualized language models that represent a word in context demonstrate significant improvements over previous static models. They exhibit transfer learning capability which means they are widely adopted for retrieval and NLP. Supervised ranking models derived from them, such as CEDR (MacAvaney et al. 2019) and T5 (Nogueira et al. 2020), are the current state-of-the-art learning-to-rank methods for a wide range of retrieval and QA benchmarks.

We apply these models to one of the fundamental challenges in retrieval, query expansion. Widely used pseudo-relevance feedback (PRF) (Lavrenko & Croft, 2001; Lv & Zhai, 2009; Zhai & Lafferty, 2001) methods perform an unsupervised expansion, by *assuming* the top retrieved documents are relevant and using this information to build a probability distribution over terms and update the query model. One of the early successful supervised approaches by Cao et al. (2008) raises the issue that not all the most frequent terms in the PRF distribution are good for expansion. They use a feature-based model to predict what terms should be used for expansion, classifying terms as positive, negative, or neutral to maximize the number of predicted positive terms. Despite this work, effective models remain a significant challenge. In the TREC Complex Answer Retrieval (CAR) track approximately 30% of all topics have a recall score of zero for widely used candidate matching methods such as BM25 and Query Likelihood, and PRF models do not have sufficient density of relevant results to be effective. Advancements in retrieval require more effective core matching algorithms to improve recall for neural ranking methods. No amount of reranking irrelevant results will provide relevance gains. Moreover, since the PRF methods assume the top retrieved documents are relevant, the quality of the initial rank list of retrieved documents gains extra importance. Therefore if a high precision ranked list is used, PRF approaches will be much more efficient. We can take advantage of a high precision neural reranker to obtain such a high-precision ranked list.

In this work we leverage contextualized word representations to address this problem. We extend our CEQE model[1]Naseri et al. (2021) which is the first work to develop an unsupervised contextualized query expansion model based on pseudo-relevance feedback. We extend our work in the following directions:

- We study the CEQE model's behavior on new collections, including TREC CAR, by extrinsic (adhoc retrieval) as well as intrinsic (its effectiveness to rank positive expansion terms) evaluation.
- We propose and evaluate a new model for query expansion, Supervised Contextualized Query Expansion with Transformers (SQET).[2] This model builds on the Transformer based architecture and treats the expansion problem as a supervised

[1] https://github.com/sherinaseri/ceqe-release.
[2] https://github.com/sherinaseri/sqet-release.

classification task leveraging the context words around the candidate expansion terms. Unlike previous hand-crafted feature approaches this uses pre-trained contextual term representations.
- Expand the qualitative analysis by providing query-by-query analysis of the CEQE model for CAR and TREC Deep Learning collections as well as a qualitative comparison of CEQE and SQET for Robust.

Overall, we study and propose new contextualized expansion methods to address the task of core matching building on proven pseudo-relevance feedback (PRF) techniques from probabilistic Language Modeling and extending them to effectively leverage contextual word representations. Further, we investigate the effect of applying them in combination with state-of-the-art neural re-ranking models. Our work addresses core research questions (RQ) in contextualized query expansion:

- **RQ1** How can contextualized representations be effectively leveraged to improve state-of-the-art unsupervised and supervised query expansion methods?
- **RQ2** How effective are neural reranking methods when performed after query expansion?
- **RQ3** How effective are query expansion methods after a first pass of high-precision neural re-ranking?

We study these questions with empirical experiments on standard TREC test collections: Robust, Complex Answer Retrieval, and Deep Learning 2019. The results on these test collections demonstrate that variations of CEQE significantly outperform previous static embedding models (based on GLoVe) in extrinsic retrieval effectiveness by approximately 18% MAP on Robust04 and 31% on TREC Deep Learning 2019 and 6-9% for recall@1000 across all datasets. We also find that SQET performs comparably to CEQE in extrinsic retrieval and slightly outperforms it for the task of term selection.

This work makes several new contributions to methods and understanding of contextualized representations for query expansion and relevance feedback:

- We develop a new supervised contextualized query expansion method, SQET, that performs supervised term classification based on PRF contexts using contextual representations.
- We compare the behavior of supervised and unsupervised models for expansion on standard TREC collections.
- We extend the study of CEQE to new test collections (TREC CAR), as well as perform a comparative analysis of key factors in its behavior.

The remainder of this paper is structured as follows. First, in Sect. 2 we provide background and related work on the use of static and contextualized word representations as well as relevance feedback models using them. Next, we introduce the CEQE and SQET contextualized expansion models in Sect. 3. Then in Sect. 5, we present the empirical evaluation of our proposed models on multiple test collections for intrinsic term selection and extrinsic retrieval effectiveness. Finally, in Sect. 6 we perform a qualitative study of factors that impact their effectiveness and highlight the differences in their behavior.

2 Background and related work

Query Expansion A widely used approach to improve recall uses query expansion from relevance feedback that takes a user's judgment of a result's relevance and uses it to build an updated query model (Rocchio 1971). Pseudo-relevance feedback (PRF) (Lavrenko & Croft, 2001; Lv & Zhai, 2009; Zhai & Lafferty, 2001) approaches perform this task automatically, *assuming* the top documents are relevant. For the unsupervised model, we build on these proven approaches based on static representations and extend them to contextualized representations. Padaki et al. (2020) investigate BERT's performance when using expanded queries and find that expansion that preserves some linguistic structure is preferable to expanding with keywords.

Embedding-based Expansion Another approach for query expansion incorporates static embeddings (Pennington et al. 2014; Mikolov et al. 2013) to find the relevant terms to the query, because embeddings promise to capture the semantic similarity between terms and are used in different ways to expand queries (Diaz et al., 2016; Kuzi et al., 2016; Zamani & Croft, 2016, 2017; Dalton et al., 2019; Roy et al., 2016; Naseri et al., 2018). These word embeddings, such as Word2Vec, GloVe, and others, learn a static word embedding for each term regardless of the context. Most basic models fail to address polysemy and the contextual characteristics of terms. All of the previous approaches use static representations that have fundamental limitations addressed by the use of contextualized representations.

Supervised Expansion There is a vein of work using supervised learning to perform pseudo-relevance feedback. Cao et al. (2008) and Imani et al. (2019) use feature-based models to try to predict what terms should be used for expansion. A common practice is to classify terms as positive, negative, or neutral and use classification methods to maximize the number of predicted positive terms. We use this labeling method to intrinsically evaluate the utility of both of our unsupervised and supervised approaches. Moreover, early work in query expansion (Xu et al., 2017) demonstrates that incorporating the context of the expansion term and particularly its proximity to query terms, improves retrieval results. By using the labeled data mentioned earlier, and the BERT contextualized language model that learns a text's representation based on its surrounding context, we train models with and without the context of the candidate expansion term to predict a relevant expansion term. Further, an end-to-end neural PRF model (NPRF) proposed by Li et al. (2018) uses a combination of models to compare document summaries and compute document relevance scores for feedback and achieves limited improvement while only using bag-of-words neural models. Later work combining BERT with an NPRF framework (Zheng et al. 2020) illustrates the importance of an effective first-stage ranking method. Recent work attempts to use Reinforcement Learning (RL) to optimize the selection of terms for query expansion (Nogueira & Cho, 2017; Montazeralghaem et al., 2020). However, despite its potential, the gains over a supervised classification model based on convolutional neural networks remain limited (Nogueira & Cho, 2017). Similar to the Nogueira and Cho (2017), for the supervised task we also frame the expansion problem as a binary classification task.

A complementary (Craswell et al. 2019) vein of work (Nogueira et al. 2019) uses generative approaches to perform *document expansion* by predicting questions to add to document. In contrast, we focus on query expansion approaches. The Brown team at TREC Deep Learning 2019 (Zerveas et al. 2020) train a weakly supervised model on 2.6% of the MS MARCO passages that match more than one query by using the OpenNMT (Klein et al. 2017) with a transformer model to paraphrase an input query into an equivalent query. The final query is obtained by concatenation of the original query

with the top paraphrased queries. Our supervised approach requires no training data. And our supervised model, SQET, formulates the expansion task not as a generation task, but as a discriminative classification model for unigrams.

Neural ranking Contextualized Transformer-based models are now widely used for ranking tasks Akkalyoncu (Yilmaz et al. 2019; Dai & Callan, 2019; Li et al., 2020; MacAvaney et al., 2019; Nogueira & Cho, 2019; Nogueira et al., 2020; Padigela et al., 2019; Qiao et al., 2019; Zhang et al., 2019). Dai and Callan (2019) score individual passages and aggregate them into document level score by taking the first, maximum, or sum of passages scores. MacAvaney et al. (2019) propose incorporating contextualized language models into existing neural ranking architectures by considering each layer of contextualized language models as one channel and integrating the similarity matrices of each layer in the neural ranking architecture. Li et al. (2020) propose overcoming the length limitation of Transformer-based models by aggregating passage *representations* rather than passage relevance scores.

