Task Adaptive Pretraining of Transformers for Hostility Detection

Tathagata Raha(✉), Sayar Ghosh Roy, Ujwal Narayan, Zubair Abid, and Vasudeva Varma

Information Retrieval and Extraction Lab (iREL),
International Institute of Information Technology, Hyderabad, Hyderabad, India
{tathagata.raha,sayar.ghosh,ujwal.narayan,
zubair.abid}@research.iiit.ac.in, vv@iiit.ac.in

Abstract. Identifying adverse and hostile content on the web and more particularly, on social media, has become a problem of paramount interest in recent years. With their ever increasing popularity, fine-tuning of pretrained Transformer-based encoder models with a classifier head is gradually becoming the new baseline for natural language classification tasks. In our work, we explore the gains attributed to Task Adaptive Pretraining (TAPT) prior to fine-tuning of Transformer-based architectures. We specifically study two problems, namely, (a) Coarse binary classification of Hindi Tweets into Hostile or Not, and (b) Fine-grained multi-label classification of Tweets into four categories: hate, fake, offensive, and defamation. Building upon an architecture that takes emojis and segmented hashtags into consideration for classification, we are able to experimentally showcase the performance upgrades due to TAPT. Our system (with team name 'iREL IIIT') ranked first in the 'Hostile Post Detection in Hindi' shared task with an F1 score of 97.16% for coarse-grained detection and a weighted F1 score of 62.96% for fine-grained multi-label classification on the provided blind test corpora.

Keywords: Task Adaptive Pretraining (TAPT) · Hostility detection · IndicBERT

1 Introduction

With the increase in the number of active users on the internet, the amount of content available on the World Wide Web, and more specifically, that on social media has seen a sharp rise in recent years. A sizable portion of the available content contains instances of hostility thereby posing potential adverse effects upon its readers. Content that is hostile in the form of, say, a hateful comment, unwarranted usage of offensive language, attempt at defaming an individual, or a post spreading some sort of misinformation circulates faster as compared to typical textual information [12,18]. Identifying and pinpointing such instances of hostility is of the utmost importance when it comes to ensuring the sanctity

T. Chakraborty et al. (Eds.): CONSTRAINT 2021, CCIS 1402, pp. 236–243, 2021.
https://doi.org/10.1007/978-3-030-73696-5_22

of the World Wide Web and the well-being of its users and as such, multiple endeavors have been made to design systems that can automatically identify toxic content on the web [1,2,10,11,15].

In this work, we focus on the problem of identifying certain Hindi Tweets which are hostile in nature. We further analyze whether the Tweet can fit into one or more of the following buckets: hateful, offensive, defamation, and fake. The popularity of pretrained Transformer-based [17] models for tasks involving Natural Language Understanding is slowly making them the new baseline for text classification tasks. In such a scene, we experiment with the idea of Task Adaptive Pretraining [7]. IndicBERT [8], which is similar to BERT [4] but trained on large corpora of Indian Language text is our primary pretrained Transformer of choice for dealing with Hindi text.

We adopt a model architecture similar to Ghosh Roy et al., 2021 [6], which leverages information from emojis and hashtags within the Tweet in addition to the cleaned natural language text. We are able to successfully portray 1.35% and 1.40% increases for binary hostility detection and on average, 4.06% and 1.05% increases for fine-grained classifications into the four hostile classes on macro and weighted F1 metrics respectively with Task Adaptive Pretraining (TAPT) before fine-tuning our architectures for classification.

Table 1. Distribution of supervised labels in training set

Label	Frequency
Non-hostile	3050
Defamation	564
Fake	1144
Hate	792
Offensive	742

Table 2. Distribution of labels in the test set

Label	Frequency
Non-hostile	873
Defamation	169
Fake	334
Hate	234
Offensive	219

2 Dataset

The dataset for training and model development was provided by the organizers of the Constraint shared task[1] [3,14]. The data was in the form of Tweets primarily composed in the Hindi language and contained annotations for five separate fields. Firstly, a coarse-grained label for whether the post is hostile or not was available. If a Tweet was indeed hostile, it would not carry the 'not-hostile' tag. Hostile Tweets carried one or more tags indicating its class of hostility among the following four non-disjoint sets (definitions for each class were provided by the Shared Task organizers):

1. **Fake News:** A claim or information that is verified to be untrue.
2. **Hate Speech:** A post targeting a specific group of people based on their ethnicity, religious beliefs, geographical belonging, race, etc., with malicious intentions of spreading hate or encouraging violence.
3. **Offensive:** A post containing profanity, impolite, rude, or vulgar language to insult a targeted individual or group.
4. **Defamation:** A misinformation regarding an individual or group.

