SE-PQA : Personalized Community Question Answering

Pranav Kasela
University of Milano-Bicocca
Milan, Italy
p.kasela@campus.unimib.it

Marco Braga
University of Milano-Bicocca
Milan, Italy
m.braga@campus.unimib.it

Gabriella Pasi
University of Milano-Bicocca
Milan, Italy
gabriella.pasi@unimib.it

Raffaele Perego
ISTI-CNR, Pisa, Italy
Pisa, Italy
raffaele.perego@isti.cnr.it

ABSTRACT

Personalization in Information Retrieval is a topic studied for a long time. Nevertheless, there is still a lack of high-quality, real-world datasets to conduct large-scale experiments and evaluate models for personalized search. This paper contributes to filling this gap by introducing SE-PQA (StackExchange - Personalized Question Answering), a new curated resource to design and evaluate personalized models related to the task of community Question Answering (cQA). The contributed dataset includes more than 1 million queries and 2 million answers, annotated with a rich set of features modeling the social interactions among the users of a popular cQA platform. We describe the characteristics of SE-PQA and detail the features associated with questions and answers. We also provide reproducible baseline methods for the cQA task based on the resource, including deep learning models and personalization approaches. The results of the preliminary experiments conducted show the appropriateness of SE-PQA to train effective cQA models; they also show that personalization remarkably improves the effectiveness of all the methods tested. Furthermore, we show the benefits in terms of robustness and generalization of combining data from multiple communities for personalization purposes.

CCS CONCEPTS

• **Information systems** → **Personalization**; • **Computing methodologies** → **Language resources**.

KEYWORDS

Question Answering; User Model; Personalization

ACM Reference Format:
Pranav Kasela, Marco Braga, Gabriella Pasi, and Raffaele Perego. 2024. SE-PQA : Personalized Community Question Answering. In *Companion Proceedings of the ACM Web Conference 2024 (WWW '24 Companion), May 13–17, 2024, Singapore, Singapore.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3589335.3651445

WWW '24 Companion, May 13–17, 2024, Singapore, Singapore

ACM ISBN 979-8-4007-0172-6/24/05.
https://doi.org/10.1145/3589335.3651445

1 INTRODUCTION

Personalization in Information Retrieval (IR) is a problem studied by the research community for a long time [5–7]. Personalized search aims to tailor the search outcome to a specific user (or group of users) based on the knowledge of her/his interests and online behavior. Given the ability of Deep Neural Network (DNN) models to face many different tasks by extracting relevant features from both texts and structured sources [10], there is the expectation of a huge potential also for their application in personalized IR and Recommender Systems (RS). However, the lack of publicly-available, large-scale datasets that include user-related information is one of the biggest obstacles to the training and evaluation of DNN-based personalized models. Some real-world datasets are commonly used in the literature to design and assess personalization models. These datasets include the AOL query log [12], the Yandex query log, and the CIKM Cup 2016 dataset. Moreover, even synthetically enriched datasets have been used such as: PERSON [14], the Amazon product search dataset [1], and a dataset based on the Microsoft Academic Knowledge Graph [4]. However, all of them have some issues. For example, ethical and privacy issues are related to using the AOL query log [2]. In contrast, the anonymization performed on the Yandex query log prevents its use for training or fine-tuning natural language models. This paper aims to fill this gap by contributing SE-PQA (StackExchange - Personalized Question Answering), a large dataset rich in user-level features that can be exploited for training and evaluating personalized models addressing the *community Question Answering* (cQA) task. SE-PQA is based on StackExchange, a popular cQA platform with a network of 178 open forums. A dump of the StackExchange user-contributed content is publicly available according to a cc-by-sa 4.0 license. With great care, we have preprocessed the original dump by building SE-PQA , a curated dataset with about 1 million questions and two million associated answers annotated with a rich set of features modeling the social interactions of the user community. The features include, for example, the positive or negative votes received by a question or an answer, the number of views, the number of users that selected a given question as a favorite one, the tags from a controlled folksonomy describing the topic dealt with, the comments that other users might have written under a question or an answer. To favor the design and evaluation of personalized models, the users in SE-PQA are associated with their past questions and answers, their social autobiography, their reputation score, and the number of views received by their profile. The cQA task can be addressed on SE-PQA with different methodologies