Recent research (Gao et al., 2020; Khattab & Zaharia, 2020; MacAvaney et al., 2020; Zhan et al., 2020; Xiong et al., 2021) uses Transformer models to produce query and document representations that can be used for (relatively) efficient first-stage retrieval. In this context, Gao et al. (2020) find that combining a representation-based model with a lexical matching component improves effectiveness. We focus on representations solely as a contextualized word representation model for the task of unsupervised query expansion. A recent work (Wang et al. 2021) extracts representative feedback embeddings based on the pseudo-relevant set of documents identified using a first-pass dense retrieval and then adds them to the dense query representation. It still relies on IDF weights from unigrams for weighting vectors.

3 Methodology

In this section first, we provide a brief overview of WordPiece representation in Sect. 3.1. Then in Sect. 3.2 we introduce our **unsupervised** model Contextualized Embedding for Query Expansion (CEQE) that utilizes contextualized representations for the task of query expansion. CEQE applies to many widely used contextualized embedding representation models, including BERT and its variants. In Sect. 3.3 we introduce our **supervised** model Supervised Query Expansion with Transformers (SQET) with two variants, SQET and SQET-Context.

3.1 Word and wordpiece representations

In contextualized models, to address the problem of out-of-vocabulary terms, subword representation such as WordPieces (Schuster & Nakajima, 2012) is used. For backward compatibility with existing word-based retrieval systems (as well as comparison with previous methods) we use words as the matching unit. We first aggregate WordPiece tokens into a contextualized vector for words. We compute the average embedding vector of word w by $\overrightarrow{w} \triangleq \frac{1}{|w|} \sum_{p_i \in w} \overrightarrow{p_i}$, where p_i is a WordPiece of word w and $|w|$ is the number of WordPieces in the word w.

3.2 Contextualized embeddings for query expansion (CEQE)

In this section, we describe the core of the CEQE model. It follows in the vein of principled probabilistic language modeling approaches, such as the Relevance Model formulation of pseudo-relevance feedback (Lavrenko & Croft, 2001). In contrast to these approaches that are based on static lexical matching, we formulate relevance based on contextualized vector representations. We build the contextualized feedback model based upon the core Relevance Model (RM) formulation:

$$p(w|\theta_R) \propto \sum_{D \in R} p(w, Q, D) \tag{1}$$

where θ_R and R respectively denote the feedback language model and the set of pseudo-relevant documents, i.e., the top retrieved documents, and w, Q and D represent word, query and document respectively. In the original RM formulation, the joint probability of $p(Q, w, D)$ is broken down as follows:

$$\sum_{D \in R} p(w, Q, D) = \sum_{D \in R} p(w, Q|D)p(D) \tag{2}$$

$$= \sum_{D \in R} p(w|Q, D)p(Q|D)p(D) \tag{3}$$

$$= \sum_{D \in R} p(w|D)p(Q|D)p(D) \tag{4}$$

where Eq. 4 is derived from the simplifying independence assumption between the query Q and term w. This assumption results in a static representation based on simple word counts and ignores the query explicitly (by assuming that the expansion term w is conditionally independent of Q given D). It only incorporates evidence indirectly through $P(Q|D)$. In contrast, the proposed CEQE parameterization doesn't assume the term w is independent of the query Q and explicitly incorporates the query focus based on similarity with contextualized vector representations. More formally:

$$\sum_{D \in R} p(w, Q, D) = \sum_{D \in R} p(w|Q, D)p(Q|D)p(D) \tag{5}$$

With a contextualized model it is no longer possible to simply count document terms – they must be grouped, simplified, or compared against a query representation. We explicitly incorporate contextualized query similarity for each word occurrence. We now break down each of the elements in Eq. 5 in more detail. Following common practice, we assume a uniform probability for $p(D)$. $p(Q|D)$ is the posterior probability of the query given a document from the retrieval model. The retrieval model can be either a Language model with Dirichlet smoothing or even BM25. For BM25 the retrieval scores are mapped to a probability distribution by applying the Softmax function on the document scores. We propose several methods to calculate $p(w|Q, D)$ below.

Centroid Representation In this approach, we create a model of the whole query and then compare it to the contextualized representation of each word mention (occurrence), m_w. In the centroid representation we define $\sigma(Q)$, the aggregation of all WordPieces of the query. Note that a representation of a query also includes special delimiter tokens. For example, in BERT

this would include [CLS] and [SEP] tokens that we find carry contextual importance. We include the [CLS] token in particular because it is often used as a representation of the input with respect to the target task. For the query centroid representation we define σ as the mean of its individual component contextual vectors: we represent query $\sigma(Q)$ by $\overrightarrow{Q} \triangleq \frac{1}{|Q|} \sum_{q_i \in Q} \vec{q}$, where q_i is a WordPiece token and $|Q|$ is the length of the query in WordPiece tokens.

We then define $p(w|Q, D)$ by comparing the similarity of individual word mentions to the query centroid representation based on a similarity function δ (e.g., cosine). If m_w^D is a mention of word w in a document D and M_w^D is the complete set of mentions of w:

$$p(w|Q,D) \triangleq \frac{\sum_{m_w^D \in M_w^D} \delta(\overrightarrow{Q}, \overrightarrow{m_w^D})}{\sum_{m^D \in M_*^D} \delta(\overrightarrow{Q}, \overrightarrow{m^D})} \tag{6}$$

The denominator is a normalization constant that considers all word mentions across the entire document to form a probability. This approach is novel because the contextualized vector m_w^D will be different for every occurrence in D because the context surrounding each mention of word w varies.

Term-based Representation In this section we propose an alternative parameterization for $p(w|Q, D)$. Instead of using the centroid of the query to compute a term's similarity to the entire query, we compute the similarity for each query term separately. If q is a query term and $\vec{q}$ is its corresponding contextualized embedding vector, this can be formulated as:

$$p(w|q,D) \triangleq \frac{\sum_{m_w^D \in M_w^D} \delta(\vec{q}, \overrightarrow{m_w^D})}{\sum_{m^D \in M_*^D} \delta(\vec{q}, \overrightarrow{m^D})} \tag{7}$$

To select a term for expansion for the query overall we perform an extra step of pooling across the similarities of individual words. This step combines the contextualized word vectors. Function f calculates the semantic similarity of word w with the whole query by combining the semantic similarity of it with each query term q. We define $f_{\max}(w, Q, D) = \max_{q \in Q} p(w|q, D)$ and $f_{\text{prod}}(w, Q, D) = \prod_{q \in Q} p(w|q, D)$ as MaxPool and MulPool, respectively. If Z' is a normalization factor that is the sum over the terms in document D, which is less computationally expensive than summing over all vocabulary terms, these can be defined as:

$$p(w|Q,D) \triangleq \frac{f_{\max/\text{prod}}(w, Q, D)}{Z'} \tag{8}$$

The final result of all of these methods is a relevance distribution over terms derived from the contextualized representations in top retrieved documents. The result is an updated query language model that can be used on its own or combined with other representations. In our experiments, we follow the *standard* variant (Abdul-Jaleel et al. 2004) of Relevance Model, RM3 which is designed to maintain information in the original query model as well as the information gained from the behaviour of the returned documents by linearly interpolating the relevance model with the original query model:

$$P'(w|\theta_R) = \lambda P(w|\theta_R) + (1 - \lambda) P(w|Q) \tag{9}$$

where $P(w|Q)$ is the original query language model which without loss of generality we confine our experiments to Query Likelihood (QL).

3.3 Supervised query expansion with transformers

In this section we describe the core of the SQET model. SQET models the problem of query expansion as a **classification task** to classify the expansion term as relevant or non-relevant to the query. Given a query Q and an expansion term w, either with a context or without, a BERT-based encoder computes the relevance score between the query Q and the expansion term w. Note that SQET is a discriminative (classification) model that learns the boundary between the relevant and non-relevant classes, rather than a generative model which learns a distribution of the individual classes. We build the set of candidate expansion terms based on the pseudo-relevance documents retrieved using a traditional IR model.

SQET represents a model that computes the relevancy score between the query Q and the expansion term w without any context for w. Following the same notation as Devlin et al. (2019) we feed the query as sentence A and the expansion term as sentence B: $[[\texttt{CLS}], Q, [\texttt{SEP}], w, [\texttt{SEP}]]$. We feed the final hidden state corresponding to `[CLS]` in the model to a single layer neural network with softmax activation function which outputs the probability that the term w is relevant to the query Q. This produces a query expansion term probability distribution over the vocabulary. Following the standard variant of the Relevance Model, RM3 (Equation 9) we perform a linear interpolation of the SQET expansion query terms with the Query Likelihood score of the original query.

