HostileNet: Multilabel Hostile Post Detection in Hindi

Mohit Bhardwaj, Megha Sundriyal, Manjot Bedi, Md Shad Akhtar, and Tanmoy Chakraborty, *Senior Member, IEEE*

***Abstract*— In this article, we deal with the task of hostile post detection in Hindi. The objective is to predict whether a social media post is hostile or not. Furthermore, if the post is hostile, we identify one or more fine-grained hostile dimensions out of the following four—*fake*, *hate*, *offensive*, and *defamation*. We propose HostileNet, a novel deep-learning framework that leverages HindiBERT-based contextual representations and hand-crafted features like lexicon, emoticon, and hashtag embeddings for hostile post classification. Moreover, we also propose a novel mechanism to fine-tune HindiBERT's attention vectors with respect to each hostile dimension. We evaluate HostileNet on the CONSTRAINT-2021 shared task dataset on hostile post detection in Hindi for both coarse-grained (hostile versus nonhostile) and fine-grained (*fake* versus *hate* versus *offensive* versus *defamation*) setups. HostileNet outperforms the best-performing system as reported in the CONSTRAINT-2021 shared task for both the setups. Furthermore, we provide a thorough analysis of the obtained results in the form of an ablation study, error analysis, attention heatmap analysis, lexicon feature analysis, and so on. We also perform in-the-wild evaluation and conduct a user survey to assess the robustness of our proposed model. We make the code and the curated multilabel hostile lexicon available for research use at https://github.com/LCS2-IIITD/HostileNet.**

***Index Terms*— Fake news, hate speech, Hindi, hostility detection, online social media, supervised learning.**

I. Introduction

THE growth of the Internet has significantly increased the use of online social media platforms as a stage for people to impart their thoughts and opinions. The Internet disseminates a massive amount of textual information on a variety of topics such as political issues, religious groups, economy, and so on. The major intentions behind hostile content (e.g., fake news, hate speech, offensive posts, etc.) are to spread false information, embed fear into the minds of the public, defame someone, or spread hatred [1]. There are several instances where the spread of hostile content has impacted the entire society. For example, during the 45th U.S. Presidential elections, around 25% of Americans visited a fake news website that tried to influence the thought process of the general public and affected the eventual outcome of the election [2]. A fake and defaming post in Bangladesh caused the destruction of several religious places of minority communities by a violent mob [3]. Considering the impact of such hostile posts, their timely detection and remedy are of utmost necessity to ensure a civilized environment.

For hostile post detection, a decent number of studies have been carried out for English and other high-resource languages [4], [5], [6], [7], [8], [9]; however, the research involving Indian languages (e.g., Hindi, Tamil, Bengali, etc.) is comparatively less explored [10], [11]. A prime reason is the unavailability of high-quality datasets. Recently, a benchmark dataset in Hindi covering four hostile dimensions was developed as part of a shared task in the CONSTRAINT-2021 [12]. Instead of a single unified strategy for all four fine-grained hostile dimensions, namely *fake*, *hate*, *offensive*, and *defamation*, existing systems [13], [14], [15], [16], [17] use binary relevance or majority voting as an ensemble technique. This prompts us to explore a unified solution for hostile post detection instead of dimension-specific solutions. To this end, in this article, we propose a novel joint architecture, HostileNet, to detect all four hostile dimensions (i.e., *fake*, *offensive*, *hate*, and *defamation*) in Hindi simultaneously. Following Bhardwaj et al. [18], we define all four hostile dimensions. A piece of information or an alleged claim that can be proven to be false is referred to as fake news. An offensive post contains profane, impolite, or rude language intended to offend any individual or group, whereas hate speech is directed at a specific individual or group of people based on their ethnicity, religious beliefs, race, or other factors, with the malicious intent of spreading hatred. Finally, defamation tends to spread false information about an individual, a group, or an institution with the goal of harming their public reputation.

A. Challenges

A hostile post is often published to disseminate misinformation, hatred, and mislead the general public [1]. As a result, it requires deep insights into interpreting the hostility even for a human being. Another crucial challenge is the regional and cultural differences that affect how a group interprets a post. For example, in India, China, and many other countries, the word "*monkey*" is not treated as a derogatory and offensive

Manuscript received 9 September 2022; revised 14 December 2022; accepted 6 February 2023. The work of Tanmoy Chakraborty was supported by the Science and Engineering Research Board (SERB), India, through the Ramanujan Fellowship under Grant SB/S2/RJN-073/2017. *(Mohit Bhardwaj and Megha Sundriyal contributed equally to this work.) (Corresponding author: Megha Sundriyal.)*

Mohit Bhardwaj, Megha Sundriyal, Manjot Bedi, and Md Shad Akhtar are with the Department of Computer Science and Engineering, IIIT Delhi, New Delhi 110020, India (e-mail: mohit19014@iiitd.ac.in; meghas@iiitd.ac.in; manjotb@iiitd.ac.in; shad.akhtar@iiitd.ac.in).

Tanmoy Chakraborty is with IIT Delhi, New Delhi 110016, India (e-mail: tanchak@iitd.ac.in).

This article has supplementary downloadable material available at https://doi.org/10.1109/TCSS.2023.3244014, provided by the authors.

Digital Object Identifier 10.1109/TCSS.2023.3244014

term, instead, monkeys are worshipped in some parts of India as a daily ritual.[1] However, it symbolizes a racist comment in the majority of the western world.[2] Furthermore, fine-grained hostile post detection adds significant complexity to the identification process due to the eminent, diverse, yet overlapping characteristics of these subcategories [18]. The distinction between hate speech and offensive speech lies in the motivation and severity of harmfulness behind the hostility—hateful posts are more severe as well as targeted toward specific groups or communities while offensive comments are less severe and may not contain harmful words usually. Similarly, fake news is often false and malicious, whereas an allegation in defamation might be true but lacks proof and does not always hold legal liability.[3] Given these discussions, it is not difficult to understand that the discrimination among these hostile dimensions is a highly challenging task and needs careful investigation in an efficient identification model.