exploiting either the textual description of questions and answers, the folksonomy, the features modeling the social interactions, or a combination of the above information sources. In this paper, we focus on IR approaches to cQA. Thus, we adapt the cQA task to an ad-hoc retrieval task where the question is seen as a query, and the answers are retrieved from the pool of past answers indexed for the purpose. In this particular setting, the system aims to retrieve a (small) ranked set of documents that contain the correct answers to the user's question. There can be multiple correct answers given to a question, so, in this case, personalization can be used to understand the user's context and background and rank higher the answers that are more relevant to the specific user.

2 THE SE-PQA DATASET

The textual posts in StackExchange forums are associated with rich social metadata information. When users ask a question to the community, they assign some tags specifying the topic to make the question searchable and visible to the users interested in it. The questions are up-voted or down-voted by the community based on their interest and adherence to the community guidelines. In many cases, the community suggests to the question author how to improve the question if it is poorly expressed or formatted. Similar treatment is given to the answers, which can be up-voted or down-voted by the community; moreover, the user who asked the question can also choose the answer he/she deems the best, which may differ from the one that received the most up-votes from others. We note, however, that 87.6% questions and answers are assigned a score given by the difference between the number of up and down-votes. A positive score thus indicates that the post has more up-votes than down-votes, while a negative score indicates that more users down-voted it. StackExchange is quite well known in the IR community: for example, it has been used to train a language model for sentence similarity [9]. To the best of our knowledge, the usage of StackExchange for Q&A tasks has been, however, limited to selecting similar sentence training pairs without exploiting user-level/social features for personalized information retrieval tasks.

With SE-PQA we overcome the previous limitations and provide a complete, curated dataset of textual questions and answers belonging to different, heterogeneous forums. In SE-PQA a user can belong to multiple communities; if we take into consideration users that wrote at least 2 questions, about 50% of them have asked questions in multiple communities. As we increase the minimum number of questions, also the percentage of users using multiple communities increases. For instance, if we take only the users that wrote at least five documents (either questions or answers) and consider both the questions and the answers written by the user, we note that of the resulting 62k users, only 23k (37%) wrote either a question or an answer in only a single community, while 40k (63%) wrote documents in at least two different communities, 26k (42%) in at least three and 18k (28%) in more than three communities.

We claim that personalization is particularly useful for multi-domain collections, where we can exploit information about users' interests in multiple topics of different domains; when the data is instead derived from a single domain or a specific topic of a domain, personalization may become less important. We provide evidence of this assertion in Section 3 where we report about our experiments applying the same personalization approach to both the complete SE-PQA dataset and to the data sampled from separate communities: the results show that personalization on the multi-domain dataset yields better improvements than on only single communities.
To increase diversity, in SE-PQA we thus combine data from multiple networks that can be categorized under the large umbrella of humanistic communities. These communities focus on different topics, but the language used is not too diverse among them. In particular, we choose the following 50 communities:

writers, workplace, woodworking, vegetarianism, travel, sustainability, sports, sound, skeptics, scifi, rpg, politics, philosophy, pets, parenting, outdoors, opensource, musicfans, music, movies, money, martialarts, literature, linguistics, lifehacks, law, judaism, islam, interpersonal, hsm, history, hinduism, hermeneutics, health, genealogy, gardening, gaming, freelancing, fitness, expatriates, english, diy, cooking, christianity, buddhism, boardgames, bicycles, apple, anime, academia.