SQET-Context aims to leverage the contextual information of the candidate expansion term in the retrieved pseudo-relevant documents. As mentioned earlier the pool of candidate expansion terms is created from the pseudo-relevant documents' terms. To provide the terms with context we define a fixed window of terms around the candidate expansion term's mention in the pseudo-relevant document with size c. Unlike the model from Dai and Callan (2019), BERT-MaxP, that calculates the relevance score of the documents' passages to the query and re-rank the result based on the calculated score, we calculate the relevancy of each *term* of the pseudo-relevant document to the query in order to improve the first round of retrieval by expanding the query with top relevant terms. Since there could be multiple mentions of a candidate expansion term in the pseudo-relevant document, we define the context of the ith mention of the candidate expansion term w as $\texttt{context}(m_w^i)$.

We form the input of the BERT-based encoder by concatenating the Query Q and the context of the the ith mention of the candidate expansion term $\texttt{context}(m_w^i)$: [`[CLS]`, Q, `[SEP]`, $\texttt{context}(m_w^i)$, `[SEP]`]. Similar to SQET model, by feeding the `[CLS]` final hidden state to a feed forward model, we get the probability of $\texttt{context}(m_w^i)$ being relevant to the query Q. To determine the relevance score of the candidate expansion term w, we apply inference using the following three aggregation functions:

- **Max** represents the probability of the candidate expansion term w by maximum relevancy score between Q and $\texttt{context}(m_w^i)$.
- **Weighted Sum (wSum)** represent the probability of the candidate expansion term w by the weighted sum of the relevancy score between Q and $\texttt{context}(m_w^i)$ derived from BERT. The formulation is as follow:

$$p(w) = \frac{1}{Z} \sum_{m_w^i \in M_w} \text{tf}(w, \texttt{context}(m_w^i)) \times \text{BERT}(Q, \texttt{context}(m_w^i)) \tag{10}$$

where M_w is the set of mentions of the candidate expansion term w, $\text{tf}(w, \texttt{context}(m_w^i))$ is the frequency of term w in $\texttt{context}(m_w^i)$, $\text{BERT}(Q, \texttt{context}(m_w^i))$ is

the relevancy score between $\texttt{context}(m_w^i)$ and Q calculated by a BERT-based ranker, and Z is merely a normalizer allowing for the weights to be turned into a probability distribution.

- **invRank** represents the probability of the candidate expansion term w by aggregating the relevancy score between Q and $\texttt{context}(m_w^i)$ according to the inversed log of rank of the $\texttt{context}(m_w^i)$. The formulation is as follow:

$$p(w) = \frac{1}{Z} \sum_{m_w^i \in M_w} \frac{1}{\log_2(rank(\texttt{context}(m_w^i)) + 1)} \times \text{BERT}(Q, \texttt{context}(m_w^i)) \tag{11}$$

M_w and BERT(Q,$\texttt{context}(m_w^i)$) and Z are defined as mentioned above.

4 Experimental setup

4.1 Datasets

We study the models on multiple standard TREC benchmark datasets: Robust, Deep Learning, and Complex Answer Retrieval (CAR). For SQET we focus on its behavior in the well studied adhoc Robust dataset.

Robust The corpus consists of Tipster disks 4 and 5 containing approximately 528K newswire articles. The evaluation topics are the 250 Robust topics (301-450, 601-700). We use the titles as queries.

TREC Deep Learning The 2019 TREC Deep Learning (TREC19-DL) Track created large labeled datasets for ad-hoc search. We perform the full document ranking task with the goal of testing new expansion methods to improve effectiveness. The evaluation has 43 test queries from Bing, and the corpus consists of 3.2 million web documents. Documents are rated on a four point graded relevance scale. The primary measure is nDCG@10.

TREC CAR TREC Complex Answer Retrieval (CAR) (Dietz et al. 2018) is a dataset curated for the TREC Complex Answer Retrieval track introduced in 2017 to address retrieval for complex topics. In this dataset each topic consists of the hierarchical skeleton of a Wikipedia article and its sections. To be more specific, [Radiocarbon dating/Measurement and results/Errors and reliability] is an example topic constructed from the hierarchical skeleton of the Radiocarbon dating Wikipedia article. Two tasks are defined for TREC CAR dataset: 1) Passage ranking and 2) Entity ranking. More specifically, for each topic heading the goal is to retrieve paragraphs and entities, respectively for each task. The most common approach to formulate a query from a topic is to concatenate the different parts of its hierarchical skeleton. The TREC CAR setup includes two types of judgments, *automatic* and *manual*. The automatic (binary) judgments are derived directly from Wikipedia and the manual judgments are created by NIST assessors. Following standard practice we use the automatic paragraph judgments which are automatically derived from articles. TREC CAR has different relevance annotations based on the section path of the topic. We take advantage of the "Tree Qrels" that consider intermediate paragraphs headings and thus contain more relevance judgments than the older CAR "Hierarchical Qrels". There are 2283 evaluation topics for BenchMarkY1Test for the Tree QRels. We note that this is an updated dataset from the original Y1 "hierarchical" dataset and is more challenging because root whole articles (full pages) are excluded. This experimental protocol follows the Y2 and Y3 task definitions, performed on the Y1 query data because automatic judgments are only available on this set. We use the standard V2 of the paragraph collection

for the unit of retrieval. It consists of approximately 30 million paragraphs from Wikipedia from December 2016. For a survey of approaches on TREC CAR, see Nanni et al. (2017) as well as its overview paper (Dietz et al. 2017) at TREC.

Evaluation Metrics Since we focus on introducing relevant documents to a candidate pool for downstream ranking, we consider both recall-focused metrics (Recall@100, Recall@1000, MAP) as well as precision-based measures (P@10/20, nDCG@10/20). For Robust, in order to compare with previous works we report precision and nDCG at cut-off 20. We report the official primary measure for TREC19-DL, nDCG@10. For significance testing, we use a paired t-test with significance at the 95% confidence interval.

4.2 Intrinsic expansion judgments

Beyond direct retrieval, we also assess term selection quality intrinsically. We directly measure the utility of individual expansion terms. Following previous work from Imani et al., we generate this term utility by performing expansion one word at a time (Imani et al. 2019). Retrieval effectiveness assesses whether a term is good (helps retrieval), bad (hurts retrieval), or neutral (has no effect). We pool the top thousand candidate expansion terms from all candidate expansion methods. These are issued to the retrieval system with the original query (each with a default weight of 0.5, the default relevance model expansion weight). This approach follows standard relevance model interpolation practice defined in Eq. 9, which removes the dependence on the original query length (instead of simply appending a word). We measure improvement based on recall@1000 with a threshold of 0.001. For Robust this results in approximately 500k candidate terms. For the intrinsic evaluation only queries with at least one positive expansion term are used. This is 181 queries for Robust with 10,068 positive terms.

4.3 Baselines

4.3.1 Unsupervised: CEQE

We study the behavior of the CEQE model in comparison with standard models from probabilistic language modeling. For the baseline retrieval we use BM25 because it is the most widely used first-pass unsupervised ranker used to generate candidate pools. We compare with two static expansion models (Kuzi et al. 2016) and a proven pseudo-relevance feedback model, the Relevance Model (Lavrenko and Croft 2001). We use the standard relevance model (RM3 variant) that performs linear interpolation of the RM expansion terms with the original query using the Query Likelihood score.

Static Embeddings For static word embeddings we use GloVe (Pennington et al. 2014) embeddings. The pre-trained 300 dimensional Glove word embeddings are extracted from a 6 billion token collection (Wikipedia dump 2014 plus Gigawords 5). These embeddings are the most effective static embeddings for a variety of tasks, including previous work (Diaz et al. 2016) on query expansion. We use the static embeddings with two variations. The *Static-Embed model* (Kuzi et al. 2016) is a global expansion model using GloVe expansion on the target collection vocabulary. For a fair comparison with CEQE, we additionally consider a *Static-Embed-PRF* variant that has its vocabulary limited to terms appearing in the PRF documents.

4.3.2 Supervised: SQET

Similar to the unsupervised model, CEQE, we study the behavior of the SQET variants in comparison with the standard models from probabilistic language modeling: BM25 and RM3.

BM25$_{invRank}$ To validate the effect of the scores obtained by BERT on the expansion terms' ranking in the SQET-Context, we replace the BERT-based score with the BM25 score and use the inverse log rank aggregation approach to calculate the final score.

MASK-QE We replace a query term with a `[MASK]` token in order to see what terms can be in the position of the masked query term. We take advantage of a pre-trained BERT model to predict the masked query term.

4.4 System details

All collections are indexed with the Galago[3] open-source retrieval system for research. The query models and feedback expansion models are all implemented using the Galago query language. We perform stopword removal and stemming using Galago's stopword list and Krovetz stemmer, respectively.

Contextualized Embedding Model We use BERT because it is the most widely used contextual representation model. We use the pre-trained BERT (BERT-Base, Uncased) model with maximum sequence length of 128. for calculating the contextualized embedding vectors.