B. Our Contributions

We summarize our contributions as follows.

1) We address the problem of hostile post detection in social media posts in Hindi. We explore two setups: coarse-grained hostile post detection as a binary classification and fine-grained hostile post detection as a multilabel classification problem.
2) We propose a unified framework, `HostileNet`, to handle the identification of four hostile dimensions—*fake*, *hate*, *offensive*, and *defamation*. `HostileNet` also incorporates a novel supervised attention-tuning module to optimize the computed attention scores against each hostile dimension.
3) Our evaluation on the CONSTRAINT-2021's [18] dataset shows state-of-the-art performance for both fine-grained and coarse-grained hostile post detection setups.
4) We report extensive analyses of `HostileNet`, including ablation analysis, heatmap analysis, error analysis, in-the-wild analysis, and so on.

The rest of the article is organized as follows. In Section II, we discuss the prominent work done in the field of hostile post detection. Section III encloses a brief description of the dataset. In Section IV, we shed light upon our proposed methodology. Section V consists of results and analyses of our proposed model, which is briefly followed by a conclusion in Section VII.

II. Related Work

The pioneering work in hostile text detection in English was put forward by Spertus [4]. This study leveraged the traditional machine-learning technique, namely Decision Tree, to detect hostile messages. Despite their straightforward approach to hostile post detection, their work drew a lot of attention and laid a foundation for this challenging task. Later, a supervised learning approach with unigrams to detect racism in tweets was proposed by Kwok and Wang [5]. They employed a Naive Bayes classifier, leveraging acquired labeled data from different Twitter accounts to learn a binary classifier for the labels—"racist" and "nonracist." Most of the early attempts in hostile post detection were based on traditional machine-learning methods focusing on predictive features, while in recent times, the study shifted toward the utilization of linguistic and syntactic features. Waseem and Hovy [6] utilized character n-grams coupled with linguistic features for hate speech detection. Davidson et al. [7] used support vector machines (SVM) for multiclass classification of a tweet into "hate," "offensive," or "neither," employing term frequency-inverse document frequency (tf-idf) weighted n-grams and part-of-speech (POS) tags. Surface-level features such as character n-grams, word n-grams, and word skip-grams were broadly used for hostile post detection [5], [6], [7]. To scrutinize the role of surface-level features, Malmasi and Zampieri [19] carried out a study and argued that the surface-level features are insufficient to distinguish hate speech from profanity.

In recent times, numerous studies attempted to use deep-learning methods for online hostile post detection. Badjatiya et al. [8] were the pioneer to employ deep learning to classify a tweet as "racist," "sexist," or "neither." They used LSTM to learn tweet embeddings with gradient boosting. Upon the same task, Sajjad et al. [20] practiced CNNs trained over GloVe embeddings, alongside other handcrafted features with logistic regression. Zampieri et al. [21] spawned the Offensive Language Identification Dataset (OLID), which comprised 14k English tweets for OffensEval 2019 shared task to detect and categorize the target of offensive language. Recently, BERT [22] has gained tremendous attention. Tran et al. [9] proposed a HaBERTor model for detecting hate speech where they pretrained BERT purely using 1.4M annotated hate speech comments. Parikh et al. [23] were the first to work on multilabel detection of accounts of sexism. They used models like BERT, Universal Sentence Encoder for sentence representation, and proposed a hierarchical combination of BiLSTMs and CNNs over word embeddings.

A. Hostile Text Detection in Hindi

Most literature on hostile post detection is concentrated on high-resource languages; consequently, only a few hostile post detection methods are available for Hindi. Mathur et al. [24] utilized a multichannel CNN-LSTM-based architecture to classify offensive tweets in Hinglish (Hindi + English) language. Likewise, Sengupta et al. [25] examined the relationship among five offense traits, namely aggression, hate, sarcasm, humor, and stance in Hinglish (Hindi + English) social media code-mixed texts. Kar et al. [11] utilized mBERT embeddings with Twitter user-level features for COVID-19-related fake news detection in Hindi and Bangla alongside English. They showed high efficacy in zero-shot learning among Hindi and Bengali due to their linguistic similarity as both are derived from the Indo-Aryan family of Indian languages. Currently, CONSTRAINT-2021 [12] and HASOC [10] are marked as the

[1] https://www.thehindu.com/news/national/kerala/monkeys-feast-in-this-daily-ritual/article25364092.ece

[2] https://en.wikipedia.org/wiki/Monkey_chanting

[3] As per section 499 of the Indian Penal Code (https://indiankanoon.org/doc/1041742/) if the allegations against someone are true and for the public good, then it is not considered defamation.

TABLE I
CONSTRAINT-2021 [18] HINDI SHARED TASK DATASET STATISTICS. FAKE, HATE, OFFENSIVE, AND DEFAMATION DENOTE THE NUMBER OF POSTS ASSOCIATED WITH THESE RESPECTIVE FINE-GRAINED HOSTILE DIMENSIONS. SINCE THIS IS A MULTILABEL HOSTILE DATASET, * INDICATES THE TOTAL COUNT FOR HOSTILE POSTS

Dataset	Hostile Posts					Non-Hostile
	Fake	Hate	Offensive	Defamation	Total*	
Train	1144	792	742	564	2678	3050
Val	160	110	103	77	376	435
Test	334	234	219	169	780	873
Overall	1638	1136	1064	810	3834	4358

most prominent shared tasks for hostile text detection in Hindi and grabbed a lot of attention from numerous researchers across the globe. Recently, Bhatnagar et al. [26] extended their previous work in CONSTRAINT-2021 and investigated the effects of data augmentation and various transformer representations for hostile post detection in Hindi.

B. Shortcomings of the Existing Systems

The main shortcomings of existing systems are as follows.

