Retrieval based approach for subjective tasks in NLP

Michał Kajstura, Joanna Baran, Jan Kocoń, Przemysław Kazienko
Wroclaw University of Science and Technology
Wyspiańskiego 27, 50-370 Wrocław, Poland
{joanna.baran, jan.kocon, przemyslaw.kazienko}@pwr.edu.pl, {kajsturamichal}@gmail.com

Abstract

Generalizing models are often unable to produce a personalized prediction for a individual user. Personalized methods often yield better results, but at the cost of an increased training complexity, especially in big-data scenarios, when frequent re-training is necessary. We propose a retrieval based approach that performs similar to SOTA personalization algorithms, but does not require any further optimization for new users.

1. Introduction

Popular classical approaches to natural language processing tasks have for a long time focused solely on developing a general classifier that does not take into account the characteristics of a single person. However, everyday language is influenced by many individual factors like one's mood, emotion, world view, and sociodemographic circumstances. This has become the reason for the dynamic development of the *personalization* trend in NLP, which has been going on for over a dozen years. Perfect for subjective tasks, where obtaining a single label is hard or even impossible, *human-based* approaches were exploited in various ways. Starting with the inclusion of user traits, modelling entire social groups, ending with focusing on individual features - thus creating perfectly tailored NLP systems. From the user's perspective, personalization can occur as an explicit input, when the person is providing information themselves, or by using implicit inference made by specially designed models (Flek 2020). In the scope of this research, we consider the last approach since it is more convenient for users (no additional action is required on their part) and thus better applicable in business.

The example of a highly subjective task is sentiment classification, where the polarity of a text depends on a person's experiences and a character. This also applies to offensive language detection and emotion recognition, as the same piece of text can invoke drastically different reactions. Moreover, it can be argued the single correct label often simply does not exist and trying to enforce it could result in a model biased to a particular culture or world view, which could lead to minorities discrimination (Dixon et al. 2018). Obtaining a gold-standard label from multiple annotations is usually done by instance-wise aggregations, such as mean or majority voting (Basile et al. 2021). The previously mentioned disadvantages can be easily mitigated with a personalized approach, trained on datasets providing unaggregated labels.

Human-centric methods are also useful in a privacy-preserving scenario where the user's data cannot leave their device. Unfortunately, many of the existing methods require frequent re-training for each new user, which usually is done by sending the data to a centralized server. Although federated learning algorithms allow for in-device gradient update computation (Mahlool and Abed 2022), they are still not feasible to use in many situations, because of significant performance overhead, especially in big-data scenarios (Gadekallu et al. 2021).

In this work, we propose a retrieval based approach, which provides personalized results using the text similarity and is easily extendable to the large number of users. We experiment with different ways of capturing the text similarity, including state-of-the-art bi-encoders and cross-encoders. The analysis performed on three datasets shows in which situations the personalized methods work best and where the improvements come from.

2. Related Work

The existing approaches to human-based NLP can be divided into two main groups - based on users personal metadata, and focused on using their past digital traces such likes, ratings, posted texts in social media, etc. The first works that addressed the need to adapt NLP approaches to subjective language have appeared since the 2010s.

2.1. Metadata-based approach

Conceptually easier, attempts to adapt personalization in NLP classification tasks are based on users' metadata and their individual or social-group features. Volkova, Wilson, and Yarowsky 2013 used basic demographic variables (e.g. gender, age) directly as input into the traditional rule-based model aiming to learn gender differences between users. Demographic adaption was also introduced in the work of Hovy 2015 and proved that models aware of that features outperformed their agnostic counterparts. Zamani et al. 2018 designed a residualized control approach by training a language model over the prediction errors of the model using the sociodemographic variables only. Later, the results were combined with factor analysis. For tasks involving human consumption, a categorical metadata was incorporated into low-dimensional basis vectors to various parts of neural networks such as bi-LSTM by Kim et al. 2019. This way of feeding user-product information improved the performance of the model and customized it to the several text classification problems.

2.2. Users trace approach

The most exploited personalization methods in the literature make advantage of digital traces left by the user. It could be published opinions, ratings or shared social network. An interesting approach to group-wise sentiment classification was presented by Gong, Haines, and Wang 2017. Taking shared opinions between different people, the authors' solution introduced a non-parametric Dirichlet Process prior over the individualized models - one for each cluster of users. This lies at the heart of the social comparison theory that humans tend to form groups with others of similar minds and ability. Inspired by the recommendation systems, the latent factor model can also be used to capture user's specific individuality due to different language habit (Kaisong Song et al. 2015).