- *Unsupervised* Since the documents in Robust are longer than 128 tokens we split the documents into chunks with a maximum size of 128 tokens. For the primary CEQE results in this section we use a single layer of the contextualized representation, the second to last layer (11) of BERT. This layer was shown to be the most effective single layer on NER (Devlin et al. 2019) and it was shown that later layers (before the last) were the most effective word representations for multiple language tasks (Peters et al. 2019) that use contextual embeddings as features. Initial preliminary experiments confirmed this finding.
- *Supervised* In SQET-Context, the window size is chosen from $\{5, 10\}$. If there are not enough terms surrounding the candidate expansion term, we pad the sentence with `[PAD]` wordPiece token. For the MASK-QE baseline, we observe that the predicted terms are sensitive to whether the input text is padded. In order to conduct our experiments with batched input, we set the maximum sequence length of BERT for query input to 12 since the maximum length of tokenized Robust04 queries using WordPiece (Schuster and Nakajima 2012) is equal to 10.

Neural ranking models For our neural models we adopt CEDR (MacAvaney et al. 2019). In particular, to align with the use of the contextualized models we use the BERT variant. For Robust, we use the CEDR-KNRM model trained by the authors (MacAvaney et al. 2019). Throughout the paper we refer to the CEDR-KNRM as CEDR. For TREC19-DL we use a CEDR variant trained on a random sample of 1000 MS MARCO train queries with early stopping to terminate when there is no validation improvement for 20 iterations.

[3] http://www.lemurproject.org/galago.php.

Parameter settings The unsupervised retrieval and feedback hyperparameters are tuned using grid search for Robust and TREC19-DL. The b and $k1$ are tuned for BM25 as well as μ for the QL model in the RM3 score. The range for BM25 parameters, b and $k1$ is [0.1, 1) with the incremental step of 0.05, and [0.1, 4) with the incremental step of 0.1, respectively. The range of values for μ parameter for the QL is between [250, 3000] with the incremental step of 250. For the CAR collection we use the provided train and test splits. We tune the retrieval hyper-parameters on a subset of the BenchmarkY1Train data using grid search. We set $\mu = 400$ for QL and $b = 0.5$, $k1 = 1.2$ for BM25. For Robust, we use five-fold cross-validation with the splits introduced by Huston and Croft (2014). For TREC19-DL the original 2019 track only used MS MARCO for training. We set hyper-parameters using five cross-validation with random splits on the topics. Moreover the average parameters reported for the cross-fold validation experiments for TREC19-DL and Robust are rounded to the nearest integer. For all PRF query expansion methods we tune the number of documents ($\{5, 10, ..., 100\}$ by 5), terms ($\{10, 20, ..., 100\}$ by 10) , and interpolation coefficient ($\{0.1, 0.2, ..., 0.9\}$ by 0.05).

Training—Supervised We fine-tune the SQET variants using a TPU v3 with a batch size of 128. The SQET model includes approximately 350K negative and 6.6K positive instances. The Context-SQET model consists of approximately 2M negative and 4K positive instances. To avoid biasing the model towards predicting non-relevant labels, which are approximately 50 times more frequent in the training set, we build each batch by sampling an equal number of relevant and non-relevant expansion terms. For both models, we use Adam (Kingma and Ba 2014) with the initial learning rate set to 2×10^{-6}, learning rate warmup proportion equal to 0.1 of the training steps, and linear decay of the learning rate.

5 Experimental results

In this section we present our experimental results for our unsupervised and supervised query expansion models. First, in Sect. 5.1 we study how to incorporate contextualized embeddings for the task of unsupervised and supervised query expansion (RQ1). Then, in Sect. 5.2 we explore the effect of variants in combination with neural ranking methods (RQ2). Next, in Sect. 5.3 we study how a reranked neural result can be used as a basis for further expansion and reranking (RQ3). Throughout these questions we compare both supervised and unsupervised contextualized language models.

5.1 Contextualized query expansion

We first evaluate the retrieval effectiveness of our expansion models in combination with unsupervised retrieval systems, such as BM25. We study this setup because expansion is widely performed on top of these simple and fast unsupervised baselines. We start with CEQE and baselines on the 2019 Deep Learning Track in Table 1 and TREC CAR in Table 2. Note that the TREC CAR results for CEQE are a new contribution of this work. After these, we compare the behavior of CEQE and SQET on the Robust collection.

Deep Learning 19 (**Table** 1) We report the official evaluation measures for the TREC 2019 Deep Learning Track (Craswell et al. 2019) as well as Recall@1000. For nDCG@10, the baseline BM25 retrieval is more effective than all expansion methods. To give an indicator of the BM25 + RM3 parameters, the average parameter settings across the folds is: 15 feedback docs, 85 expansion terms, and interpolation weight of 0.4. Similar to Robust,

Table 1 Ranking effectiveness of CEQE on unsupervised baseline retrieval for Deep Learning 2019 Track for the task of full document ranking

Model	P@10	nDCG@10	mAP@1000	Recall@100	Recall@1000
BM25	0.6535	**0.5730**	0.3513	0.4053	0.6950
BM25 + RM3	0.6256	0.5343	0.3975‡	0.4434‡	0.7750‡
Static-Embed	0.6186	0.5427	0.3373	0.3973	0.7179
Static-Embed-PRF	0.5605	0.4925	0.3166	0.3715	0.6737
CEQE-Centroid	0.5580	0.5580	0.4144‡	0.4464‡	0.7804‡
CEQE-MulPool	0.6442	0.5563	0.3724‡	0.4295‡	0.7560‡
CEQE-MaxPool	**0.6581**	0.5614	0.4161†‡	0.4506‡	0.7832‡
CEQE-MaxPool-RM3comb	0.6535	0.5579	**0.4178**†‡	**0.4507**‡	**0.7843**‡
TREC 2019 Median	0.6597	0.5834	0.2984	0.3748	0.5484
TREC 2019 Best	0.8093	0.7260	0.4280	0.4670	0.7553

The superscript † and ‡ denote statistical significance over BM25 + RM3 and Static-Embed, respectively

Bold indicates the best result in each column

Table 2 Ranking effectiveness of CEQE on unsupervised baseline retrieval for the Complex Answer Retrieval (CAR) Track

Model	mAP@1000	R-Prec	Recall@100	Recall@1000
BM25	0.1102	0.0857	0.3680	0.5867
BM25 + RM3	0.1119	0.0881	0.3782	0.6056
Static-Embed	**0.1144**	**0.0895**	0.3796	0.5900
Static-Embed-PRF	0.1135	0.0879	**0.3880** †	0.6014
CEQE-Centroid	0.1124	0.0869	0.3806	0.6138†‡
CEQE-MulPool	0.1020	0.0801	0.3615	0.6018
CEQE-MaxPool	0.1127	0.0877	0.3801	0.6141†‡
CEQE-MaxPool-RM3Comb	0.1122	0.0871	0.3808	**0.6155**†‡

The superscript † and ‡ denote statistical significance over BM25 + RM3 and Static-Embed-PRF, respectively

Bold indicates the best result in each column

we observe that a tuned RM3 outperforms the static embedding methods across all measures. CEQE-MulPool and CEQE-MaxPool also outperform the static embedding model across all measures. The best performing *expansion* method is CEQE-MaxPool, outperforming RM3 (Note that this comparison is among the individual expansion methods excluding CEQE-MaxPool-RM3comb, TREC 2019 Median and TREC 2019 Best runs). The interpolation of CEQE-MaxPool and RM3 yields small improvements over MaxPool, indicating that RM3 is not adding significantly new information. We note that given the small sample size (43 topics), none of the unsupervised methods show statistically significant differences between them. As shown later, that requires performing expansion on top of neural rankings.

Although our experimental setup is based on cross-fold validation (rather than tuning on MARCO), we include the reported values from the Deep Learning track overview (Craswell et al. 2019) for reference. Importantly, we observe that the CEQE-MaxPool

outperforms all submitted TREC systems on recall@1000 and is in the top five for recall@100. Moreover, we observe that the CEQE-MaxPool performs competitively with the TREC 2019 Median run in P@10. It's noteworthy that the unsupervised CEQE-MaxPool 'traditional' model is only slightly lower than the median for P@10 and nDCG@10 with runs that include many state-of-the-art neural models. More specifically, the TREC 2019 Median and TREC 2019 Best are among the runs that take advantage of a train dataset with more than 36K queries and are specifically tuned to improve nDCG metric. However, as stated earlier the CEQE models does not take advantage of any train data and its focus is to improve recall by including relevant documents in top 100 or top 1000 retrieved document to later to be used as a first stage run for the neural re-rankers. Moreover, since CEQE is a query expansion technique it is prone to drift the query by introducing extraneous words (Croft et al. 2010) which can result in drop in the performance in terms of precision-based metrics.

Complex Answer Retrieval (Table 2) We follow previous expansion work on CAR (Dalton et al. 2019), and use BenchmarkY1Tree with the root topic titles removed. This is the recommended setup from the CAR organizers, and is an updated version of the widely used hierarchical judgments (and therefore slightly different from reported hierarchical values (Nogueira et al. 2019)). The baselines are comparable to the Lucene runs provided by the track organizers.