1) *Low-resource languages:* The most significant deficiency in the domain of hostile post detection is the lack of research in low-resource languages.
2) *Hostile post detection in single dimension only:* Even in high-resource languages, most existing systems work in only one dimension of hostility [4], [5], [6], [7]. Hence, we do not have unified systems that can efficiently classify various subcategories of hostile content such as fake news, hate speech, offensive, and defaming posts.
3) *Less explainability:* Even if some multilabel systems tackle a subset of hostile categories, these models have significantly less reliability and explainability as to why a post was predicted as, say, hateful and defaming [9], [23].

III. DATASET

We use the CONSTRAINT-2021 [18] hostile post detection dataset for the evaluation of HostileNet. The dataset comprises 8192 posts in Hindi collected from different online social media platforms. The dataset has two major categories—hostile and nonhostile, with close to even distribution at 47:53 ratio, respectively. The hostile posts are further categorized into one or more of the four fine-grained labels—*fake*, *hate*, *offensive*, and *defamation*. For completeness, we provide the descriptions of these dimensions in the Supplementary material.

We present brief statistics of the dataset in Table I. We observe that the hostile posts are not perfectly balanced across four dimensions—*defamation* class has less than 50% samples in comparison to the *fake* class. Dataset analysis shows that on average a nonhostile post has roughly 32% more punctuation marks than a hostile post, which suggests that people who spread hostile content bother less about the syntactic correctness of their content and more about the harmful aspect. We also observe that offensive posts have one user mention on average, reflecting that the dataset mainly consists of directed offensive content. For experiments, we follow CONSTRAINT-2021 Hindi shared task's train, validation, and test split ratio of 70:10:20, respectively.

IV. PROPOSED METHODOLOGY

A high-level architecture of our proposed model, HostileNet, is shown in Fig. 1. It has three main components—a network consisting of the HindiBERT [27] framework, a module to optimize the four attention heads of HindiBERT (one attention head per hostile dimension—*defamation*, *fake*, *hate*, and *offensive*), and a module to incorporate hand-crafted features such as lexicon, emoticon, and hashtag embeddings. First, we pretrain HindiBERT with CONSTRAINT-2021 hostile post detection dataset. Moreover, during preprocessing, we compute a multilabel lexicon using our Algorithm 1 and create four gold attention vectors for each sample (one for each hostile dimension—*defamation*, *fake*, *hate*, and *offensive*). The gold attention vector of a post for a hostile dimension (say, *offensive*) is a list of normalized scores that sum up to 1, such that the scores signify the importance/weightage of each token in the post with respect to the hostile dimension (*offensive*). During training, to fine-tune HindiBERT, we compute four attention vectors using four attention heads of HindiBERT and optimize the Kullback–Leibler (KL) divergence score between the computed attention vectors and the gold attention vectors. The objective is to learn the relevant and important tokens as close as to the training distribution. In parallel, we encode lexicon-specific features and fuse them into the network through concatenation. Finally, we employ a small multilayer perceptron network for classification. Since one post can belong to more than one hostile dimension, we utilize four sigmoid neurons at the output layer and optimize the classification loss through binary cross-entropy (BCE).

Formally, let $p = \{w_1, w_2, \ldots, w_n\}$ be a post in the dataset consisting of n words. At first, we normalize the text. For this, we replace each emoticon in the post with its corresponding textual definition, for example, we convert 🙏 to "*folded_hands*.[4]" Then, we tokenize the post using sentence piece tokenizer and subsequently pad the sequence up to T length for consistency among all posts.

A. Preprocessing

In this section, we describe the compilation of a multilabel lexicon for hostile texts. We use the lexicon for leveraging the hand-crafted features in HostileNet and also to obtain the gold attention vector for each hostile dimension.

1) *Multilabel lexicon algorithm:* We summarize the lexicon creation process in Algorithm 1. Given the set of all training posts as the input, the algorithm returns a multilabel lexicon dictionary,[5] where the key can be any token

[4]https://pypi.org/project/emoji/

[5]A dictionary is a general-purpose data structure for storing a group of objects in the form of key-value pairs.