However, the recent works are mostly focused on usage of deep neural networks instead of classical machine learning techniques, especially on the SOTA transformer-based architectures. In short summary, those approaches often consist of two stages - global, shared model pre-training and local, personalized fine-tuning. In the first phase, the model is trained on a vast amounts of aggregated, non-personalized data, resulting in a global model unable to incorporate person-level information. After that, the shared model is fine-tuned for each user using their data. There are multiple ways of performing this step. The most basic one is to optimize the whole model, which results in a separate set of weights for each user (Schneider and Vlachos 2019), causing a significant computational and storage overhead. There are, however, methods to share a single model between users and learn a unique representation for each person. This representation is then combined with the text representation to produce a user-informed prediction (Zhong et al. 2021, Kocoń et al. 2021). Even though these methods mitigate most of the memory related issues, they continue to require a user embedding optimization, which is considerably easier than training the entire model, but still can be difficult if the number of users is very large, or they change frequently.

Methods based on user label aggregation (Kanclerz et al. 2021) do not require training of user embeddings, and thus can be easily applied in big-data scenarios. However, this algorithm does not take the textual content into account. User embedding is fixed and depends only on past texts written or annotated by the user. This results in poor performance if the evaluated sample is different from the others or introduces a new topic.

3. Retriever

In this section, we first explain how Retriever works, and then discuss the text similarity calculation and the training process.

3.1. Method

Retriever is a method combining text representations obtained from a language model, like RoBERTa, and an aggregated user-level score computed by a retrieval module. Texts previously written or annotated by the user are retrieved from the database. Then the text similarity scores are computed and used to calculate a user score representing their preferences. There are various ways of aggregating multiple labels into a single score. In the experiments, we used a simple mean, weighted average and a KNN based aggregation which averages the labels of the K most similar samples. Textual features are concatenated with a user score. This personalized representation is then passed to a linear classifier for a person-informed prediction. Figure 1a shows components of the entire system.

3.2. Text similarity

In Retriever, text similarity scores influence the aggregation of previous users' text labels. The labels of samples most similar to the current text have the greatest impact on the final user score. Conversely, labels of unrelated texts should be discarded in the aggregation process.

The simplest method of aggregating multiple targets is a weighted arithmetic mean Similarity score s between a pair of texts, play a role of weighting coefficients. If, therefore, another text is very similar to the sample being evaluated, it will have a weight close to 1, and the weight of the differing text will be close to 0.

$$s(t_i, T, L) = \frac{1}{N-1} \sum_{n=1}^{N} \mathbb{1}_{n \neq i} \cdot l_n \cdot \text{similarity}(t_i, t_n) \tag{1}$$

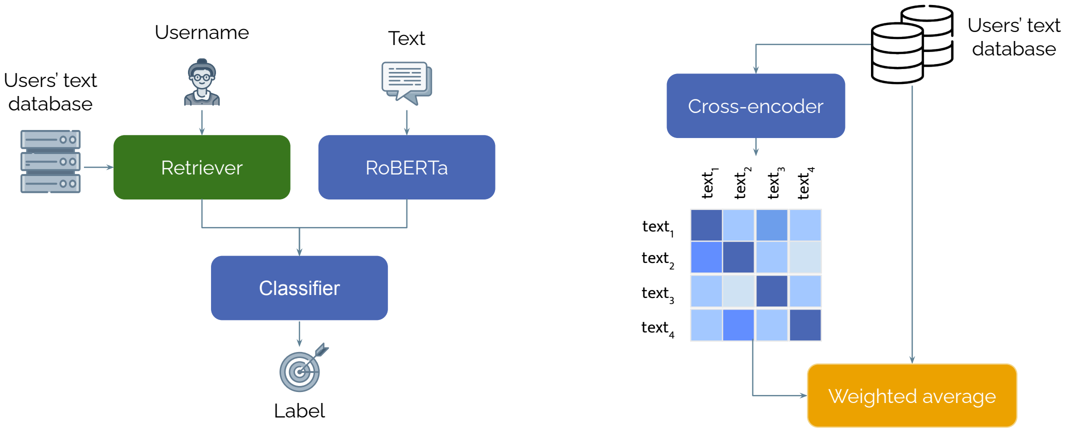
Where s denotes similarity score, t_i is the text being currently predicted, T is a sequence of all user's texts and L is a sequence of all user's labels. Similarly, for KNN based method, only the K most similar samples are considered during label aggregation.