The CAR collection is particularly challenging for feedback models because there are few relevant paragraphs per query in the collection, approximately 3.5 on average. Also, recall for CAR topics is lower by more than 10% for BM25 and 18% for BM25 + RM3 when compared with the other test collections. The PRF feedback parameters learned on BenchmarkY1Train are 20 feedback paragraphs, 50 feedback terms, and an interpolation weight of 0.9. This indicates almost all weight is being given to the original query (which is also longer with multiple Wikipedia headings).

The results show that the CEQE-MaxPool outperforms the existing static methods for Recall@1000. In fact it provides the only statistically significant improvement over the BM25 baseline. The interpolation of CEQE-MaxPool and RM3 yields marginal improvements over MaxPool alone, indicating the CEQE is relatively robust on its own.

We observe small gains over RM3 from the static embedding models. In particular, the static-embed-PRF has the best Recall@100 of the expansion runs. The static Glove embedding has the best MAP score. We hypothesize that requiring the terms to be in both GloVe and PRF documents is providing a useful filter when there are few relevant documents retrieved. CEQE is competitive and insignificantly different in other measures. All in all, the main objective of CEQE is to do query expansion to include more relevant terms to the query in order to include more relevant documents in the first stage ranking. Query expansion can introduce query drift due to extraneous words or weighting of terms (Croft et al. 2010). Thus, they perform better with recall-oriented metrics compared to precision-oriented metrics.

5.1.1 Comparing CEQE and SQET on Robust

We now study the behavior of the CEQE unsupervised model and compare it with the SQET supervised model on Robust in Table 3. All The Static-Embed variants, CEQE variants and SQET variants outperform the baseline BM25 retrieval method across all measures. MASK-QE is the only expansion method that performs worse than the BM25 baseline.

Table 3 Ranking effectiveness on the Robust collection

Model	P@20	nDCG@20	mAP@1000	Recall@100	Recall@1000
BM25	0.3657	0.4193	0.2574	0.4165	0.6933
BM25 + RM3	0.3998	0.4517	0.3069	0.4610‡	0.7588‡
Static-Embed	0.3675	0.4285	0.2615	0.4217	0.7125
Static-Embed-PRF	0.3781	0.4400	0.2703	0.4324	0.7231
CEQE-Centroid	0.3922	0.4462	0.3019‡	0.4593‡	0.7653†‡
CEQE-MulPool	0.3847	0.4360	0.2845‡	0.4517‡	0.7435‡
CEQE-MaxPool	0.4040‡	**0.4587**	0.3086‡	0.4651‡	**0.7689**†‡
CEQE-MaxPool-RM3Comb	**0.4042**	0.4577	**0.3104**‡	**0.4656**‡	0.7636‡
CEQE-MaxPool(fine-tuned)	0.3986‡	0.4528	0.3071‡	0.4647‡	0.7626‡
MASK-QE	0.3655	0.4223	0.2539	0.4144	0.6940
SQET	0.3695	0.4307	0.2606	0.4231	0.6991
SQET-Context$_{invRank}$	0.3777	0.4392	0.2835	0.4448	0.7461
SQET-Context$_{invRank}$-RM3Comb	**0.4018**	**0.4575**	**0.3127**	**0.4710**†	**0.7733**†
SQET-Context$_{invRank}$CEQE-MaxComb	0.4040	0.4611†	0.3140	0.4756†	0.7783†

The superscript † and ‡ denote statistical significance over BM25 + RM3 and Static-Embed-PRF, respectively

Bold indicates the best value in each section of the table

The static embedding models outperform BM25, but do not perform as well as the Relevance Model (RM3). The Static-Embed-PRF method that only uses terms in the PRF documents' vocabulary is more effective across all measures over the Static-Embed approach with a global vocabulary. We hypothesize that this may be due to the fact that the query results provide a topically focused vocabulary and filters out generally similar noise. RM3 significantly outperforms the Static-Embed method for MAP, but not other measures. To give an indicator of the BM25 + RM3 parameters, the average parameter settings across the folds is: 22 feedback docs, 71 expansion terms, and interpolation weight of 0.3. We observe that all CEQE variants outperform the static embedding models. The results show CEQE-MaxPool is the best CEQE variant method. The Centroid method is slightly lower than MaxPool, and both outperform multiplicative pooling. The CEQE-MaxPool result outperforms the BM25+RM3 across all measures and in Recall@1000 is significant over both static embedding methods and BM25+RM3, which demonstrates the utility of context-dependent embeddings.

The CEQE-MaxPool-RM3Comb which is a combination of CEQE-MaxPool and RM3 shows a small insignificant improvement over the CEQE-MaxPool result. CEQE-MaxPool(fine-tuned) shows the result of using MaxPool with 'fine-tuned' contextual embeddings from a BERT model trained for ranking on Robust. The results show small and insignificant differences across all measures. It is almost identical to vanilla embedding effectiveness after being combined with RM3. This indicates that, when used for CEQE-based expansion, pre-trained models are comparable in effectiveness to ones fine-tuned for ranking. Therefore, we did not continue conducting experiments with a fine-tuned model for TREC19-DL and CAR. To our knowledge these are the best unsupervised query expansion results for Robust that do not use external collections.

The SQET supervised method outperforms the baseline BM25, but does not outperform the BM25+RM3 baseline. The SQET-Context$_{invRank}$ outperforms the MASK-QE

Table 4 Ranking effectiveness of neural ranking on top of query expansion methods for Robust

Model	P@20	nDCG@20	mAP@1000	Recall@100	Recall@1000
BM25 + RM3	0.3998	0.4517	0.3069	0.4610	0.7588
BM25 + CEDR (MacAvaney et al. 2019)	0.4713	0.5458	0.3312	0.4983	0.6933
(BM25 + RM3) + CEDR	0.4719	0.5435	0.3500†	0.5192†	0.7570†
(BM25 + CEQE-MaxPool) + CEDR	0.4735	0.5462	0.3532†	0.5258†‡	0.7719†‡
(BM25 + SQET-Context$_{invRank}$) + CEDR	**0.4783**	**0.5487**	0.3475	0.5194	0.7449
(BM25 + SQET-Context$_{invRank}$ RM3Comb) + CEDR	0.4741	0.5437	**0.3543**†	**0.5261**†‡	**0.7722**†‡

The superscript † and ‡ indicate significance over BM25 + CEDR and (BM25 + RM3) + CEDR with re-ranking the top 1000, respectively. Bold indicates the best value in each section of the table

and SQET, but does not outperform the BM25+RM3 baseline on its own. Examining the results, we hypothesize that this is because of the importance in term weighting with multiple expansion terms. The SQET-Context$_{invRank}$ model is only trained to classify the boundary between relevant expansion terms and non-relevant, and the predicted scores are not effective for term weighting in the query language model. We combine the RM3 and SQET-Context$_{invRank}$ using linear interpolation in SQET-Context$_{\text{invRank}}$Comb, tuned for average precision. This demonstrates that combining the signals from unsupervised RM3 model and supervised SQET result in further gains. The resulting model is significantly better than the RM3 expansion in recall@100 and recall@1000 metrics.

Finally, the last row of the table, SQET-Context$_{invRank}$CEQE-MaxComb, shows the result of the linear interpolation of the CEQE-MaxPool and the SQET-Context$_{invRank}$ tuned on mean average precision and that both of models provide gain in the final results. This model outperforms the BM25+RM3 across nDCG@20, recall@100, recall@1000 metrics.

5.2 PRF effect on neural reranking

We now study how PRF methods impact the effectiveness of neural reranking models (RQ2). It is important to have effective expansion in the first pass to retrieve sufficient numbers of documents to rerank. The results of our experiments on Robust for unsupervised CEQE as well as supervised SQET-Context models are shown in Table 4. Applying neural reranked models baselines designed for document ranking, CEDR (MacAvaney et al. 2019), on expanded query runs results in significant gains to average precision, recall@100, and recall@1000 for both RM3, CEQE and SQET-Context. Replacing RM3 with CEQE for expansion results in significant improvement over Recall@100 and Recall@1000. Also, re-ranking the SQET-Context$_{invRank}$ model with CEDR results in highest P@20 and nDCG@20.

5.3 Expansion after reranking

In this section we study how a reranked neural result can be used as a basis for further expansion and reranking (RQ3). This is a critical step because there must be a sufficient number of relevant documents in the top ranks for PRF to be effective. We evaluate

multi-round supervised reranking based on expansion runs for Robust for CEQE-MaxPool model in Table 5. The top of the table shows results from the leading neural ranking and PRF approaches, including Neural PRF (Li et al. 2018), CEDR, and Birch (Yilmaz et al. 2019). The results in this section all perform re-ranking on 1000 results from the baseline. We experimented with reranking 100 results and found it consistently performed worse. The baseline model run is BM25+CEDR followed by RM3 expansion with CEDR reranking, which we denote as *(BM25 + CEDR) + RM3 + CEDR*. The results show it outperforms Birch in nDCG@20 and P@20, as well as its own previous result for P@20 on just BM25. Replacing RM3 with CEQE for the expansion consistently outperforms the previous best CEDR results across all measures and significantly over Recall@1000. The runs compare performing RM3 and CEQE-MaxPool on the CEDR baseline (which reranks an initial BM25 first run). The second pass results are then reranked again using CEDR. The result has further improvement over previous approaches. The same trend continues, with the CEQE-MaxPool outperforming the reranked RM3 run.