The training, validation, and test splits are done temporally to avoid any kind of data leakage. The training set includes all questions written from *2008-09-10* to *2019-12-31* (included), the validation set is formed by questions asked between *2019-12-31* and *2020-12-31* (included), while the test set contains the questions from *2019-12-31* till *2022-09-25* (included). There are a total of 1, 125, 407 questions in the dataset, 1, 001, 706 of which have at least one answer (89% of all questions) and 525, 030 of which have a response that the questioner has selected as the best one (47% of all questions). We are left with 822 974 training questions, 78 854 validation questions, and 99 878 test questions after the temporal splits. There are 2, 173, 139 answers and 588, 688 users. Many users in the communities register themselves just for asking a question and then never use their accounts again. In fact, the dataset has a median of 1 user-generated document (either a question or an answer), with about 80% of users having no more than 2 documents. The text in the dataset is preprocessed by removing HTML tags present in the original documents. The dataset is available at Zenodo[1].

Even though SE-PQA can be used for many IR tasks (e.g., duplicate and related question retrieval or expert finding), we address here the cQA task only, by illustrating how it can be addressed by using the resources in SE-PQA . The addressed cQA task focuses on satisfying the information needs expressed in user questions by retrieving relevant documents from a collection of historical answers posted by the community members. We infer the relevance of an answer to a question from the number of up-votes given by community members. Concerning the experiments involving personalized cQA models, we only consider relevant the single answer that is explicitly labeled as the best answer by the user who submitted the question. In order to address the above-defined cQA task, we preprocess the collection of answers of SE-PQA . Specifically, we discard answers with negative scores since they are assumed to be of low quality and not relevant to the cQA task. This cleaning step affects about 100k answers. As a result, 2, 073, 370 answers are left in the dataset. Moreover, we discard all the questions that have not received an answer. To create the set of relevant answers for the questions, *i.e., the golden standard*, we consider the answers

[1] https://doi.org/10.5281/zenodo.7940964

given to each question. A total of 525,030 questions out of 1,001,706 have an answer selected as the best by the user who asked the question. We are sure that this answer received a positive score from the community since we have removed all the answers with negative scores. Thus, the answer given to a question **q** of the user **u** and selected as best can be considered as relevant both to the question **q** (positive score from the community) and to the user **u** (selection of the best answer). By using this information, we define two versions of this dataset: the *base* version, where we consider as relevant for a question all the answers having a positive score, and the personalized (*pers*) version, which, instead, considers relevant for both the user and the question only the single answer that the user selected as the best answer. We note that both versions of the dataset, *base* and *pers*, can be used for personalized cQA, with the following difference: in the *pers* version each query is potentially personalizable, while the *base* version also includes queries that cannot always be used to train personalized models since the choices of the answers preferred by the users are not always available. A variety of user-generated information from the training set can be used in the personalization phase. For each question, we include all the user posts (questions and answers of the user asking the question) that were written prior to the question being asked. This is done to avoid any data leakage for query-wise training, but the user data is not limited to these documents; in fact, one can also consider the social interaction between users, the tags assigned by the users to the previous questions asked along with their meaning, the badges earned by users. Furthermore, the dataset includes the biographic text (*about me*) self-introducing each user, a rich set of numeric features (e.g., user reputation score, number of up-votes and down-votes of each post, number of views), a set of temporal information (e.g., user and post creation date, last access date).

3 PRELIMINARY EXPERIMENTS WITH SE-PQA

In this section, we describe the experimental setup and introduce the methods employed to showcase SE-PQA on the cQA task. Finally, we discuss the results of the preliminary experiments conducted.

Experimental settings. We adopt a two-stage ranking architecture aimed at trading-off effectiveness and efficiency by applying two increasingly accurate and computationally expensive ranking models. The first stage is inexpensive and recall-oriented. It aims at selecting for each query a set of candidate documents that are eventually re-ranked by the second, precision-oriented ranker. The first stage is based on elasticsearch, and uses BM25 as a fast ranker. To increase the recall in the set of candidate documents retrieved by the first stage, we optimize BM25 parameters by performing a grid search driven by Recall at 100 on a subset of 5000 queries randomly sampled from the validation set. The optimal values found for *b* and *k1* are 1 and 1.75, respectively. For the second, precision-oriented stage, we rely on a linear combination of the scores computed by BM25, a neural re-ranker based on a pre-trained language model, and, when used, a personalization model exploiting the user history, represented by the tags used by the users. In all the experiments the second stage re-ranks the top 100 results retrieved with BM25.