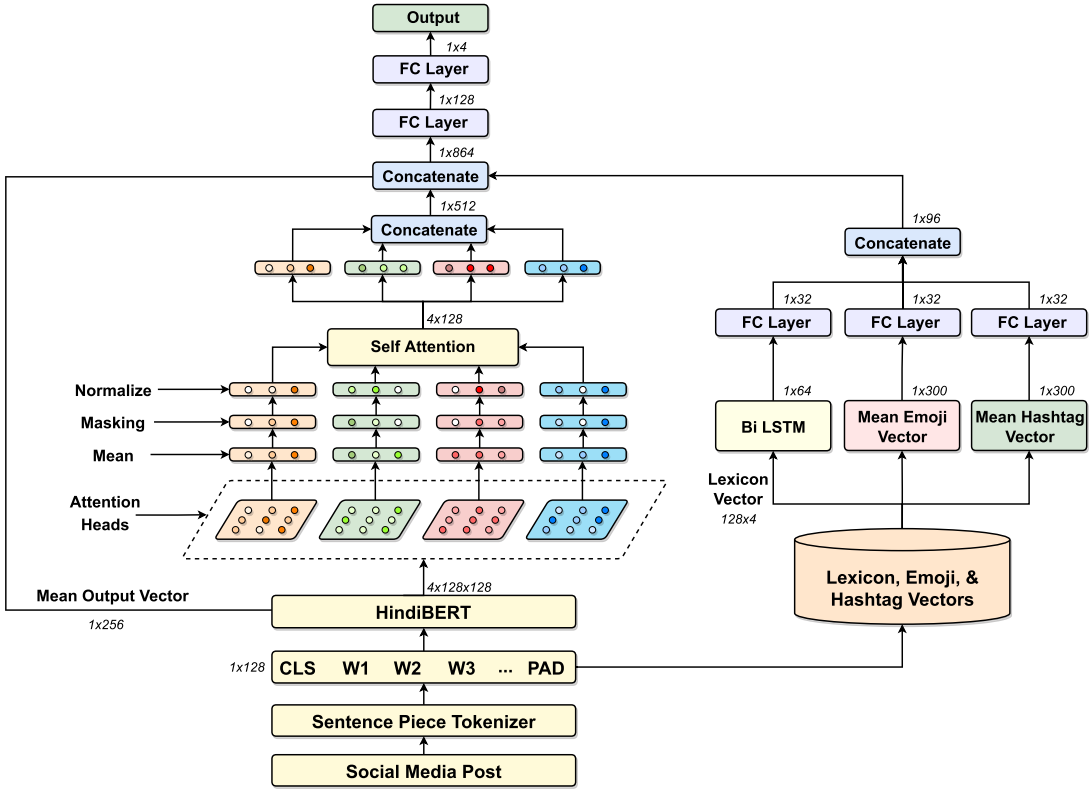

Fig. 1. Proposed architecture of `HostileNet` for the multilabel hostile post detection in Hindi.

from the dataset and its value consists of a list of five normalized scores that sum up to unity. These five scores show the token's association within *defamation*, *fake*, *hate*, *offensive*, and *nonhostile* dimensions, respectively. The lth dimension's lexicon score for a token lies in the interval $[0, 1]$, where a score close to 1 denotes a strong association of the token toward the corresponding dimension (l) and vice versa. To begin with, we calculate the token frequency f_w^l for each token w in vocabulary V, against each dimension l. The token frequency is then normalized by the total number of instances of w corresponding to distinct dimensions in (1)

$$f_w^l = \frac{f_w^l}{\sum_{j \in L} f_w^j} \tag{1}$$

$$f_w^l = \frac{f_w^l}{C^l}. \tag{2}$$

This allows us to minimize the association of frequent and common words in hostile dimensions. Furthermore, we normalize the scores for each dimension l, by the total number of training samples corresponding to lth dimension-C^l [see (2)]. This allows us to handle the skewness in the dataset, for example, a token may have a higher frequency for a dimension than others due to an imbalanced dataset. To ensure good segregation among dimensions, we subtract the normalized score from the cumulative score of all other dimensions. Finally, we compute the softmax function to obtain the probability distributions for all the tokens in the interval $[0, 1]$ as shown in (3). We assign zero lexicon scores to nonrelevant tokens such as [CLS], [SEP], and [PAD]. We provide a detailed justification for each equation from our algorithm in the Supplementary material

$$\text{Lex}_w^l = \text{Softmax}\left(f_w^l - \sum_{k \in L, k \neq l} f_w^k\right) \quad \forall l \in L. \tag{3}$$

2) *Gold attention vectors:* We took inspiration from Zou et al. [28] for the creation of gold attention vectors and lexicon-based supervised attention tuning. For a preprocessed, tokenized, and padded post p of length T, we create a gold attention vector g^l for hostile dimension l using the multilabel lexicon mentioned in Algorithm 1. To create a gold attention vector g^l, we first extract the lth dimension's lexicon score for each token in p and form a vector $v \in \mathbb{R}^T$, where T is the length of the input post after preprocessing, tokenization, and padding. Next, we compute the gold attention vector $g^l \in \mathbb{R}^T$ by applying a softmax function over vector v. We can think of g^l as a gold attention vector for a post p against dimension l because g^l gives us the probability distribution signifying the importance/weightage of each token in post p with respect to the hostile dimension l as observed from the training set.

B. Context-Rich Representation

To obtain the hidden context representation for each token $w_i \in p$, we employ a pretrained HindiBERT [27] model.

Algorithm 1 Multilabel Lexicon Algorithm

Input: The set of all posts P, in the training set.
Output: A multi-label lexicon score dictionary Lex, where a key can be any token from the dataset, and its value consists of a list of five normalized scores, which shows the token's association within *defamation*, *fake*, *hate*, *offensive*, and *nonhostile* dimensions respectively.

$L \leftarrow$ Total number of dimensions/labels in the training set
$Lex \leftarrow$ Empty dictionary
$C \leftarrow \{C^1, C^2, \ldots, C^L\}$ ▷ *Number of training samples for each dimension*
for *each token* w *in the training corpus* P **do**
  Calculate f_w^l - frequency of w in each dimension $l \in L$
end
for *each token* $w \in Lex$ **do**
  $f_w^l = \frac{f_w^l}{\sum_{j \in L} f_w^j}$
  $f_w^l = \frac{f_w^l}{C^l}$
  $Lex_w^l = Softmax(f_w^l - \sum_{k \in L, k \neq l} f_w^k)$
end

It incorporates the Electra [29] architecture and has been pretrained on 8 GB of the OSCAR common crawl dataset and 1 GB of the Wikipedia dataset in Hindi. We extract a 256-D embedding vector to represent the post as the mean of T tokens.

C. Fine-Tuning Attention Vectors

To learn the relevant and important tokens as close as to the training distribution, we fine-tune four attention heads of HindiBERT, one for each hostile dimension in HostileNet. We hypothesize that the association of one attention head per label will help the model cater specifically to each label. In transformer-based architectures like BERT, we have multiple attention heads. Each attention head is used to learn a different set of weight matrices for queries, keys, and values from the same input that allow them to learn different aspects of the same input sentence. Since our task involves learning different hostile dimensions, we hypothesize that if we could tune each attention head with respect to each hostile dimension, the model will be able to learn different aspects of hostility from the same input sentence. We utilize four attention heads of HindiBERT,[6] where each attention head corresponds to one hostile dimension.