3.3. Training

During the training stage, all the user's texts and labels from the training set, apart from the currently used sample, are utilized to compute an aggregated score. Also for validation and testing, the method considers only the labels of training examples, preventing data leakage.

The model is trained to minimize a standard cross-entropy loss for classification with respect to a single, shared parameter set θ.

$$\mathcal{L}_{\text{CE}}(t_i, y_i, T, L; \theta) = -\log Pr(y_i | [t_i; s(t_i, T, L)]) \tag{2}$$



(a) Retrieval based system

(b) Retrieval module

Figure 1: An overview of the proposed method - Retriever. It combines text representation from a language model with an aggregated user score. The average user label is weighted by a text similarity computed using a cross-encoder.

$$\theta = \underset{\theta}{\operatorname{argmin}} \mathcal{L}_{\text{CE}}(t_i, y_i, T, L; \theta) \tag{3}$$

Where y_i denotes the class of i-th example.

4. Experiments and Results

4.1. Tasks and Datasets

We evaluated methods on three datasets involving different tasks. Sentiment140 (Go 2009) is a heuristically annotated dataset of tweets that was collected using a predefined set of queries. The annotations are based on emoticons contained in the texts. We used the data split methodology described by (Li, Sanjabi, and Smith 2019) which determined the proportion of user partitioning.

For the IMDB dataset (Diao et al. 2014), which involves film reviews from a popular website, we used the data splits proposed by Zhong et al. 2021. The user reviews were divided into training, validation and test in the ratio of 0.8, 0.1, 0.1, meaning that a single user appears in all splits. In addition, only users with less than 50 samples are considered to simulate the low data regime scenario.

Measuring Hate Speech dataset offers multiple dimensions like sentiment analysis, insults, incitement to violence and hate speech detection. We focused on the latter task because it proved difficult for generalizing, or non-personalized, methods.

4.2. Experimental setup

We used the RoBERTa-base language model (Liu et al. 2019) both as baseline and in personalized approaches. For text similarity calculations, we utilized MPNet based Bi-Encoder, trained on various text-pair datasets (Kaitao Song et al. 2020). The Cross-Encoder was based on a RoBERTa trained on Semantic Text Similarity Benchmark (Cer et al. 2017). Both text similarity models are available in the Sentence Transformers library (Reimers and Gurevych 2019).

Models were fine-tuned using AdamW optimizer with learning rate 1e-5, linear warm up schedule, batch size 16, and maximum sequence length 512 for 50000 training steps, and the best model was selected according to the validation F-score. For the KNN-based aggregation, K was set to 3. Experiments were repeated 5 times and the mean F1 score was reported.

In order to analyze the performance of the personalized approach, we created separate subsets of the Sentiment140 dataset. The first subset contained only highly polarized users, that is, users for whom the fraction of positive or negative tweets was above 70%. We noticed that for many users, the similarities between texts were almost uniform, resulting in a poor performance of the proposed model. If the two texts are almost identical and differ in labels, the retriever module cannot fetch relevant tweets. The second subset was formed from users who wrote different texts and aims to highlight the advantages of the proposed method, as it uses the content-aware aggregation of user's scores based on text similarity. The text diversity was measured as a standard deviation of cosine similarity in a user's text similarity matrix. Individuals with a diversity above 0.5 were considered to be writing diverse tweets.

4.3. Results

A comparison of the methods on the three datasets is shown in Table 4.3. While the UserIdentifier outperforms other methods for Sentiment140 and IMDB datasets, Retriever-Cross's performance is almost the same without the need to re-train for each new user. For the Measuring Hate Speech dataset, UserIdentifier is worse than the standard generalizing RoBERTa model, while both Retriever variants perform better. A simple Retriever-Mean aggregation method that uses a basic arithmetic mean to calculate the user score, offers a slight improvement over the baseline, but performs worse than other alternatives. Retriever-Cross achieves better results, but at the significantly higher computa-

	Sentiment140	IMDB	MHS
Baseline	85.9.	46.7	56.3
UserIdentifier	**87.9**	**50.4**	56.1
Retriever-Mean	86.3	47.7	56.6
Retriever-Bi	87.1	48.8	56.7
Retriever-Cross	87.7	49.4	**57.2**
Retriever-Cross-KNN	87.4	49.8	56.6

Table 1: UserIdentifier outperforms other methods for Sentiment140 and IMDB datasets. For Measuring Hate Speech (MHS) both variants of Retriever outperform RoBERTa (baseline) and UserIndentifier.