Neural models. In the second stage of our approach, three neural models are employed. The first model is MiniLM, which was trained and tuned using billions of training pairs, including StackExchange data, therefore it is utilized without further fine-tuning. The second one is DistilBERT, and the third one is MonoT5-small. For the DistilBERT and MonoT5-small models, fine-tuning is performed using all the training queries of SE-PQA . In both cases, for each query, one positive document and one negative document is randomly sampled from the list retrieved by BM25. Additionally, for DistilBERT, an in-batch random negative is further sampled [8]. We fine-tune DistilBERT for 10 epochs, with a batch size of 16 and a learning rate of 10^{-6} by using Triplet Margin Loss, with a margin $\gamma = 0.5$. MonoT5-small [11] is based on a T5-small re-ranker, which is fine-tuned on the MS MARCO passage dataset. To further fine-tune MonoT5-small, we follow the same setting proposed in [11]. We rely on Adapter modules [6, 13], which are trained for 10 epochs and have a hidden dimension of 48. We use AdamW as the optimizer and set the random seed to 42 for reproducibility purposes.

Personalized TAG model for cQA. For a given answer **a** produced in response to a query **q** formulated by a user **u**, a personalisation score is computed as explained below. Given a question **q**, asked by user **u** at time **t**, let $T_{u,t}$ be the set of tags assigned by **u** to all her/his questions posted before **t** (including **q**). $T_{u,t}$ thus represents the interests of **u** as expressed in her/his previous interactions. The authors of the answers to the query **q** do not have the possibility of tagging explicitly their answers, so for each answer, we consider the tags associated with the answered questions. Specifically, given an answer **a** from a user **u'**, we represent **a** with the set $T_{u',t}$, i.e., the set of all the tags associated to the questions to which **u'** answered before **t** (excluding **q**). In computing $T_{u',t}$, we do not consider the tags associated with the current question **q** to avoid data leakage. The TAG model assigns to each answer **a**, which has been retrieved for the question **q** in the first stage, the following score:

$$s_a = \frac{|T_{u',t} \cap T_{u,t}|}{|T_{u,t}| + 1},$$

where we add 1 in the denominator as a smoothing factor, needed for cases where set $T_{u,t}$ is empty. The rationale behind the proposed formula is that an answer is assigned a higher score if the question author shares similar interests (represented by means of tags he/she assigned to her/his asked questions) to the answerer.

Score combination. The final ranking is obtained by computing the weighted sum of the normalized scores from the individual models mentioned above, i.e., by using the weights λ_{BM25}, λ_{Neural}, and λ_{TAG}, for the BM25, neural and TAG models, respectively, with $\sum_i \lambda_i = 1$. The λ values are optimized on the validation set by performing a grid search in the interval $[0, 1]$ with step 0.1.

Evaluation Metrics and Results. We use P@1, NDCG@3, NDCG@10, Recall@100, and MAP@100 as our evaluation metrics. The cut-offs considered are low as it is important to find the relevant results at the top of the ranked lists. All the metrics are computed by using the *ranx* library [3] The results of the experiments conducted are reported in Table 1 and 2 for the *base* and *pers* versions of the dataset, respectively. In all tables, the symbol * indicates a statistically significant improvement over the respective non-personalized method not using any contribution from the TAG model. Statistical significance is assessed with a Bonferroni-corrected two-sided paired

Table 1: Results for the cQA task on Base SE-PQA .