A collection of 5728 supervised training examples were provided which we split into training and validation sets in an 80–20 ratio, while a set of 1653 Tweets served as the blind test corpora. The mapping from a particular class to its number of training examples has been outlined in Table 1. The distribution of labels within the test set is shown in Table 2. Note that the test labels were released after the conclusion of the shared task. Throughout, a post marked as 'non-hostile' cannot have any other label while the remaining posts can theoretically have n labelings, $n \in \{1, 2, 3, 4\}$.

3 Approach

In this section, we describe our model in detail and present the foundations for our experiments. We acknowledge that the language style for social media text differs from that of formal as well as day-to-day spoken language. Thus, a model whose input is in the form of Tweets should be aware of and be able to leverage information encoded in the form of emojis and hashtags. We base our primary architecture on that of Ghosh Roy et al., 2021 [6] with a few modifications.

3.1 Preprocessing and Feature Extraction

Similar to Ghosh Roy et al., 2021 [6], the raw input text is tokenized on whitespaces plus symbols such as commas, colons, and semicolons. All emojis and hashtags are extracted into two separate stores. The cleaned Tweet text which is the primary information source for our model is free from non-textual tokens

[1] constraint-shared-task-2021.github.io.

including smileys, URLs, mentions, numbers, reserved words, hashtags, and emojis. The tweet-preprocessor[2] python library was used for categorizing tokens into the above-mentioned classes.

To generate the centralized representation of all emojis, we utilize emoji2vec [5] to generate 300 dimension vectors for each emoji and consider the arithmetic mean of all such vectors. We use the ekphrasis[3] library for hashtag segmentation. The segmented hashtags are arranged in a sequential manner separated by whitespaces and this serves as the composite hashtag or 'hashtag flow' feature. Thus, we leverage a set of three features, namely, (a) the cleaned textual information, (b) the collective hashtag flow information, and (c) the centralized emoji embedding.

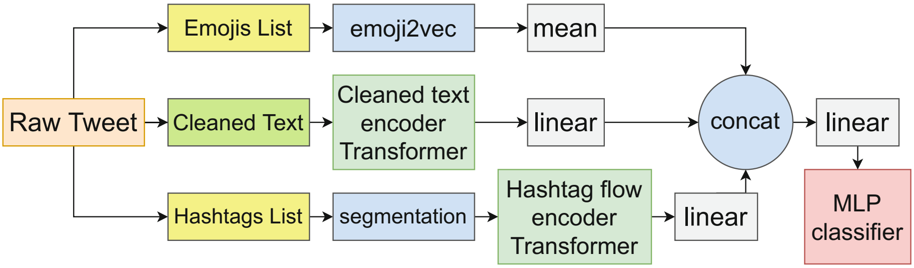

Fig. 1. Model architecture

3.2 Architecture

In this subsection, we outline the overall flow of information pieces from the set of input features to label generation. We leverage two Transformer models to generate embeddings of dimension size 768 for each of the cleaned text and the hashtag flow features. The two Transformer-based embeddings are passed through two linear layers to yield the final vector representations for cleaned text and hashtag collection. The three vectors: cleaned text, composite hashtag, and centralized emoji representations are then concatenated and passed through a linear layer to form the final 1836-dimension vector used for classification. A dense multi-layer perceptron serves as the final binary classifier head. The overall information flow is presented in Fig. 1. For the multi-label classification task, we trained our architecture individually to yield four separate binary classification models. In all cases, we performed an end-to-end training on the available training data based on cross-entropy loss.

[2] github.com/s/preprocessor.
[3] github.com/cbaziotis/ekphrasis.