Subset	Retriever-Mean	Retriever-Cross
All	86.3	87.7
Polarized	90.6	90.5
Diverse	87.0	89.3

Table 2: The performance of Retriever-Cross and Retriever-Mean models for artificial data subsets. Both Retriever versions achieve a considerably better results than for the standard Sentiment140. Retriever-Mean works well for the Polarized subset, as the content of texts is not relevant in this case. However, for the Diverse subset, the results are lower than it's content-aware alternative.

tional cost, because it requires performing expensive forward passes through the network for each text pair. The KNN-based aggregation performs similarly to the standard weighted average.

The results for artificially created data subsets, described in section 4.1, are presented in Table 4.3. The performance of Retriever-Cross model for both Polarized and Diverse data subsets is considerably higher than for the standard Sentiment140. The High F1-score for Polarized split can be attributed to the fact that it consists of users expressing opinions with similar sentiment. Personalized approaches can use the labels of the previously written posts, so it naturally excels when the user is single-sided. For the Diverse subset, the text retrieval module is more useful, because a certain small number of text pairs are more closely related than the rest. As a result, the aggregated user's score better reflects the individual's preference for the currently evaluated text.

5. Discussion

5.1. Conclusions

Our proposed Retriever model proved to perform on par with other personalized SOTA approach which uses specially trained unique human representation. However, choosing text similarity between past users' written opinions has one major advantage above other methods - frequent retraining is not necessary. This makes Retriever easier to deploy in real-world applications. The only limitation of the model is the need to have a certain number of texts from a single person in order to correctly predict the label in a subjective task.

We also noticed that in a scenario where training samples are different from each other by context, our model achieved the highest score, suggesting that for difficult tasks of a varied nature, it may be the best choice. This is confirmed by the results for the Measuring Hate Speech dataset, where the texts concern different subjects and topics, in contrast to IMDB - here we find only film reviews, often with similar content.

5.2. Future work

To extend our work in the future, we plan to further train the cross-encoder part of the model to provide similarity scores of even higher quality. Some optimization techniques to reduce compute overhead should also be applied. Finally, to better assess the performance of the proposed method, the experimental part must be extended to include a comparison to other human-centric models mentioned in Section 2.

References

Flek, Lucie (July 2020). "Returning the N to NLP: Towards Contextually Personalized Classification Models". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, pp. 7828–7838.

Dixon, Lucas et al. (2018). "Measuring and Mitigating Unintended Bias in Text Classification". In.

Basile, Valerio et al. (2021). "Toward a Perspectivist Turn in Ground Truthing for Predictive Computing". In: *CoRR* abs/2109.04270.

Mahlool, Dhurgham Hassan and Mohammed Hamzah Abed (2022). "A Comprehensive Survey on Federated Learning: Concept and Applications". In: *CoRR* abs/2201.09384.

Gadekallu, Thippa Reddy et al. (2021). "Federated Learning for Big Data: A Survey on Opportunities, Applications, and Future Directions". In: *CoRR* abs/2110.04160.

Volkova, Svitlana, Theresa Wilson, and David Yarowsky (Oct. 2013). "Exploring Demographic Language Variations to Improve Multilingual Sentiment Analysis in Social Media". In: *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Seattle, Washington, USA: Association for Computational Linguistics, pp. 1815–1827.

Hovy, Dirk (July 2015). "Demographic Factors Improve Classification Performance". In: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Beijing, China: Association for Computational Linguistics, pp. 752–762.

Zamani, Mohammadzaman et al. (2018). "Residualized Factor Adaptation for Community Social Media Prediction Tasks". In.