Model	P@1	NDCG@3	NDCG@10	R@100	MAP@100	λ
BM25	0.330	0.325	0.359	0.615	0.320	-
BM25 + TAG	0.355*	0.349*	0.383*	0.615	0.342	(.7;.3)
BM25 + DistilBERT	0.404	0.400	0.435	0.615	0.389	(.3;.7)
BM25 + DistilBERT + TAG	0.422*	0.415*	0.448*	0.615	0.402*	(.3;.5;.2)
BM25 + T5	0.448	0.442	0.471	0.615	0.426	(.1;.9)
BM25 + T5 + TAG	0.463*	0.454*	0.482*	0.615	0.436*	(.1;.8;.1)
BM25 + MiniLM	0.473	0.459	0.486	0.615	0.443	(.1;.9)
BM25 + MiniLM + TAG	**0.493***	**0.475***	**0.500***	0.615	**0.457***	(.1;.8;.1)

Table 2: Results for the cQA task on Pers SE-PQA .

Model	P@1	NDCG@3	NDCG@10	R@100	MAP@100	λ
BM25	0.279	0.353	0.394	0.707	0.362	-
BM25 + TAG	0.306*	0.383*	0.425*	0.707	0.392*	(.7;.3)
BM25 + DistilBERT	0.351	0.437	0.478	0.707	0.441	(.3;.7)
BM25 + DistilBERT + TAG	0.375*	0.460*	0.500*	0.707	0.463*	(.3;.5;.2)
BM25 + T5	0.376	0.469	0.506	0.707	0.468	(.1;.9)
BM25 + T5 + TAG	0.400*	0.491*	0.525*	0.707	0.489*	(.1;.8;.1)
BM25 + MiniLM	0.403	0.491	0.525	0.707	0.490	(.1;.9)
BM25 + MiniLM + TAG	**0.426***	**0.512***	**0.543***	0.707	**0.509***	(.1;.8;.1)

student's t-test with 99% confidence. Moreover, in the column labeled λ we report the optimized weights used for combining the scores computed by BM25, Neural and TAG models. From the two tables, we notice first that neural re-rankers are effective and that the methods using MiniLM outperform those based on DistilBERT and T5. This was somehow expected due to the huge training set used to train MiniLM. Moreover, it is worth noting that also DistilBERT and T5, fine-tuned for just 10 epochs on the proposed dataset, improves by nearly 22% and 33% in MAP@100 the BM25 performance, respectively. However, the most notable result is that TAG improves, by a statistically significant margin, any cQA method it is combined with, thus showing the advantages of personalization. The improvements are apparent for all the metrics considered and are larger on the *pers* version of the dataset (Table 2), where non-personalizable queries are removed. Finally, in order to validate our hypothesis that personalization is more useful on a multi-domain, heterogeneous collection than on a single-domain, homogeneous one, we perform a series of experiments considering single-domain data extracted from SE-PQA . Specifically, we consider 50 partitions of SE-PQA (only base version) built by isolating the data from the 50 communities. We apply to each one of these subsets the non-personalized and personalized combinations of models using the best performing MiniLM, and measure the performance according to the same metrics used for the previous cQA tests. For a fair comparison, we performed for each community the optimization of the λ weights on single-domain validation data. For 25 out of 50 communities, we notice that personalization does not lead to any improvement, i.e., $\lambda_{TAG} = 0$, while on other 13 communities, we do not observe statistically significant improvements for P@1 over the non-personalized methods. Since statistical significance is also affected by the size of the sample, we also computed the performance metrics averaged on all the runs with single-domain data. In terms of the absolute performance boost due to the TAG model, we achieve a 2% improvement on P@1 when using all communities together, while the boost decreases to 1.1% when considering the communities separately. The results of these experiments are reported integrally in the Zenodo page.

4 CONCLUSION AND FUTURE WORKS

We expect SE-PQA dataset being useful for many researchers and practitioners working in personalized IR and in the application of deep learning techniques for personalization. In recent years, the IR community spent important effort in studying personalization. However, a comprehensive dataset for evaluating and comparing different approaches is still missing. This work aims to fill this gap with a large-scale dataset covering the activity of StackExchange users in a time span of 14 years. We detailed all the information available in the dataset and discussed how it can be exploited for training and evaluating classical and personalized models for the cQA task. We focused on IR approaches based on a two-stage architecture where the second re-ranking stage exploits a combination of the scores computed by BM25, Neural, and TAG models. The results of the preliminary experiments conducted show that personalization works effectively on this dataset, improving by a statistically significant margin state-of-the-art methods based on pre-trained large language models. The analysis conducted and the peculiarities of the SE-PQA resource suggest several lines of future investigation. For example, in this work we employed a relatively simple user model for personalization, and we leave the development of more complex personalized models for future works that could exploit user features of SE-PQA that were not used in the proposed models.