Let $A = \{A^1, A^2, \ldots, A^L\}$ be the set of query-key attention matrices of the last layer of HindiBERT for post p. Let $A^l \in A$ represent the lth query-key attention matrix for post p. A^l is a 2-D matrix of $T \times T$ dimension, where the indices represent the tokens in p and T is the length of a post after preprocessing, tokenization, and padding. Values in each row represent the attention score for a token against itself and all other tokens in p. We take the row-wise mean of A^l to obtain an attention vector $a^l \in \mathbb{R}^T$, which represents the average attention received by each token in the input with respect to the attention head A^l. Subsequently, we mask the attention scores of various nonrelevant tokens, such as [CLS], [SEP], [PAD], and so on, to obtain the masked attention vector $m^l \in \mathbb{R}^T$. Furthermore, we normalize m^l to obtain $n^l \in \mathbb{R}^T$, by applying a masked softmax function. It allows us to redistribute the probability mass to the remaining tokens such that $\sum_t n_t^l = 1$ and still maintains 0 as the attention score for all masked tokens.

Finally, we optimize the label-wise KL divergence loss between gold attention vector g^l and normalized HindiBERT attention scores n^l for every label l in a post s as shown in the following equation:

$$\mathbb{L}_{\mathrm{KD}}^l\left(g^l \| n^l\right) = \left\{k_1^l, k_2^l, \ldots, k_T^l\right\} \tag{4}$$

where $k_t^l = n_t^l(\log(n_t^l) - g_t^l)$. Moreover, for L labels, we have a total KL divergence loss as shown in the following equation:

$$\mathbb{L}_{\mathrm{KD}}(g \| n) = \sum_{l=1}^{L} \lambda_{\mathrm{KD}}^l * \mathbb{L}_{\mathrm{KD}}^l\left(g^l \| n^l\right) \tag{5}$$

where λ_{KD}^l is a hyperparameter to control label imbalance issues in KL divergence. This allows the model to tune each head pertinent to the respective hostile dimension.

D. Hand-Crafted Features

To supplement neural network-based contextual representation, we incorporate multilabel lexicon vectors computed through our Algorithm 1 and encode them through a BiLSTM layer. Following Guibon et al. [30], we also encode emoticons present in the input post to leverage their semantics in HostileNet. In a study, Wang et al. [31] showed that hashtags play a crucial role in influencing information virality and social movements. Thus, utilizing hashtags information in identifying hostile posts can be helpful [7]. We combine these three vectors with the self-attended vectors of HostileNet for the final classification.

1) *Lexicon embedding:* To create a context-aware lexicon embedding using our multilabel lexicon, we pass the lexicon scores for all tokens in the input post through a BiLSTM to get a set of hidden state vectors $h = \{h_1, h_2, \ldots, h_T\}$. We take the sum over all the hidden states h_t, $t \in [1, T]$, and pass it through a fully-connected layer to obtain the lexicon embedding for the post.
2) *Emoticon embedding:* We take the mean of the 300-D vector representations of all the emojis present in the input post using emoji2vec [32] and pass it through a fully-connected layer.
3) *Hashtag embedding:* To incorporate hashtag information, we use Twitter's hashtag segmenter [33] to segment hashtags in the input post. We then use multilingual IndicFT [34] word embedding model to obtain 300-D static representation for each segment of the hashtag. Finally, we take the mean of all segments obtained from all the hashtags in the post and pass them through a

[6]Note that in the case of more hostile dimensions, we can train HindiBERT with more attention heads.

fully-connected layer to get the overall hashtag embedding for a post.

E. Final Prediction

Subsequent to the creation of normalized attention vectors for each hostile dimension from HindiBERT, we fuse them through a self-attention mechanism followed by a concatenation operation. The concatenated vector along with the hand-crafted features are fed to a multilayered perceptron for the final classification. As mentioned earlier, the samples in the CONSTRAINT-2021 dataset are of multilabel nature; therefore, we employ four sigmoid neurons with the BCE loss for the predictions. For optimizing `HostileNet`, we sum up BCE and KL divergence attention losses

$$\mathbb{L}(s) = \text{BCE}(s) + \sum_{l=1}^{L} \lambda^l * \mathbb{L}_{\text{KD}}^l \left(g^l \| n^l \right). \tag{6}$$

V. Experiments and Results

In this section, we discuss our experimental results and report a comparative analysis against various baselines. We also illustrate thorough analyses of `HostileNet`'s performance using ablation and different choices of supervised attention losses for our model. We then demonstrate the explainability and error analysis of our best model.

A. Baseline Models

Here, we define various existing systems that we employ for the comparative study. All these systems were part of the CONSTRAINT-2021 shared task challenge and ranked amongst the best systems: 1) **Albatross** [13]: the authors first fine-tune BERT to classify hostile and nonhostile posts. Subsequently, another classifier is trained for each dimension; 2) **Bestfit AI** [14]: the authors use Relational Graph Convolutional Networks (RGCNs) and multilingual BERT's pooled output to capture the semantic and contextual knowledge, respectively; 3) **Monolith** [15]: the authors utilize IndicBERT to train a binary coarse-grained classifier for hostile post detection and four separate classifiers for the fine-grained classification; 4) **IREL IIIT-H** [16]: the authors utilize pretrained IndicBERT and further fine-tune IndicBERT using AllenAI's pretraining implementation[7]; and 5) **Zeus** [17]: the authors fine-tune five BERT classifiers and apply a majority voting-based ensemble for the final predictions.

B. Experimental Setup

We pad each tokenized post to a maximum length of 128 tokens. We fine-tune `HostileNet` on the validation set by varying mutliple hyperparameters—*dropout* [0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5], *learning rate* [0.1, 0.01, 0.001, 0.0001, 0.00001], and *batch size* [4, 8, 16, 32]. Finally, we choose the optimized configuration as *dropout* = 0.25, *learning rate* = 0.001, and *batch size* = 16 for all experiments. We train the model for a maximum 50 *epochs* with early stopping criteria having *patience* = 20. We employ *Adam* as the optimizer with a decay of 0.001 and linear scheduler with a warm-up. For the coarse-grained classification, we optimize BCE with the *hostile* class_weight as 1.13. Similarly, in fine-grained classification, the class_weights are taken as 4.74, 2.34, 3.38, and 3.64 for the *defamation*, *fake*, *hate*, and *offensive* classes, respectively. In both cases, class_weight is calculated using $k/|l|$, where $|l|$ is the number of samples for the label l and k is the total number of samples in our training set.