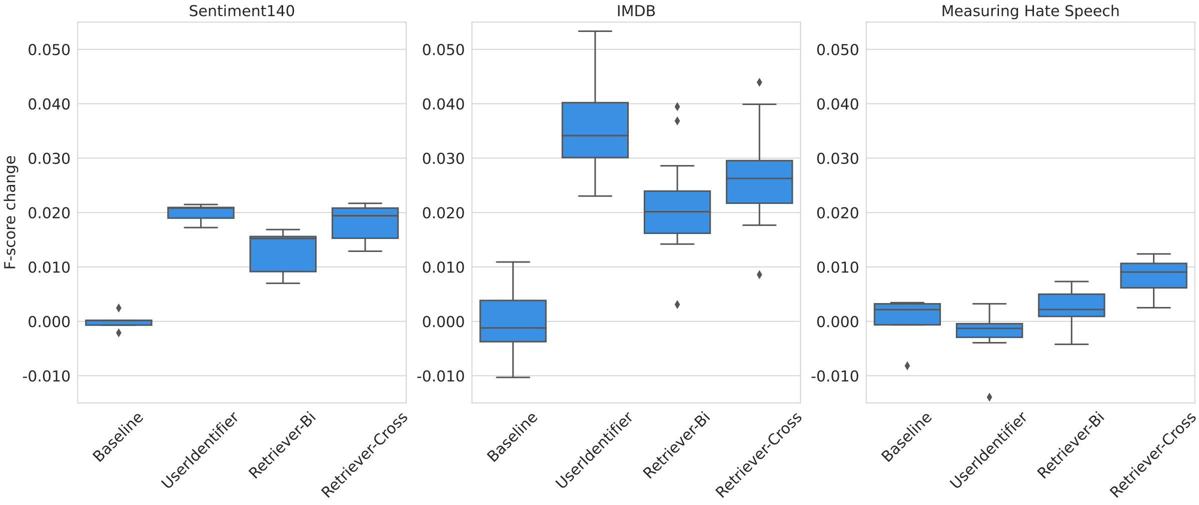


Figure 2: Performance comparison between examined methods. Reported F1 scores are relative to the mean of the baseline F1 score. The proposed method is better than the baseline method in all cases and matches the UserIdentifier, except for the Measuring Hate Speech dataset, for which it proved to be better. Furthermore, Retriever using a cross-encoder to measure text similarity was more effective for all datasets. Retriever-Bi is a model using Bi-Encoder, and Retriever-Cross uses a Cross-Encoder as a retriever module.

Kim, Jihyeok et al. (2019). *Categorical Metadata Representation for Customized Text Classification*.

Gong, Lin, Benjamin Haines, and Hongning Wang (2017). “Clustered Model Adaption for Personalized Sentiment Analysis”. In: *Proceedings of the 26th International Conference on World Wide Web*. WWW ’17. Perth, Australia: International World Wide Web Conferences Steering Committee, pp. 937–946.

Song, Kaisong et al. (July 2015). “Personalized Sentiment Classification Based on Latent Individuality of Microblog Users”. In.

Schneider, Johannes and Michail Vlachos (2019). “Mass Personalization of Deep Learning”. In: *CoRR* abs/1909.02803.

Zhong, Wanjun et al. (2021). “UserAdapter: Few-Shot User Learning in Sentiment Analysis”. In: *FINDINGS*.

Kocoń, Jan et al. (2021). “Learning Personal Human Biases and Representations for Subjective Tasks in Natural Language Processing”. In: *2021 IEEE International Conference on Data Mining (ICDM)*, pp. 1168–1173.

Kanclerz, Kamil et al. (2021). “Controversy and Conformity: from Generalized to Personalized Aggressiveness Detection”. In: pp. 5915–5926.

Go, Alec (2009). “Sentiment Classification using Distant Supervision”. In.

Li, Tian, Maziar Sanjabi, and Virginia Smith (2019). “Fair Resource Allocation in Federated Learning”. In: *CoRR* abs/1905.10497.

Diao, Qiming et al. (Aug. 2014). “Jointly modeling aspects, ratings and sentiments for movie recommendation (JMARS)”. In: *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. KDD ’14. New York, NY, USA: Association for Computing Machinery, pp. 193–202.

Liu, Yinhan et al. (2019). “RoBERTa: A Robustly Optimized BERT Pretraining Approach”. In: *CoRR* abs/1907.11692.

Song, Kaitao et al. (2020). “MPNet: Masked and Permuted Pre-training for Language Understanding”. In: *CoRR* abs/2004.09297.

Cer, Daniel et al. (Aug. 2017). “SemEval-2017 Task 1: Semantic Textual Similarity Multilingual and Crosslingual Focused Evaluation”. In: *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. Vancouver, Canada: Association for Computational Linguistics.

Reimers, Nils and Iryna Gurevych (Nov. 2019). “Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks”. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.