ACKNOWLEDGMENTS

We acknowledge the CINECA award under the ISCRA initiative, for the availability of high-performance computing resources and support.

REFERENCES

[1] Qingyao Ai, Yongfeng Zhang, Keping Bi, Xu Chen, and W. Bruce Croft. 2017. Learning a Hierarchical Embedding Model for Personalized Product Search. In *Proc. of SIGIR '17 (SIGIR '17)*.
[2] Michael Barbaro, Tom Zeller, and Saul Hansell. 2006. A face is exposed for AOL searcher no. 4417749. *New York Times* 9, 2008 (2006), 8.
[3] Elias Bassani. 2022. ranx: A Blazing-Fast Python Library for Ranking Evaluation and Comparison. In *ECIR*.
[4] Elias Bassani, Pranav Kasela, Alessandro Raganato, and Gabriella Pasi. 2022. A Multi-Domain Benchmark for Personalized Search Evaluation. In *CIKM '22*.
[5] Alexey Borisov, Ilya Markov, Maarten de Rijke, and Pavel Serdyukov. 2016. A Context-Aware Time Model for Web Search. In *Proc. of ACM SIGIR '16 (SIGIR '16)*. Association for Computing Machinery, New York, NY, USA, 205–214.
[6] Marco Braga, Alessandro Raganato, Gabriella Pasi, et al. 2023. Personalization in bert with adapter modules and topic modelling. In *Proc.s of IIR 2023*. 24–29.
[7] Silvia Calegari and Gabriella Pasi. 2013. Personal ontologies: Generation of user profiles based on the YAGO ontology. *Information Processing & Management* 49, 3 (2013), 640–658. Personalization and Recommendation in Information Access.
[8] Matthew Henderson, Rami Al-Rfou, Brian Strope, Yun-hsuan Sung, Laszlo Lukacs, Ruiqi Guo, Sanjiv Kumar, Balint Miklos, and Ray Kurzweil. 2017. Efficient Natural Language Response Suggestion for Smart Reply.
[9] HuggingFace. 2021. *Train a Sentence Embedding Model with 1B Training Pairs*. HuggingFace. https://huggingface.co/blog/1b-sentence-embeddings
[10] Bhaskar Mitra and Nick Craswell. 2018. An Introduction to Neural Information Retrieval. *Foundations and Trends® in Information Retrieval* 13, 1 (2018), 1–126.
[11] Rodrigo Nogueira, Zhiying Jiang, Ronak Pradeep, and Jimmy Lin. 2020. Document Ranking with a Pretrained Sequence-to-Sequence Model. In *Findings of EMNLP 2020*, Trevor Cohn, Yulan He, and Yang Liu (Eds.).
[12] Greg Pass, Abdur Chowdhury, and Cayley Torgeson. 2006. A Picture of Search. In *Proc. of InfoScale '06*.
[13] Clifton Poth, Hannah Sterz, Indraneil Paul, Sukannya Purkayastha, Leon Engländer, Timo Imhof, Ivan Vulić, Sebastian Ruder, Iryna Gurevych, and Jonas Pfeiffer. 2023. Adapters: A Unified Library for Parameter-Efficient and Modular Transfer Learning. In *Proc. of the EMNLP '23*.
[14] Shayan A. Tabrizi, Azadeh Shakery, Hamed Zamani, and Mohammad Ali Tavallaei. 2018. PERSON: Personalized information retrieval evaluation based on citation networks. *Information Processing & Management* 54, 4 (2018), 630–656.