Topic-Regularized Authorship Representation Learning

Jitkapat Sawatphol, Nonthakit Chaiwong,
Can Udomcharoenchaikit, and **Sarana Nutanong**.
School of Information Science and Technology,
Vidyasirimedhi Institute of Science and Technology, Thailand
{jitkapat.s_s20,nonthakitc_pro,canu_pro,snutanon}@vistec.ac.th

Abstract

Authorship attribution is a task that aims to identify the author of a given piece of writing. We aim to develop a generalized solution that can handle a large number of texts from authors and topics unavailable in training data. Previous studies have proposed strategies to address only either unseen authors or unseen topics. Authorship representation learning has been shown to work in open-set environments with a large number of unseen authors but has not been explicitly designed for cross-topic environments at the same time. To handle a large number of unseen authors and topics, we propose Authorship Representation Regularization (ARR), a distillation framework that creates authorship representation with reduced reliance on topic-specific information. To assess the performance of our framework, we also propose a cross-topic-open-set evaluation method. Our proposed method has improved performances in the cross-topic-open set setup over baselines in 4 out of 6 cases.

1 Introduction

Authorship attribution is a task that aims to identify the authors of anonymous texts. Applications of this task include academic and forensic ones, such as finding the authors of literary works, historical writings (Koppel and Seidman, 2013; Juola, 2013; Stover et al., 2016) or threatening online messages (Abbasi and Chen, 2005; Lambers and Veenman, 2009; Coulthard, 2012).

Solution Design Factors. Three factors affect our solution design. First, our technique should be able to handle a large number of authors due to the endless number of candidate authors in the real world. Second, we want our technique to allow style comparison of texts written by unseen authors so that we do *not* have to adjust the model every time a new author is introduced. Third, our technique should be effective with out-of-distribution topics (Mikros and Argiri, 2007) since it is impractical to assume that the training data covers all possible topics during runtime.

Existing Techniques. Prior research efforts on authorship attribution have focused on solving either out-of-distribution in topics or authors. For out-of-topic, methods such as text distortion (Stamatatos, 2017), multi-task learning (Song et al., 2019), and data augmentation (Rivera-Soto et al., 2021) have been used in conjunction with classification algorithms to reduce topic bias and improve performance on unseen topic texts. For out-of-author, Hay et al. (2020) and Rivera-Soto et al. (2021) have used representation learning to handle thousands of unseen authors. Such methods aim to convert texts into fixed-length embeddings. This paradigm allows the comparison of unseen author texts without pre-defining a fixed number of author classes at training time. Yet, to the best of our knowledge, no study has proposed a representation learning method that is explicitly designed to deal with out-of-topic and out-of-author simultaneously.

Proposed Research. In this paper, we propose *Authorship Representation Regularization (ARR)*. Our objective is to enhance the cross-topic capability of authorship representation models that can handle a large number of unseen authors. The principle of our method lies in the self-distillation framework that reduces the authorship representation's reliance on topic-specific information. Our experimental results reveal improvements in large-scale cross-topic-open-set authorship attribution over existing representation learning baselines in 4 out of 6 cases, as well as demonstrated minimal performance tradeoff in in-distribution-topic setup.

Contributions. Our work has the following contributions:

(i) We propose *Authorship Representation Regularization (ARR)*, a framework that can be applied to enhance cross-topic performances of any existing authorship representation encoders with any model architecture.

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 1076 - 1082
December 7-11, 2022

(ii) We introduce an evaluation method to assess the performance of *cross-topic-open-set* authorship attribution methods.

(iii) Our proposed framework achieves improved performances in cross-topic-open-set setup over baselines in 4 out of 6 cases.

2 Proposed Method

Authorship representation learning has shown to be effective for large-scale open-set authorship attribution (Hay et al., 2020; Rivera-Soto et al., 2021). However, these approaches have not been explicitly designed to help with generalization toward unseen topics. We hypothesize that we can improve generalization by reducing the topic information of an authorship representation. Therefore, we propose a solution based on the concept of supervised contrastive learning (Khosla et al., 2020) and confidence regularization (Utama et al., 2020). Our framework can be applied to remove bias from any text encoder model regardless of the architecture.

We propose *Authorship Representation Regularization (ARR)*, a framework to obtain topic-regularized authorship representation, i.e., an embedding that can be used to compare writing style similarity with minimum topic influence.