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

C. Performance of `HostileNet`

We present our comprehensive result in Table II for both setups—coarse-grained and fine-grained tasks. In coarse-grained task, IREL IIIT-H [16] reports the best weighted F1-score of 97.16 in the CONSTRAINT-2021 shared task closely followed by the Albatross [13] model with weighted F1-score of 97.10. In comparison, `HostileNet` yields a slightly better score (97.52) than the wining system.

In the fine-grained setup, we report F1-scores for each hostile dimension along with the weighted F1-score. The top performing systems at CONSTRAINT-2021 shared task are Zeus (45.52), Bestfit AI (82.44), IREL IIIT-H (59.78), and Monolith (61.20) for *defamation*, *fake*, *hate*, and *offensive* dimensions, respectively. In comparison, `HostileNet` obtains improved performances in *defamation*, *fake*, and *hate* classes. Moreover, on average, `HostileNet` outperforms the best system by $\sim$ 2%—it reports a 66.32 weighted F1-score compared to 64.40 of Zeus. Note that none of the top-performing systems are consistent—they report the best result for one dimension only even though they trained separate systems for each dimension. On the other hand, our proposed model, `HostileNet`, is a unified system and achieves state-of-the-art performances in three out of four dimensions—it reports comparative scores in the *offensive* dimension. Furthermore, it obtains state-of-the-art performance in both the fine-grained and coarse-grained setups on average. We achieve a weighted F1-score of 68.64 over fivefold cross validation on the entire fine-grained CONSTRAINT-2021 dataset. Thus, the obtained results signify the robustness of `HostileNet` in detecting four hostile dimensions. In addition to the weighted F1-score metric used by CONSTRAINT-2021, we also report Jaccard score (JS), macro F1-score (m-F1), and Hamming loss (HL) for multilabel fine-grained setup in Table II. We observe that our model outperforms all the baseline systems for JS and m-F1, while being third best in HL (the lower the value, the better the system).

After establishing the efficacy of `HostileNet`, we perform a series of ablation studies to understand the effect of various submodules in the architecture. We begin by removing the hand-crafted features (hashtag, emoticon, and lexicon embeddings) from `HostileNet` in sequence. We report the ablation results at the lower part of Table II. In a fine-grained setup, we observe a decrease of 0.56 weighted F1-score points with the removal of hashtag embeddings from `HostileNet`. For the same setup, a drop of 0.4 is observed in the case of coarse-grained. The drop in performance reflects the role

TABLE II

RESULTS OF OUR HOSTILENET ARCHITECTURE COMPARED WITH TOP BASELINES ON THE CONSTRAINT-2021 [18] DATASET ALONG WITH ABLATION RESULTS OF HOSTILENET (LAST FOUR ROWS). HASHTAGS, EMOTICONS, AND LEXICONS DENOTE THE HAND-CRAFTED FEATURE EMBEDDINGS AS DESCRIBED IN SECTION IV. JS DENOTES JS, m-$F1$ DENOTES MACRO $F1$-SCORE, AND HL DENOTES HL FOR MULTILABEL FINE-GRAINED SETUP

Model	Fine-Grained								Coarse-Grained
	Class-wise				Overall				
	Def F1	Fake F1	Hate F1	Off F1	w-F1	JS	m-F1	HL	w-F1
CONSTRAINT-2021 Baseline	39.92	68.69	49.26	41.98	54.20	35.77	51.92	35.22	84.22
Albatross	42.80	81.40	49.69	56.49	61.11	42.07	57.59	27.91	97.10
Bestfit AI	31.54	82.44	58.56	58.95	62.21	43.01	57.87	**23.75**	96.61
Monolith	42.00	77.41	57.25	61.20	62.50	42.73	58.7	24.77	95.83
IREL IIIT-H	44.65	77.18	59.78	58.80	62.96	43.96	60.10	24.35	97.16
Zeus	45.52	81.22	59.10	58.97	64.40	45.40	61.20	25.44	96.07
HostileNet	**48.96**	**82.93**	60.14	61.02	**66.32**	**47.53**	**63.26**	24.45	**97.52**
(−) Hashtags	48.81	81.80	58.51	**62.16**	65.76	46.98	62.82	26.05	97.21
(−) Emoticons	45.74	82.35	**60.88**	59.17	65.31	46.35	62.03	24.80	96.24
(−) Lexicons	47.32	79.74	57.19	60.90	64.17	45.28	61.28	27.94	96.85
(−) Pretraining	44.44	80.86	58.98	58.82	64.02	44.98	60.77	33.39	96.55

of hashtag embeddings in influencing information virality and social movement, as shown initially by Wang et al. [31]. Subsequently, we ignore the emoticon embeddings and observe performance drops of 0.45 and 0.99 F1-score points in the fine-grained and coarse-grained setups, respectively. In the next step, once again the performance drop is observed when we skip the lexicon embedding in HostileNet as well. In the last row of Table II, we also see the effect of using the pretrained HindiBERT model on the training of HostileNet. Overall, the removal of hand-crafted features and pretraining have adverse effects on both fine-grained and coarse-grained setups with a considerable drop of 2.30 and 1.28 weighted F1-score points, respectively. The above ablation results cement our intuition of leveraging the hand-crafted features for improved learning of HostileNet.