A common approach when applying BERT-based neural ranking is to perform learning-to-rank to combine the BERT and retrieval score. A simple proven approach is linear interpolation of the underlying retrieval score with neural ranking model (Yilmaz et al., 2019; Yang et al., 2019). We apply this to the two best runs, learning the interpolation using the previously described cross-validation setup. The results demonstrate that linear interpolation with these expansion runs continues to show gains. The interpolation with CEQE-MaxPool is the best performing, and compared with the previous Birch shows over 5% relative gain P@20 and nDCG@20 as well as improving MAP. These results show that multiple rounds of expansion and reranking can continue to result in significant improvements.

5.4 Intrinsic expansion evaluation

In this section we examine the effectiveness of the expansion approaches to rank positive expansion terms that improve Mean Average Precision (at 1000) when added to the query. This experiment evaluates a method's ability to identify good expansion terms in isolation. The results are shown in Table 6 for the key expansion models to compare for Robust. Since a fixed top-k expansion terms are usually selected for expansion we evaluate the intrinsic evaluation with set-based precision numbers at common thresholds for the number of expansion terms. The results show that a well-tuned Relevance Model significantly outperforms query expansion models based on static embeddings. In contrast, we find that CEQE provides improvements in early ranks for P@10 and P@20. All the CEQE models significantly improve over static embedding models across all metrics. And further, we find that CEQE-MaxPool significantly outperforms the Relevance Model expansion effectiveness for P@10 and P@20. It is insignificantly different from the Relevance Model at rank 100. This indicates that the strength of CEQE is selecting a higher number of "good" terms earlier, allowing improved effectiveness with fewer expansion terms.

The SQET-Context$_{invRank}$ is the best performing model in the early ranks among all models, but is slightly outperformed by SQET-Context$_{wSum}$ at rank 100. The SQET model is significantly outperformed by the Relevance Model baseline, SQET-Context$_{wSum}$ and SQET-Context$_{invRank}$ across all measures. This indicates the power of BERT when it is provided with the context and term relations. Moreover, SQET-Context$_{Max}$ is also outperformed by Relevance Model, SQET-Context$_{wSum}$ and SQET-Context$_{invRank}$. This shows that the different context around the candidate term across the corpus provides

Table 5 Ranking effectiveness of multi-round neural re-ranking and expansion for Robust

Model	P@20	nDCG@20	mAP@1000	Recall@100	Recall@1000
Neural PRF-DRMM (Li et al. 2018)	0.4064	0.4576	0.2904	–	–
BM25 + CEDR (MacAvaney et al. 2019)	0.4713	0.5458	0.3312	0.4983	0.6933
Birch (Yilmaz et al. 2019)	0.4657	0.5325	0.3697	–	–
(BM25 + CEDR) + RM3	0.4458	0.5211	0.3321	0.4881	0.7751†
(BM25 + CEDR) + RM3 + CEDR	0.4783	0.5499	0.3574†	0.5291†	0.7751†
(BM25 + CEDR) + RM3 + CEDR Interp	0.4837†	0.5565	0.3739†	0.5440†	0.7751†
(BM25 + CEDR) + CEQE-MaxPool	0.4504	0.5250	0.3366	0.4931	0.7874†‡
(BM25 + CEDR) + CEQE-MaxPool + CEDR	0.4799	0.5516	0.3601†	0.5332†	0.7874†‡
(BM25 + CEDR) + CEQE-MaxPool + CEDR Interp	**0.4904**†	**0.5621**†	**0.3773**†	**0.5486**†	**0.7874**†‡

The superscript † and ‡ indicate significance over BM25 + CEDR and (BM25 + CEDR) + RM3 baselines, respectively

Bold indicates the best result in each column

Table 6 Intrinsic ranking evaluation of positive expansion terms on Robust

Model	P@10	P@20	P@100
Relevance Model	0.1693	0.1419	0.0871
Static-Embed	0.1008	0.0780	0.0511
Static-Embed-PRF	0.1357	0.1083	0.0655
CEQE-MulPool	0.1349	0.1174	0.0737
CEQE-Centroid	0.1751	0.1481	0.0826
CEQE-MaxPool	0.1830†	0.1500†	0.0841
MASK-QE	0.0544	0.0515	0.0414
SQET	0.1207	0.1104	0.0758
SQET-Context$_{Max}$	0.1332	0.1085	0.0695
SQET-Context$_{wSum}$	0.1763	0.1556†	0.0921
SQET-Context$_{invRank}$	**0.1942**†	0.1560†	0.0900
BM25$_{invRank}$	0.1610	0.1336	0.0802
RRF(CEQE-MaxPool, SQET-Context$_{invRank}$)	0.1938†	**0.1583**†	**0.0976**†

The superscript † denotes the statistical significance over the Relevance Model

Bold indicates the best result in each column

valuable information for the ranking of the term. Also, SQET-Context$_{invRank}$ outperforms the BM25$_{invRank}$ highlighting the effect of BERT scores in the ranking. The poor performance of the MASK-QE demonstrates that since the pre-trained model is ranking all the terms in its vocabulary, it is a noisy model and cannot generate a good ranking of expansion terms for the target corpus. We investigate the effect of combing the knowledge coming from our unsupervised model, CEQE-MaxPool and our supervised model, SQET-Context$_{invRank}$ by calculating the Reciprocal Rank Fusion (RRF) of their expansion terms ranking. The RRF(CEQE-MaxPool, SQET-Context$_{invRank}$) significantly outperforms the Relevance Model across all measure. This shows that both of the two methods provide valuable and different signals in ranking expansion terms. Also, the RRF improves upon both CEQE-MaxPool and SQET-Context$_{invRank}$ across the P@20 and P@100.

6 Qualitative behavior analysis

6.1 Query-by-query analysis

To better understand the ranking behaviour of our proposed model, we compare the top ranked expansion terms of RM1, CEQE-MaxPool and SQET-Context$_{invRank}$ in Table 7. We illustrate the performance of our approach using [Topic 405, cosmic event] and [Topic 685, oscar winner selection] which performed well in the extrinsic evaluation (more than 10% improvement of mAP when comparing CEQE-Max and BM25+RM3). The first row has the terms (unstemmed) with the greatest improvement for the query.

We observe that the CEQE model identifies mostly all of the positive terms from RM as well as introducing additional relevant terms for both topic 405 and 685. More generally, we see that the CEQE terms appear to have a stronger semantic relationship with the query terms. The RM terms appear most loosely related and have additional noise terms, including

Table 7 Example query expansion terms for Topic [405 , cosmic events] and [685, oscar winner selection] in Robust collection

Topic 405	Cosmic events
Positive terms:	Astronomers, astronomical, bang, big, galaxies, light, matter, particle, particles, Physicist, scientists, space, theory, universe, years
RM:	Energy, **space**, solar, **particle**, **earth**, radiation, proton, article, ray, large, **Universe**, type, fluence, magnitude, **particles**
CEQE-MaxPool:	**Space**, **universe**, radiation, energy, earth, solar, **particles**, **big** , **years** , **Matter**, dust, article, ray , **bang**, galactic , **scientists**
SQET − Context:	**Universe**, **years**, **astronomers**, radiation, **scientists**, **bang**, **matter**, **Galaxies**, dust, energy, **physicist**, time, **big**, research, **theory**, **astronomical**
Topic 685	Oscar winner selection
Positive terms:	Academy, academys, nominations, nomination, critics, members, award, Awards, branch, ignored, true, films, film, directors, director, filmmaker
RM:	Best, **film**, picture, million, **academy**, years, **award**, home, Edition, **films**, man, four, 1, 5
CEQE-Maxpool:	**Film**, **academy**, picture, winners, **award**, **films**, million, oscars, Box, presented, **awards**, **director**, years, **nominations**
SQET-Context:	**Awards**, oscars, **nominations**, nominees, years, edition, **award**, nominated Winners, home, **films**, dga, winning, **film**, **academy**, ua, **nomination**

This includes the important intrinsic positive labels, Relevance Model, CEQE-MAxPool and SQET-Context$_{invRank}$ expansion terms. Terms with positive intrinsic labels are bolded

Table 8 Win/Loss comparison to BM25 on Robust

Model	Win	Neutral	Loss
BM25	–	–	–
BM25 + RM3	151	26	73
CEQE-MaxPool	154	23	73
SQET-Context$_{invRank}$	149	32	69

general terms like ‘article‘, ‘large‘ and ‘type‘ for topic 405. We hypothesis, this is because RM focuses on terms that co-occur across multiple PRF documents, but it does not explicitly model the relationship to the query. In contrast CEQE explicitly focuses on the query. As a result, the CEQE model produces fewer terms that co-occur by chance. Further, for topic 405 SQET-Context$_{invRank}$ ranks more positive expansion terms in higher ranks in comparison with RM1. Also, the SQET-Context$_{invRank}$ rank two terms ‘astronomers‘ and ‘astronomical‘ in the top ranks that both RM and CEQE-MaxPool have missed. Moreover for topic 685, SQET-Context$_{invRank}$ is able to exclude the digits, while RM is ranking them in top expansion terms.