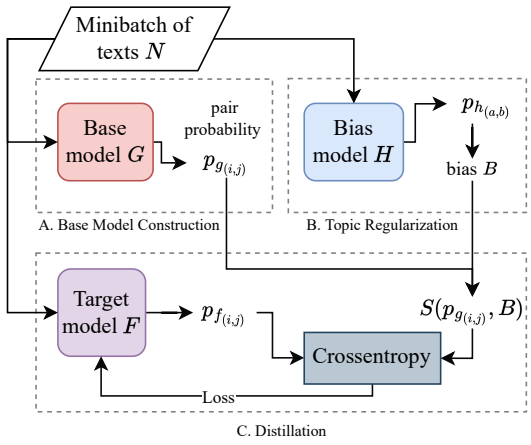

Figure 1: An illustration of the training pipeline for our Authorship Representation Regularization framework.

Our pipeline consists of three steps. **Step A: Base Model Construction.** Construct a base model G for authorship representation. **Step B: Topic Regularization.** Create a bias model H and re-scale the base model's output to reduce topic dependency. **Step C: Distillation.** Transfer knowledge into target model F to create a topic-regularized embedding space. As shown in Figure 1, these training steps are described as follows.

Step A: Base Model Construction. First, we train a base encoder G on an authorship representation learning objective. We then freeze all the parameters of the base model.

At target encoder training time, we sample a minibatch represented by a set of texts N. We calculate the probability score $p_{g_{(i,j)}}$ from the cosine similarity score of each pair $(i, j) \in N \times N$ to use in the next step.

We define $N \times N = \{(i, j) : i \in N \wedge j \in N\}$. For each (i, j), we compute cosine similarity of encoded representation of text i and another text j from encoder G. We denote the L2 normalized representation of text i computed from G as g_i and denote variable τ as the temperature scaling hyperparameter.

$$\mathrm{p}_{g_{(i,j)}} = \frac{\exp(g_i \cdot g_j)/\tau}{\sum_{k=1/\{i\}}^{|N|} \exp(g_i \cdot g_k)/\tau} \quad (1)$$

We also use Eq. 1 to derive probability score $p_{h_{(a,b)}}$ from text pairs encoded by topic bias model H and $p_{f_{(i,j)}}$ from target model F. We only calculate scores for text pairs where $i \neq j$.

Step B: Topic Regularization. We perform topic regularization using a bias model H that is designed to encode topic similarity. We use TF-IDF as a proxy for a topic bias model.

We denote $(a, b) \in M \subset N \times N$. where M only includes text pairs (a, b) with the same author. Afterward, we compute the probability score $p_{h_{(a,b)}}$ derived from the similarity score of the vector representations of text pairs (a, b), encoded by bias model H. Then, we aggregate $p_{h_{(a,b)}}$ into a single value B. For each minibatch, B represents the degree of topic bias for every same-author pair in the minibatch.

$$B = \frac{1}{|M|} \sum_{\substack{(a,b)\in M \\ a \neq b}} \mathrm{p}_{h_{(a,b)}} \quad (2)$$

After obtaining B, we apply a scaling function S to $p_{g_{(i,j)}}$ and B to obtain a topic-regularized probability score $S(\mathrm{p}_{g_{(i,j)}}, B)$.

$$S(\mathrm{p}_{g_{(i,j)}}, B) = \frac{\mathrm{p}_{g(i,j)}^{(1-B)}}{\sum_{k=1}^{|N|} \mathrm{p}_{g(i,k)}^{(1-B)}} \quad (3)$$

Step C: Distillation. Finally, we train the target model with the same model architecture and pre-trained weights as the base model. We minimize the loss function calculated from each text pair (i, j). The loss function is the cross-entropy

between the base model's topic-regularized probability score and the target model's probability score. The final loss value is computed from a mean of $L_{i,j}$ for some i and j where $i \neq j$.

$$L_{(i,j)} = S(\mathrm{p}_{g_{(i,j)}}, B) \cdot \log(\mathrm{p}_{f_{(i,j)}}) \quad (4)$$

At inference time, the target model will be a single encoder that can produce a representation similar to the base model representation with topic regularization applied.

3 Evaluation Method

As stated in Section 1, we want our solution to handle a large number of classes as well as deal with texts from both unseen authors and topics. This section describes the strategies to assess our method as follows.

Dataset. To assess the capability to handle a large number of authors, we choose three datasets that contain thousands of authors from three heterogeneous genres: Amazon reviews (Ni et al., 2019), Reddit (Baumgartner et al., 2020), and Fanfiction (Bevendorff et al., 2020, 2021).