D. Explainability Using Tuned Attention Vectors

We also analyze the attention vectors as computed by HostileNet. Table III demonstrates the heatmaps for two test samples. In addition, we also report the gold attention scores for each hostile dimension for comparison. The ground-truth labels for the samples are (*defamation* and *fake*) and (*hate* and *offensive*), which HostileNet correctly predicts with attention tuning. In sample 1, we observe that HostileNet puts greater attention on words like "आरोप" (Aarop | Blame), "सोची समझी" (Sochi Samji | Thought out), and so on, for the *defamation* class and words like "पुलवामा" (Pulwama | Pulwama) (related to Pulwama Attack 2019[8]) and "अभिनंदन" (Abhinandan | Abhinandan), an Indian Air Force pilot who was held captive in Pakistan in counter-strike, are very well highlighted for the *fake* class. Similarly, in sample 2, more attention is given to the words "दंगों" (Dango | Riots) and "युद्ध" (Yudh | War) which goes on to show the provocative nature of this hateful post. On the other hand, for *offensive* class, we notice that the word "कु#या" (Ku##ia | B##ch) is the second most attended word after the username of the victim. In both cases, it can be further observed that the HostileNet's attention scores are very close to the gold attention scores. It suggests that the optimization of the KL divergence between the model's attention vectors and gold attention vectors facilitates the model to learn the relevant and important tokens as close as to the training distribution. Apart from KL divergence, we also experiment with other loss functions such as Mean Squared Error (MSE) and Asymmetric Loss (ASL). It is evident from Table IV that KL divergence loss has the best effect on the learning of HostileNet in fine-grained setup followed by AL.

To further establish the efficiency of the attention vector tuning, we also present the heatmaps of HostileNet's attention vectors without any fine-tuning (i.e., no optimization with respect to the gold attention vector). It is evident that the model without (w/o) tuning finds it difficult to attend to the relevant words in the post; hence, it fails to predict the hostile dimensions correctly—it predicts (*defamation*, *hate*, and *offensive*) as the hostile labels for both the samples. Furthermore, in the absence of the optimization of attention vectors, HostileNet reports a performance degradation of two points in F1-score (refer Table IV), thus supporting our claim that fine-tuning attention vectors for each hostile dimension indeed has a positive effect on the overall performance.

E. Multilabel Lexicon Analysis

In this section, we present our analysis of the multilabel lexicon for each hostile dimension. For each token, we select the label with the maximum score, thus creating a list of tokens for each label. The number of tokens associated with the *defamation*, *fake*, *hate*, and *offensive* dimensions are 2652, 3646, 2146, and 2417, respectively. The remaining 4923 tokens correlate with the nonhostile dimension. Table V

[8]https://en.wikipedia.org/wiki/2019_Pulwama_attack

TABLE III

ATTENTION HEATMAPS FOR TWO HOSTILE SAMPLES FROM THE TEST SET. FOR EACH DIMENSION, WE PRESENT THE RESPECTIVE ATTENTION SCORES (DARKER SHADE REPRESENTS HIGHER WEIGHT) AS COMPUTED BY HOSTILENET WITH (W/) AND WITHOUT (W/O) TUNING THE ATTENTION VECTORS. WE ALSO REPORT THE GOLD ATTENTION VECTORS FOR EACH DIMENSION FOR COMPARISON. FOR THE GIVEN SAMPLES, THE GROUND-TRUTH LABELS ARE (DEFAMATION AND FAKE) AND (HATE AND OFFENSIVE), RESPECTIVELY

		Attention Vector	Attention Heat Map
Sample 1	Defamation	Gold	अभिनंदन ने बीजेपी पर आरोप लगाते हुए बयान दिया है कि पुलवामा हमला बीजेपी की सोची समझी साजिश थी #FactCheck
		HostileNet w/ tuning	अभिनंदन ने बीजेपी पर आरोप लगाते हुए बयान दिया है कि पुलवामा हमला बीजेपी की सोची समझी साजिश थी #FactCheck
		HostileNet w/o tuning	अभिनंदन ने बीजेपी पर आरोप लगाते हुए बयान दिया है कि पुलवामा हमला बीजेपी की सोची समझी साजिश थी #FactCheck
	Fake	Gold	अभिनंदन ने बीजेपी पर आरोप लगाते हुए बयान दिया है कि पुलवामा हमला बीजेपी की सोची समझी साजिश थी #FactCheck
		HostileNet w/ tuning	अभिनंदन ने बीजेपी पर आरोप लगाते हुए बयान दिया है कि पुलवामा हमला बीजेपी की सोची समझी साजिश थी #FactCheck
		HostileNet w/o tuning	अभिनंदन ने बीजेपी पर आरोप लगाते हुए बयान दिया है कि पुलवामा हमला बीजेपी की सोची समझी साजिश थी #FactCheck
Sample 2	Hate	Gold	@Username ट्रम्प हमारे लोकतंत्र के लिए खतरा हैं ! इसके अलावा , अगर मेरा पक्ष किसी एक मुद्दे पर हार जाता है , तो मैं इस कु#या को बंद करने की योजना बनाता हूं ! दंगों ! गृह युद्ध !
		HostileNet w/ tuning	@Username ट्रम्प हमारे लोकतंत्र के लिए खतरा हैं ! इसके अलावा , अगर मेरा पक्ष किसी एक मुद्दे पर हार जाता है , तो मैं इस कु#या को बंद करने की योजना बनाता हूं ! दंगों ! गृह युद्ध !
		HostileNet w/o tuning	@Username ट्रम्प हमारे लोकतंत्र के लिए खतरा हैं ! इसके अलावा , अगर मेरा पक्ष किसी एक मुद्दे पर हार जाता है , तो मैं इस कु#या को बंद करने की योजना बनाता हूं ! दंगों ! गृह युद्ध !
	Offensive	Gold	@Username ट्रम्प हमारे लोकतंत्र के लिए खतरा हैं ! इसके अलावा , अगर मेरा पक्ष किसी एक मुद्दे पर हार जाता है , तो मैं इस कु#या को बंद करने की योजना बनाता हूं ! दंगों ! गृह युद्ध !
		HostileNet w/ tuning	@Username ट्रम्प हमारे लोकतंत्र के लिए खतरा हैं ! इसके अलावा , अगर मेरा पक्ष किसी एक मुद्दे पर हार जाता है , तो मैं इस कु#या को बंद करने की योजना बनाता हूं ! दंगों ! गृह युद्ध !
		HostileNet w/o tuning	@Username ट्रम्प हमारे लोकतंत्र के लिए खतरा हैं ! इसके अलावा , अगर मेरा पक्ष किसी एक मुद्दे पर हार जाता है , तो मैं इस कु#या को बंद करने की योजना बनाता हूं ! दंगों ! गृह युद्ध !