3.3 Task Adaptive Pretraining

We turn to Gururangan et al., 2020 [7], which showcases the boons of continued pretraining of Transformer models on natural language data specific to certain domains (Domain Adaptive Pretraining) and on the consolidated unlabelled task-specific data (Task Adaptive Pretraining). Their findings highlighted the benefits of Task Adaptive Pretraining (TAPT) of already pretrained Transformer models such as BERT on downstream tasks like text classification. We experimented with the same approach for our task of hostility detection in Hindi having IndicBERT as our base Transformer model. Our results (in Sect. 4) clearly showcases the gains attributed to this further pre-training with the masked language modeling (MLM) objective. Note that only the cleaned text encoder Transformer model is the one undergoing TAPT. The hashtag sequence encoder Transformer is initialized to pretrained IndicBERT weights. We create a body of text using all of the available training samples and in that, we add each sample twice: firstly, we consider it as is i.e. the raw Tweet is utilized, and secondly, we add the cleaned Tweet text. A pretrained IndicBERT Transformer is further pretrained upon this body of text with the MLM objective and we use these Transformer weights for our cleaned text encoder before fine-tuning our architecture on the training samples.

Table 3. Results on the validation split for every category (% weighted F1 scores)

Metric	Without TAPT	With TAPT	Gains
Hostility (coarse)	96.87	98.27	1.40
Defamation	86.47	86.31	−0.16
Fake	89.53	90.99	1.46
Hate	85.69	87.06	1.37
Offensive	87.12	88.66	1.54

Table 4. Results on the validation split for every category (% macro F1 scores)

Metric	Without TAPT	With TAPT	Gains
Hostility (coarse)	96.84	98.19	1.35
Defamation	59.43	63.38	3.95
Fake	83.69	86.52	2.83
Hate	70.77	74.20	3.43
Offensive	68.72	74.73	6.01

Table 5. Shared task results: top 3 teams on public leaderboard (% F1 scores)

Metric	iREL IIIT (Us)	Albatross	Quark
Hostility (coarse	97.16	97.10	96.91
Defamation	44.65	42.80	30.61
Fake	77.18	81.40	79.15
Hate	59.78	49.69	42.82
Offensive	58.80	56.49	56.99
Weighted (fine)	62.96	61.11	56.60

4 Results

In Tables 3 and 4, we present metrics computed on our validation set. We observe 1.35% and 1.40% increases in the macro and weighted F1 scores for binary hostility detection and on average, 4.06% and 1.05% increases in macro and weighted F1 values for fine-grained classifications into the four hostile classes. In all classes (except for 'Defamation' where a 0.16% performance drop is seen for the Weighted F1 metric), the classifier performance is enhanced upon introducing the Task Adaptive Pretraining. In Table 5, we present our official results with team name 'iREL IIIT' on the blind test corpora and compare it to the first and second runner-ups of the shared task.

5 Experimental Details

We used AI4Bharat's official release of IndicBERT[4] as part of Hugging Face's[5] Transformers library. All of our experimentation code was written using PyTorch[6] [13]. We considered maximum input sequence length of 128 for both of our Transformer models, namely, the cleaned text encoder and the hashtag flow encoder. Transformer weights of both of these encoders were jointly tuned during the fine-tuning phase. We used AllenAI's implementation[7] of Task Adaptive Pretraining based on the Masked Language Modeling objective. The continued pretraining of IndicBERT upon the curated task-specific text was performed for 100 epochs with other hyperparameters set to their default values. The cleaned text encoder was initialized with these Transformer weights before the fine-tuning phase.

For fine-tuning our end-to-end architecture, we used Adam [9] optimizer with a learning rate of 1e−5 and a dropout [16] probability value of 0.1. All other hyperparameters were set to their default values and the fine-tuning was continued for 10 epochs. We saved model weights at the ends of each epoch and

[4] github.com/AI4Bharat/indic-bert.

[5] huggingface.co/.

[6] pytorch.org/.

[7] github.com/allenai/dont-stop-pretraining.

utilized the set of weights yielding the best macro F1 score on the validation set. The same schema of training and model weight saving was adopted for the coarse binary hostility detector as well as the four binary classification models for hate, defamation, offensive, and fake posts.