Further, Table 8 shows the win/loss comparison to BM25 for three expansion methods: BM25+RM3, CEQE-MaxPool and SQET-Context$_{invRank}$. The CEQE-MaxPool has the highest wins across three methods. However, CEQE-MaxPool and BM25+RM3 have similar behavior with losses. The SQET-Context$_{invRank}$ model alleviates the losses by using supervision, but it is more conservative and has the highest neutrals.

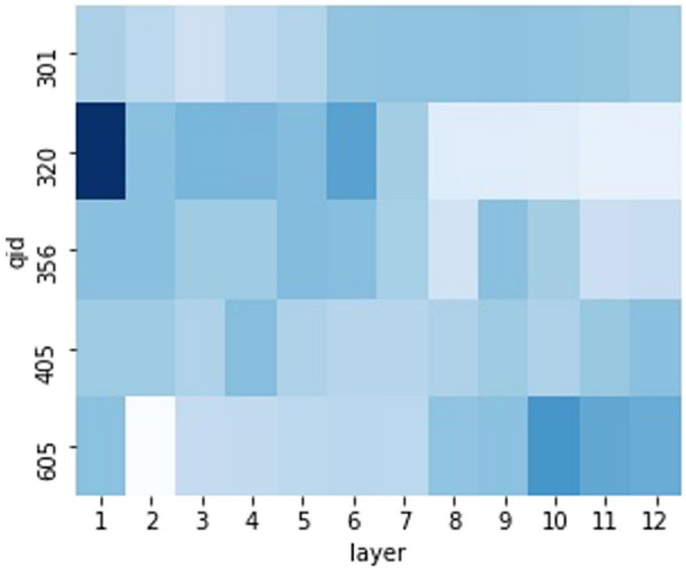

Fig. 1 Heatmap of mAP value for 5 different queries across 12 different layers. Dark blue indicates the best result and lighter color indicates the worst result. Layer 12 represents the last layer of the BERT model

6.2 Tensor representation analysis

In this section, we discuss initial results for a tensor representation using multiple representations for terms and queries. We perform initial experiments using maximum/average pooling-based aggregation methods to select terms based on their similarity scores across layers. These naive pooling methods do not show significant differences over using the best layer.

We need better methods for selecting the correct layer for the particular query. Figure 1 shows the heatmap of mAP value for 5 different queries across 12 different layers. We can see the highest mAP value is scattered across different layers for different queries. We performed an oracle test and found that if we were able to select the best performing layer for each query (only using the best layer), we would increase MAP by over 7% on Robust. Combining the evidence from multiple layers could perhaps increase this further.

6.3 Computational cost analysis

We use a BERT-based model to produce contextualized embeddings for query expansion, which incurs similar computational costs as BERT-based reranking methods like CEDR (MacAvaney et al. 2019). Generating these embeddings is the most computationally-intensive step of all such methods. Compared to other query expansion approaches, our work's computational costs are most similar to BERT-QE's (Zheng et al. 2020). Both approaches consist of a query expansion step that requires processing with a BERT model followed by an (optional) reranking step that again processes the top document

with a BERT model. The core focus of this work is on effectiveness, although efficiency is an important area for future research.

7 Conclusion

In this work, we study the task of query expansion in both unsupervised and supervised settings by leveraging recent advances in contextualized language models. Query expansion remains a fundamental retrieval task that is needed because it generates higher recall candidate pools to use as a base for reranking with more expensive neural methods. We introduce a new unsupervised method, CEQE, for query expansion that extends relevance feedback approaches to recent advances in contextualized language models. CEQE addresses fundamental challenges using context-dependent term representations for unsupervised pseudo-relevance feedback. We study its empirical effectiveness on multiple standard test collections and the results demonstrate that they are statistically significantly superior to previous static embedding approaches on all three collections for Recall@1000 in addition to Recall@100 for Robust and TREC19-DL. In addition to the recall-based metrics, the best CEQE approach statistically significantly outperforms the static-embedding based approaches in precision-based metrics (P@20 and mAP@100) for Robust collection and (mAP@100) for TREC19-DL. Further, it improves over the other metrics (nDCG@20) for Robust as well as (P@10, nDCG@10) for TREC19-DL compared to the static embedding approaches, however it's not statistically significant. Moreover, CEQE demonstrates comparable performance in terms of Recall@100, and precision-based metrics (mAP@100, R-Prec) for the TREC CAR dataset compared to static embedding approaches.

Besides, CEQE approaches statistically significantly outperform the previous word-based expansion method, the Relevance Model on Recall@1000 for Robust and TREC CAR datasets as well as improves it for TREC19-DL. For other metrics on all three collections, CEQE approaches show comparable , and sometimes more effective performance. In general CEQE approaches are most effective in recall-based metrics compared to precision-based metrics.

We also perform experiments showing how unsupervised expansion methods can be used in combination with neural re-ranking methods. We test performing expansion before neural ranking as well as after it. We find the performing expansion after a first pass of high-precision neural reranking provides additional gains in recall that further benefit another pass of reranking. It opens the possibility that additional multiple rounds of expansion may improve effectiveness further. Moreover, we compare the behavior of contextual expansion models that are supervised, that classify expansion terms as relevant or non-relevant to the query in a PRF context. We explore different approaches to adapt BERT for this task with our proposed SQET method and results show that BERT is powerful when it is augmented with the context of the candidate expansion term.

There are still many areas for future work to explore leveraging the contextualized language models in the task of query expansion in both unsupervised and supervised settings. In particular, how to effectively use multiple representations (layers) generated by contextualized embedding models. Our initial explorations found that combinations of multiple representations can be used in place of a single one. Furthermore, there are many BERT variants and other Transformer-based models that show promise as ways to improve effectiveness. Designing and fine-tuning a multi-task contextual based model to learn the

relevant expansion terms and also its correct weighting for updating the query language model for the final task of ranking is also an area for future work.

Acknowledgments This work was supported in part by the Center for Intelligent Information Retrieval and in part by funded by the EPSRC Fellowship titled "Neural Conversational Information Seeking Assistant", grant reference number EP/V025708/1. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

References

Abdul-Jaleel, N., Allan, J., Croft, W. B., Diaz, F., Larkey, L., Li, X., Smucker, M. D. & Wade, C. (2004) Umass at trec 2004: Novelty and hard. Computer Science Department Faculty Publication Series, 189.

Akkalyoncu Yilmaz, Z., Yang, W., Zhang, H., & Lin, J. (2019, November). Cross-domain modeling of sentence-level evidence for document retrieval. In Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (emnlp-ijcnlp). Hong Kong, China: Association for Computational Linguistics.

Cao, G., Nie, J. Y., Gao, J., & Robertson, S. (2008). Selecting good expansion terms for pseudo-relevance feedback. In Proceedings of the 31st annual international ACM SIGIR conference on research and development in information retrieval. New York, NY, USA: ACM.

Craswell, N., Mitra, B., Yilmaz, E., & Campos, D. (2019). Overview of the trec 2019 deep learning track. In Proceedings of the twenty-eight text retrieval conference, TREC 2019, gaithersburg, maryland, usa, november 13–15, 2019.

Croft, W. B., Metzler, D., & Strohman, T. (2010). Search engines: Information retrieval in practice (Vol. 520). Addison-Wesley Reading.

Dai, Z., & Callan, J. (2019). Deeper text understanding for ir with contextual neural language modeling. In Proceedings of the 42nd international acm sigir conference on research and development in information retrieval. New York, NY, USA: Association for Computing Machinery.

Dalton, J., Naseri, S., Dietz, L., & Allan, J. (2019). Local and global query expansion for hierarchical complex topics. In European conference on information retrieval (pp. 290–303).

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understand- ing. In Proceedings of the 2019 conference of the north American chapter of the association for computational linguistics: Human language tech- nologies, vol. 1 (long and short papers). Minneapolis, Minnesota: Association for Computational Linguistics.

Diaz, F., Mitra, B., & Craswell, N. (2016). Query expansion with locally-trained word embeddings. In Proceedings of the 54th annual meeting of the association for computational linguistics (volume 1: Long papers).

Dietz, L., Gamari, B., & Dalton, J. (2018). Trec car 2.1: A dataset for complex answer retrieval.