Train-validation-test split. To measure the capability to handle unseen authors and topics, we propose a train-test split scheme to create a cross-topic open-set environment for authorship attribution, as illustrated in Figure 2. This scheme can be used with any data labeled with author and topics. The number of samples, authors, and topics of the datasets we used in our experiments are described in Table 1.

	Training		
	samples	author	topic
Amazon	1,312,124	2,118	4
Reddit	1,457,517	37,517	4,849
Fanfiction	131,812	25,390	1,200
	Cross-topic test		
	samples	author	topic
Amazon	164,015	2,118	18
Reddit	206,693	37,517	13,251
Fanfiction	39,998	27,613	400
	In-distribution-topic test		
	samples	author	topic
Amazon	164,016	2,118	4
Reddit	196,658	30,975	4,720

Table 1: Dataset statistics on Amazon, Reddit and Fanfiction after applying our data split scheme

Training data. First, we split the training portion from the original dataset by randomly selecting samples from the authors and topics that have the most samples. For each dataset, we use manually selected thresholds to determine the candidates for training data, which is elaborated in Appendix A.2.

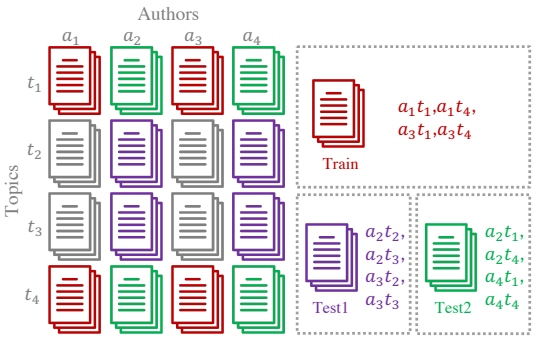

Figure 2: An illustration of our train-test split scheme. The x-axis shows the samples written by authors $[a_1, a_2, a_3, ...]$, sorted from authors with the most samples (left) to the least (right). The y axis represents the samples in topics $[t_1, t_2, t_3, ...]$, sorted from topics with the most samples (top) to least (bottom). After sampling a portion into the training set (red), the eligible candidates for the test set is purple for cross-topic (Test1) and green for in-distribution-topic (Test2).

Cross-topic test data. This test set aims to describe the effectiveness of feature representations in a setup where observed topical and authorship information might have minimal benefits. Therefore, we sample the cross-topic-open-set test data so that the test author and topic set do not overlap with the training set.

In-distribution-topic test data. Additionally, we want to assess our method's performance in an in-distribution environment. Therefore, we sample another in-distribution-topic test data to measure performance. In this scenario, the author set in test data does not overlap with the training data.

Validation data. We also randomly sample the training data into a smaller subset to use as validation data to tune hyperparameters during the training process. The size of the validation set is randomly selected to be the same as the cross-topic test set.

Comparative Studies. We compare our method against existing authorship representation techniques using the described train-validation-test split. For each model and hyperparameter setting, we train on three different random seeds. For each seed, we validate the model to pick the best hyperparameter, then evaluate with each of the two described test data. For each model, we report the mean score of the three seeds in Section 4.

Competitive Methods. Transformer-based (Vaswani et al., 2017) models has shown high performance in large-scale authorship attribution with unseen authors (Rivera-Soto et al., 2021). Therefore, we compare our method with two models based on transformers: **Multiclass log loss (MLL)** (Hay et al., 2020) and **Contrastive loss (CL)** (Rivera-Soto et al., 2021; Khosla et al., 2020). We use pre-trained sBERT (Reimers and Gurevych, 2019)[1] as the base encoder. Then, we apply mean pooling to the hidden vectors from the last encoder layer. Finally, we fine-tune the encoder with one of the two loss functions. We also include **zero-shot** results from the sBERT model without fine-tuning. Additionally, we also include two simple statistical representation: **Bag of words (BOW)** and **Term frequency-inverse document frequency (TF-IDF)**.

Evaluation Measures. We use evaluation process and metrics with respect to that of Rivera-Soto et al. (2021). At testing time, we further divide the test data into two subsets. Firstly, we pick 50% of each author's texts and use them as a query set. We use the rest of the test samples as a target set. Additionally, we also add texts from authors with only a single sample into the target set to serve as distractors. For each query in the query set, we perform a nearest neighbor search using cosine similarity on the encoded representation of each query and text in the target set. We use recall@8 (R@8) and mean reciprocal rank (MRR) as the performance metrics in our experiments.