6 Conclusion

In this paper, we have presented a state-of-the-art hostility detection system for Hindi Tweets. Our model architecture utilizing IndicBERT as the base Transformer, which is aware of features relevant to social media style of text in addition to the cleaned textual information is capable of both identifying hostility within Tweets and performing a fine-grained multi-label classification to place them into the buckets of hateful, defamation, offensive, and fake. Our studies proved the efficacy of performing Task Adaptive Pretraining (TAPT) of Transformers before using such models as components of a to-be fine-tuned architecture. We experimentally showed 1.35% and 1.40% gains for coarse hostility detection and average gains of 4.06% and 1.05% for the four types of binary classifications, on macro and weighted F1 score metrics respectively in both cases. Our system ranked first in the 'Hostile Post Detection in Hindi' shared task with an F1 score of 97.16% for coarse-grained detection and a weighted F1 score of 62.96% for fine-grained classification on the provided blind test corpora.

References

1. Badjatiya, P., Gupta, M., Varma, V.: Stereotypical bias removal for hate speech detection task using knowledge-based generalizations. In: The World Wide Web Conference, WWW 2019, pp. 49–59. Association for Computing Machinery, New York (2019). https://doi.org/10.1145/3308558.3313504
2. Badjatiya, P., Gupta, S., Gupta, M., Varma, V.: Deep learning for hate speech detection in tweets. In: Proceedings of the 26th International Conference on World Wide Web Companion, WWW 2017 Companion, pp. 759–760. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva (2017). https://doi.org/10.1145/3041021.3054223
3. Bhardwaj, M., Akhtar, M.S., Ekbal, A., Das, A., Chakraborty, T.: Hostility detection dataset in Hindi (2020)
4. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
5. Eisner, B., Rocktäschel, T., Augenstein, I., Bosnjak, M., Riedel, S.: emoji2vec: learning Emoji representations from their description. CoRR abs/1609.08359 (2016). http://arxiv.org/abs/1609.08359
6. Ghosh Roy, S., Narayan, U., Raha, T., Abid, Z., Varma, V.: Leveraging multilingual transformers for hate speech detection. In: Working Notes of FIRE 2020 - Forum for Information Retrieval Evaluation. CEUR (2021)
7. Gururangan, S., et al.: Don't stop pretraining: adapt language models to domains and tasks. arXiv preprint arXiv:2004.10964 (2020)

8. Kakwani, D., et al.: IndicNLPSuite: monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for Indian languages. In: Findings of EMNLP (2020)
9. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization (2017)
10. Kumar, R., Ojha, A.K., Zampieri, M., Malmasi, S. (eds.): Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018). Association for Computational Linguistics, Santa Fe, August 2018. https://www.aclweb.org/anthology/W18-4400
11. Mandl, T., et al.: Overview of the HASOC track at FIRE 2020: hate speech and offensive content identification in Indo-European languages). In: Working Notes of FIRE 2020 - Forum for Information Retrieval Evaluation. CEUR, December 2020
12. Mathew, B., Dutt, R., Goyal, P., Mukherjee, A.: Spread of hate speech in online social media. In: Proceedings of the 10th ACM Conference on Web Science, pp. 173–182, June 2019. https://doi.org/10.1145/3292522.3326034
13. Paszke, A., et al.: PyTorch: an imperative style, high-performance deep learning library (2019)
14. Patwa, P., et al.: Overview of CONSTRAINT 2021 shared tasks: detecting English COVID-19 fake news and Hindi hostile posts. In: Chakraborty, T., et al. (eds.) CONSTRAINT 2021. CCIS, vol. 1402, pp. 42–53. Springer, Cham (2021)
15. Pinnaparaju, N., Indurthi, V., Varma, V.: Identifying fake news spreaders in social media. In: Cappellato, L., Eickhoff, C., Ferro, N., Névéol, A. (eds.) CLEF 2020 Labs and Workshops, Notebook Papers. CEUR-WS.org, September 2020
16. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. **15**(56), 1929–1958 (2014). http://jmlr.org/papers/v15/srivastava14a.html
17. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems, pp. 5998–6008 (2017)
18. Vosoughi, S., Roy, D., Aral, S.: The spread of true and false news online. Science **359**(6380), 1146–1151 (2018). https://doi.org/10.1126/science.aap9559. https://science.sciencemag.org/content/359/6380/1146