Dietz, L., Verma, M., Radlinski, F., & Craswell, N. (2017). Trec complex answer retrieval overview. In Proceedings of the twenty-sixth text retrieval conference, TREC 2017, gaithersburg, maryland, usa, november 15–17, 2017. Retrieved from https://trec.nist.gov/pubs/trec26/papers/ Overview-CAR.pdf

Gao, L., Dai, Z., Chen, T., Fan, Z., Durme, B. V., & Callan, J. (2021, March). *Complement lexical retrieval model with semantic residual embeddings*. In European Conference on Information Retrieval (pp. 146–160). Springer, Cham.

Huston, S., & Croft, W. B. (2014). *Parameters learned in the comparison of re-trieval models using term dependencies*. Ir: University of Massachusetts.

Imani, A., Vakili, A., Montazer, A., & Shakery, A. (2019). Deep neural net- works for query expansion using word embeddings. In European confer- ence on information retrieval (pp. 203–210).

Khattab, O., & Zaharia, M. (2020). ColBERT: Effcient and effective passage search via contextualized late interaction over BERT. In Proceedings of the 43rd annual international acm sigir conference on research and development in information retrieval (sigir 2020).

Kingma, D. P., & Ba, J. L. (2015). ADAM: A method for stochastic optimization 3rd International Conference on Learning Representations, {ICLR} 2015, San Diego, CA, USA, May 7–9, 2015, Conference Track Proceedings. In Conference Track Proceedings.

Klein, G., Kim, Y., Deng, Y., Senellart, J., & Rush, A. M. (2017, July). OpenNMT: Open-Source Toolkit for Neural Machine Translation. In Proceedings of ACL 2017, System Demonstrations (pp. 67–72).

Kuzi, S., Shtok, A., & Kurland, O. (2016). Query expansion using word embeddings. In Proceedings of the 25th acm international on conference on information and knowledge management.

Lavrenko, V., & Croft, W. B. (2001). Relevance based language models. In Proceedings of the 24th annual international acm sigir conference on research and development in information retrieval. New York, NY, USA: ACM.

Li, C., Sun, Y., He, B., Wang, L., Hui, K., Yates, A., Sun, L. & Xu, J. (2018) Nprf: A neural pseudo relevance feedback framework for ad-hoc information retrieval. Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing.

Li, C., Yates, A., MacAvaney, S., He, B., & Sun, Y. (2020). Parade: Passage representation aggregation for document reranking. arXiv preprint arXiv:2008.09093.

Lv, Y., & Zhai, C. (2009). A comparative study of methods for estimating query language models with pseudo feedback. In Proceedings of the 18th acm conference on information and knowledge management.

MacAvaney, S., Nardini, F. M., Perego, R., Tonellotto, N., Goharian, N., & Frieder, O. (2020, July). Expansion via prediction of importance with contextualization. In Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval (pp. 1573–1576).

MacAvaney, S., Yates, A., Cohan, A., & Goharian, N. (2019). CEDR: contex-tualized embeddings for document ranking. In Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval, SIGIR 2019, paris, France, July 21–25, 2019.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems.

Montazeralghaem, A., Zamani, H., & Allan, J. (2020). A reinforcement learning framework for relevance feedback. In Proceedings of the 43rd international acm sigir conference on research and development in information retrieval (pp. 59–68).

Nanni, F., Mitra, B., Magnusson, M., & Dietz, L. (2017). Benchmark for complex answer retrieval. In Proceedings of the ACM SIGIR international conference on theory of information retrieval (pp. 293–296). New York, NY, USA: ACM.

Naseri, S., Dalton, J., Yates, A., & Allan, J. (2021, March). CEQE: Contextualized embeddings for query expansion. In European Conference on Information Retrieval (pp. 467–482). Springer, Cham.

Naseri, S., Foley, J., Allan, J., & O'Connor, B. T. (2018). Exploring Summary-Expanded Entity Embeddings for Entity Retrieval. In CIKM Workshops.

Nogueira, R., & Cho, K. (2017, September). Task-Oriented Query Reformulation with Reinforcement Learning. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (pp. 574–583).

Nogueira, R., & Cho, K. (2019). Passage re-ranking with bert. arXiv preprint arXiv:1901.04085 .

Nogueira, R., Jiang, Z., Pradeep, R., & Lin, J. (2020, November). Document Ranking with a Pretrained Sequence-to-Sequence Model. In Findings of the Association for Computational Linguistics: EMNLP 2020 (pp. 708–718).

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

Padaki, R., Dai, Z., & Callan, J. (2020, April). Rethinking query expansion for BERT reranking. In European conference on information retrieval (pp. 297–304). Springer, Cham.

Padigela, H., Zamani, H., & Croft, W. B. (2019). Investigating the successes and failures of bert for passage re-ranking. arXiv preprint arXiv:1905.01758.

Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (emnlp).

Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. In Proceedings of the 2018 conference of the North American chapter of the association for computational linguistics: Human language technologies, volume 1 (long Paper). New Orleans, Louisiana: Association for Computational Linguistics.

Peters, M. E., Ruder, S., & Smith, N. A. (2019). To tune or not to tune? Adapting pretrained representations to diverse tasks. In Repl4nlp@acl.

Qiao, Y., Xiong, C., Liu, Z., & Liu, Z. (2019). Understanding the behaviors of bert in ranking. arXiv preprint arXiv:1904.07531.

Rocchio, J. J. (1971). Relevance feedback in information retrieval. In G. Salton (Ed.), *The smart retrieval system: Experiments in automatic document processing* (pp. 313–323). Englewood Cliffs NJ: Prentice-Hall.

Roy, D., Paul, D., Mitra, M., & Garain, U. (2016, July 21). Using word embeddings for automatic query expansion.

Schuster, M., & Nakajima, K. (2012). Japanese and korean voice search. 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).

Wang, X., Macdonald, C., Tonellotto, N., & Ounis, I. (2021, July). Pseudo-relevance feedback for multiple representation dense retrieval. In Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval (pp. 297–306).
Xiong, L., Xiong, C., Li, Y., Tang, K.F., Liu, J., Bennett, P., Ahmed, J. and Overwijk, A., (2021). Approximate nearest neighbor negative contrastive learning for dense text retrieval. In International conference on learning representations.
Xu, J., & Croft, W. B. (2017, August). Quary expansion using local and global document analysis. In Acm sigir forum (Vol. 51, No. 2, pp. 168–175). New York, NY, USA: ACM.
Yang, W., Xie, Y., Lin, A., Li, X., Tan, L., Xiong, K., Li, M. and Lin, J. (2019). End-to-end open-domain question answering with bertserini. In Proceedings of the 2019 conference of the north american chapter of the association for computational linguistics (demonstrations).
Yilmaz, Z. A., Wang, S., Yang, W., Zhang, H., & Lin, J. (2019). Applying bert to document retrieval with birch. In Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th inter- national joint conference on natural language processing (emnlp-ijcnlp): System demonstrations (pp. 19–24).
Zamani, H., & Croft, W. B. (2016). Embedding-based query language models. In Proceedings of the 2016 acm international conference on the theory of information retrieval.
Zamani, H., & Croft, W. B. (2017). Relevance-based word embedding. In Proceedings of the 40th international acm sigir conference on research and development in information retrieval (pp. 505–514).
Zerveas, G., Zhang, R., Kim, L., & Eickhoff, C. (2020). Brown university at trec deep learning 2019. arXiv preprint arXiv:2009.04016.
Zhai, C., & Lafferty, J. (2001). Model-based feedback in the language modeling approach to information retrieval. In Proceedings of the tenth interna-tional conference on information and knowledge management. New York, NY, USA: ACM.
Zhan, J., Mao, J., Liu, Y., Zhang, M., & Ma, S. (2020). Repbert: Con-textualized text embeddings for first-stage retrieval. arXiv preprint arXiv:2006.15498.
Zhang, H., Song, X., Xiong, C., Rosset, C., Bennett, P. N., Craswell, N., & Tiwary, S. (2019). Generic intent representation in web search. In Pro-ceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval, SIGIR 2019, paris, france, July 21–25, 2019.
Zheng, Z., Hui, K., He, B., Han, X., Sun, L., & Yates, A. (2020). Bert-qe: Contextualized query expansion for document re-ranking. In Proceed-ings of the 2020 conference on empirical methods in natural language processing: Findings (pp. 4718–4728).

Authors and Affiliations

Shahrzad Naseri[1] · Jeffrey Dalton[2] · Andrew Yates[3] · James Allan[1]

✉ Shahrzad Naseri
shnaseri@cs.umass.edu

Jeffrey Dalton
jeff.dalton@glasgow.ac.uk

Andrew Yates
a.c.yates@uva.nl

James Allan
allan@cs.umass.edu

[1] University of Massachusetts Amherst, Amherst, USA

[2] University of Glasgow, Glasgow, Scotland

[3] University of Amsterdam, Amsterdam, Netherlands