4 Experimental Results

We conducted experimental studies according to the evaluation method described in Section 3. Tables 2 and 3 show results from the cross-topic and in-distribution-topic studies, respectively.

Cross-topic. Table 2 shows that ARR provides improvements in 4 out of 6 cases, i.e., 3 out of 3 for MLL and 1 out of 3 for CL. The performances of both MLL and CL baselines for the Amazon dataset are improved by 1.9% for R@8 and 2.25% for MRR on average. Also, in Reddit and Fanfiction dataset, there are improvements in the MLL baseline at an average of 6.95% for R@8 and 8.7% for MRR. However, we have also observed performance penalties for CL baseline at 1.4% for R@8 and 1% for MRR with our method applied.

[1]https://huggingface.co/sentence-transformers/paraphrase-distilroberta-base-v1

	Amazon		Reddit		Fanfiction	
	R@8	MRR	R@8	MRR	R@8	MRR
BOW	0.401	0.270	0.073	0.061	0.300	0.262
TF-IDF	0.364	0.235	0.128	0.091	0.235	0.221
Zero-shot	0.429	0.286	0.125	0.090	0.154	0.134
MLL	0.720	0.583	0.167	0.114	0.080	0.011
MLL + ARR	**0.731**	**0.597**	**0.182**	**0.124**	**0.204**	**0.176**
CL	0.699	0.560	**0.253**	**0.175**	**0.333**	**0.264**
CL + ARR	**0.726**	**0.591**	0.247	0.170	0.310	0.249

Table 2: Experimental results on cross-topic-open-set setup. "X+ARR" denotes a base model X with our ARR method applied. Bold figures denote the better models between base model and base model + ARR

	Amazon		Reddit	
	R@8	MRR	R@8	MRR
BOW	0.343	0.236	0.095	0.083
TF-IDF	0.239	0.149	0.140	0.110
Zero-shot	0.298	0.181	0.142	0.110
MLL	**0.918**	**0.852**	**0.253**	**0.183**
MLL + ARR	0.900	0.834	0.247	0.177
CL	**0.935**	**0.873**	**0.300**	**0.214**
CL + ARR	0.922	0.854	0.292	0.209

Table 3: Experimental results on in-distribution-topic open-set setup. "X+ARR" denotes a base model X with our ARR method applied. Bold figures denote the better models between base model and base model + ARR

In-distribution topic. Table 3 shows that ARR reveals performance penalties in in-distribution topic setup. That is, for the MLL model, our method reveals an average of 1.2% penalty in both R@8 and MRR compared to base models in Amazon and Reddit datasets. Additionally, our method applied to CL models reveals 0.25% penalty in R@8 and 1.2% in MRR for both datasets.

Discussion. Results from the cross-topic study reveal that ARR is effective in 4 out of 6 cases. Such results show the effectiveness of our method in reducing the influence of topical information. However, the method also reveals performance penalties in scenarios where topic shortcut seems beneficial, as shown in in-distribution topic experiments. This result is expected since our method reduces the usage of topical information in a text representation. Furthermore, we have also observed performance penalties in some cases from cross-topic experiments (Reddit and Amazon). We hypothesize that the resemblance between these experiments is caused by topic information leakage.

Topic information leakage. It is important to note that the Reddit and Fanfiction datasets have more topics than the Amazon dataset, i.e., 4,849

Training	Test
Captain America (Fanfiction)	**Doctor Strange (Fanfiction)**
Peggy wound up sitting with **Steve, Thor**, and **Thor**'s date, whose name Peggy did not have the opportunity to learn. [...] Tony and Pepper had elected to [...]	Strange followed **Thor** out of the room and into the meeting hall, seating him next to himself and Iron Man. "Hi, I'm **Tony**." **Tony** said, offering a hand.
Literature (Reddit)	**Poetryreading (Reddit)**
I agree, I also think there is something intriguing about the **setting** and **tone** and depravity of it all. I seem them as honest and genuine **character** portraits about someone who don't see any purpose in life, they are completely devoid of any pretension or attempt to impress us.	Goosebumps. You hit the **tone** of this perfectly. McGough is one of my favourites and the majority of his **poems** have a very light, **comedic** feel to them. This one, though, is simplistic but can be interpreted as very menacing... which you carried off well. Really nicely done, sweetheart.

Table 4: Hand-picked examples of text excerpts from topics in the Reddit and Fanfiction training and cross-topic test data that contain overlapping topical information (highlighted in bold). Fanfiction excerpts contain overlapping entity mentions, while Reddit excerpts contain words commonly used in literary analysis.

and 1,200 topics in comparison to 4 topics, respectively. Since we randomly split these topics into training, validation, and test sets, Reddit and Fanfiction are more prone to topic information leakage than Amazon. To illustrate, in the Fanfiction dataset, the topic of "Captain America" can be in the training set while "Doctor Stange" can be in the test data. For the Reddit dataset, the topic of "literature" and "poetryreading" are similar but our split method does *not* prevent them from being assigned to training and test data separately. Table 4 shows examples of texts from the overlapping topics in Reddit and Fanfiction datasets. Since these topics have overlapping information, learning a topic shortcut from the former can still benefit the latter. These leaked topics share the same named entities and concepts that diminishes the "unseen topics" aspect of the cross-topic test sets. Together, these observations suggest that it might be beneficial for future cross-topic experiments to use a train-validation-test split that considers the similarity between topics to prevent information leakage.

5 Conclusion

In conclusion, we propose authorship attribution solutions that can handle large amount of unseen authors and topics.

Firstly, we present *Authorship Representation Regularization*, a self-distillation framework that helps authorship representation to generalize toward unseen topics and authors at scale.

Secondly, we propose studies in authorship attribution with a *cross-topic-open-set* environment to assess our method. Our experimental results show that our framework can improve recall@8 and MRR over baselines in 4 out of 6 cases in cross-topic environments. However, our method's effectiveness is diminished in the in-distribution topic (or topic leaked) scenarios where models can still use topic-related features to help discriminate the text's writing styles.

In future works, it is interesting to investigate the cross-topic data split that can prevent the topic information leakage issue. Such investigation should help create a more challenging evaluation method for cross-topic authorship attribution, as well as help create an authorship attribution method that is robust toward various real-world applications.

Limitations

In this section, we describe the limitation of our studies in the terms of topic information leakage and dataset properties.

First, our data split uses the topic label acquired from each text's labeled category. However, such "topics" are not guaranteed to be distinct from each other. Therefore, there seems to be a topic information leakage in Reddit and Fanfiction datasets, as described in the discussion in Section 4.

Moreover, the datasets used in our experiments are obtained only from online texts written in English language. To the best of our knowledge, these datasets are the only sufficiently large data sources. A large size ensures that after applying our data split, the dataset still has a sufficiently large number of samples with diverse authors and topics. As a result of our limited selection of datasets, our findings might not apply to texts in other domains, such as historical or forensic writings. Furthermore, our proposed method has experimented only on English language texts, and its finding might not apply to languages with different morphosyntactic properties. For example, it is possible that our proxy for topic bias model (TF-IDF) might not be as effec-

tive on text in languages with grammatical genders, which have more morphological variations.

Acknowledgement

This study is partially supported by the Digital Economy Promotion Agency Thailand.

References

Ahmed Abbasi and Hsinchun Chen. 2005. Applying authorship analysis to extremist-group web forum messages. *IEEE Intelligent Systems*, 20(5):67–75.

Jason Baumgartner, Savvas Zannettou, Brian Keegan, Megan Squire, and Jeremy Blackburn. 2020. The pushshift reddit dataset. In *Proceedings of the international AAAI conference on web and social media*, volume 14, pages 830–839.

Janek Bevendorff, Berta Chulvi, Gretel Liz De La Peña Sarracén, Mike Kestemont, Enrique Manjavacas, Ilia Markov, Maximilian Mayerl, Martin Potthast, Francisco Rangel, Paolo Rosso, et al. 2021. Overview of pan 2021: Authorship verification, profiling hate speech spreaders on twitter, and style change detection. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 419–431. Springer.

Janek Bevendorff, Bilal Ghanem, Anastasia Giachanou, Mike Kestemont, Enrique Manjavacas, Ilia Markov, Maximilian Mayerl, Martin Potthast, Francisco Rangel, Paolo Rosso, Günther Specht, Efstathios Stamatatos, Benno Stein, Matti Wiegmann, and Eva Zangerle. 2020. Overview of pan 2020: Authorship verification, celebrity profiling, profiling fake news spreaders on twitter, and style change detection. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, pages 372–383, Cham. Springer International Publishing.

Malcolm Coulthard. 2012. On admissible linguistic evidence. *JL & Pol'y*, 21:441.

Julien Hay, Bich-Lien Doan, Fabrice Popineau, and Ouassim Ait Elhara. 2020. Representation learning of writing style. In *Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)*, pages 232–243, Online. Association for Computational Linguistics.

Patrick Juola. 2013. How a computer program helped reveal jk rowling as author of a cuckoo's calling. *Scientific American*, 20:13.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33:18661–18673.

Moshe Koppel and Shachar Seidman. 2013. Automatically identifying pseudepigraphic texts. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1449–1454, Seattle, Washington, USA. Association for Computational Linguistics.

Maarten Lambers and Cor J Veenman. 2009. Forensic authorship attribution using compression distances to prototypes. In *International Workshop on Computational Forensics*, pages 13–24. Springer.

George K Mikros and Eleni K Argiri. 2007. Investigating topic influence in authorship attribution. In *PAN*.

Jianmo Ni, Jiacheng Li, and Julian McAuley. 2019. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. 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)*, pages 188–197, Hong Kong, China. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Rafael A. Rivera-Soto, Olivia Elizabeth Miano, Juanita Ordonez, Barry Y. Chen, Aleem Khan, Marcus Bishop, and Nicholas Andrews. 2021. Learning universal authorship representations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 913–919, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Wei Song, Chen Zhao, and Lizhen Liu. 2019. Multi-task learning for authorship attribution via topic approximation and competitive attention. *IEEE Access*, 7:177114–177121.

Efstathios Stamatatos. 2017. Authorship attribution using text distortion. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1138–1149.

Justin Anthony Stover, Yaron Winter, Moshe Koppel, and Mike Kestemont. 2016. Computational authorship verification method attributes a new work to a major 2nd century a frican author. *Journal of the Association for Information Science and Technology*, 67(1):239–242.

Prasetya Ajie Utama, Nafise Sadat Moosavi, and Iryna Gurevych. 2020. Mind the trade-off: Debiasing NLU models without degrading the in-distribution performance. In *Proceedings of the 58th Annual Meeting of*

the Association for Computational Linguistics, pages 8717–8729, Online. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

A Appendix

A.1 Reproducibility.

The source code and configurations used to reproduce our experiments are available at https://www.github.com/jitkapat/TopicReg

A.2 Additional dataset information

Topic label acquisition. To allow cross-topic data split, the topics of each text must be labeled. We use the metadata that are available for each text as topics. For the Amazon dataset, we use product review categories as topics. For the Reddit dataset, we consider texts from different subreddit (subforums with specific interests) as different topics. For Fanfiction, we use the fandom label (fandom describes the original story that each fan-written fiction is based on, e.g., Harry Potter) as topics.

Train-validation-test-split parameters. For Amazon and Reddit, we use the train-test split described in Section 3. We use hand-picked percentage threshold values of authors and topics with the most samples as training candidates. We use 10% author threshold and 20% topic threshold for both Amazon and Reddit. We pick 80% of the training candidates in both datasets as the training data. The rest of each dataset is then sampled into validation data and test data as described in 3. Finally, we downsample the Reddit dataset into 10% to get a similar dataset size and training time to other datasets.

However, for Fanfiction, we use the data split introduced in PAN2021 authorship verification (Bevendorff et al., 2021), which does not include an in-distribution-topic but unseen author test data. We also use the test data from PAN2020 (Bevendorff et al., 2020) as the validation data for our Fanfiction experiments.

Text anonymity. Since all texts in every dataset we use have been collected from publicly accessible websites, we did not additionally anonymize any mention of people or organizations.

A.3 Computation details

Computing Infrastructures. We use a single Tesla A100 GPU on a single machine to train each model in all of our experiments.

Model Parameters. All deep learning baselines (CL and MLL) use the same pre-trained sBERT encoder, which has 82.1 million parameters. Although ARR training includes both base model and target model, only 82.1 million parameters of the target model are updated.

Run time. The average training time for each model in our experiments is approximately 6 hours. In total, we have trained 117 models (including model variations in learning objectives, hyperparameters, and random seed.) with a total training time of approximately 700 hours. At testing time, it took an average of 20 minutes to perform inference and evaluation on the whole test set of at most about 207,000 samples.

A.4 Hyperparameters

In our experiments, we search for the best hyperparameters for each base model and ARR-enhanced model using manual search. We choose the best model based on recall@8 evaluated on a validation set. We vary the following values as hyperparameters and random seeds.

1. learning rate = [1e−3, 1e−4, 1e−5]
2. temperature = [0.5, 0.1, 0.05, 0.01]
3. random seed = [0, 43, 314]

Additionally, we set the batch size to 64 for all models. Also, we set the number of epochs to 3 for both MLL and CL baselines and the number of epochs to 1 for additional ARR training, to avoid over-fitting.