Sample 1 Ground truth: [Defamation, Fake]; HostileNet w/ tuning: [Defamation, Fake]; HostileNet w/o tuning: [Defamation, Hate, Offensive];
Sample 2 Ground truth: [Hate, Offensive]; HostileNet w/ tuning: [Hate, Offensive]; HostileNet w/o tuning: [Defamation, Hate, Offensive];

TABLE IV

RESULTS OF OUR MODEL HOSTILENET WITH DIFFERENT CHOICE OF LOSS FUNCTIONS IN ORDER TO TUNE HINDIBERT'S ATTENTION HEADS. MSE, ASL, AND KD STAND FOR MSE, ASYMMETRIC, AND KL DIVERGENCE LOSS FUNCTIONS. CG DENOTES COARSE-GRAINED SETUP

Loss	Fine-Grained					CG
	Def F1	Fake F1	Hate F1	Off F1	w-F1	w-F1
None	46.81	81.39	58.31	58.15	64.31	96.36
$\mathbb{L}_{MSE}$	42.81	81.02	58.84	**61.06**	64.26	96.79
$\mathbb{L}_{ASL}$	46.59	80.96	59.79	60.08	64.92	96.55
$\mathbb{L}_{KLD}$	**48.96**	**82.93**	**60.14**	61.02	**66.32**	**97.52**

lists a few sampled tokens for each dimension. We present our observations for each label as follows.

1) For the defamation dimension, we observe that a significant number of tokens revolve around politics—it correlates with the fact that some supporters of political parties try to malign or defame each other.
2) In the case of fake news, we observe the presence of various country names, such as *India*, *China*, *Japan*, and COVID-19-related terms. It could be because the dataset curation period [18] overlaps with the early stage of the pandemic and comprises many unverified and fake news.
3) Hate posts in India majorly revolve around religious and casteism slurs. Our curation of hate lexicon rightfully captures such tokens (*Hindu*, *Muslim*, *Caste*, *Religious*, etc.) as listed in Table V. Moreover, various political parties use terms such as "*foreigner*" and "*patriot*" to breed hate against some individual or a community.
4) We observe that our algorithm correctly maps the majority of the swear and slang words in the dataset, such as "*d#g*,[9]" "*bi##h*," "*ra##al*," "*s##erF##er*," "*m##erF##er*," and so on, to *offensive* dimension.
5) In the case of the nonhostile category, most of the tokens are neutral in nature and simple day-to-day innocuous words such as *earn*, *reader*, *professor*, *afternoon*, *metro*, and so on.

Overall, our analysis shows that the multilabel lexicon was able to capture the semantics of the tokens with reasonable precision and assists the model in improved performance.

F. Quantitative Error Analysis

In this section, we quantitatively analyze the errors committed by HostileNet. From Fig. 2, we observe that in all cases except for the *fake* label, the *false-positives* are quite high. Moreover, the precision for the *defamation* label is particularly low as the system reports higher *false-positives* than the *true-positives*. On the other hand, both *false-positives* and *false-negatives* are comparatively on the lower side in *fake* class detection; hence, HostileNet yields good F1-scores of 82.93. We relate the above phenomena with the available number of samples for each dimension in the dataset (refer

[9]A derogatory term in Hindi.

TABLE V

MAPPING OF TOKENS FROM THE CONSTRAINT-2021 DATASET TO THE DIMENSION HAVING MAXIMUM LEXICON SCORE CALCULATED USING OUR PROPOSED METHODOLOGY. WE PRESENT THE TOKENS IN ORIGINAL DEVANAGARI, FOLLOWED BY ITS ENGLISH TRANSLATION FOR READABILITY. SLURS WORDS ARE DEPICTED IN ITALICS AND HOSTILE SLURS ARE CENSORED WITH HASHES (#)

Label	Lexicons in Hindi (English)
Defamation	आतंकी (Terrorist), फूल (Fool), हिटलर (Hitler), कंगना (Kangana), बीजेपी (BJP), कांग्रेस (Congress), प्रवक्ता (Spokesman), चुनाव (Elections), भ्रष्ट (Corrupt), क्रांति (Revolution), ड्रामा (Drama)
Fake	पुलिस (Police), कोरोना (Corona), भारत (India), जापान (Japan), चीन (China), छूट (Discount), रिकॉर्ड (Record), जल्द (Immediate), गिरफ्तार (Arrest), गाँधी (Gandhi), शक्तियों (Powers), विघटन (Dissolution), वक्तव्य (Statement)
Hate	हिन्दु (Hindu), मुसलमान (Muslim), धर्मप (Religious), जातियों (Castes), सरकार (Government), अधिकार (Rights), विरोध (Protest), संविधान (Constitution), नरक (Hell), हत्या (Killing), फांसी (Hanging), अभिमान (Pride), देशभक्त (Patriot), विदेशी (Foreigner)
Offensive	स#ला (*Ra#cal*), कु#ता (*Dog*), कु#्या (*Bi##h*), क##ने (*Ba##ard*), बह###द (*S###erF###er*), मा####द (*M###erF###er*), तर्क (Argument), मर्द (Male), दलितों (Dalits), ताना (Taunt), गोमांस (Beef), पागल (Mad), विफलता (Failure)
Non-Hostile	बिजनेस (Business), अर्न (Earn), मेट्रो (Metro), रैल (Rail), मुस्कान (Smile), पाठक (Reader), प्रोफेसर (Professor), सैन्य (Military), उड़ान (Flight), दोपहर (Afternoon), राष्ट्रीय (National), अंतर्राष्ट्रीय (International), बढ़ती (Increase), प्रकाश (Light)

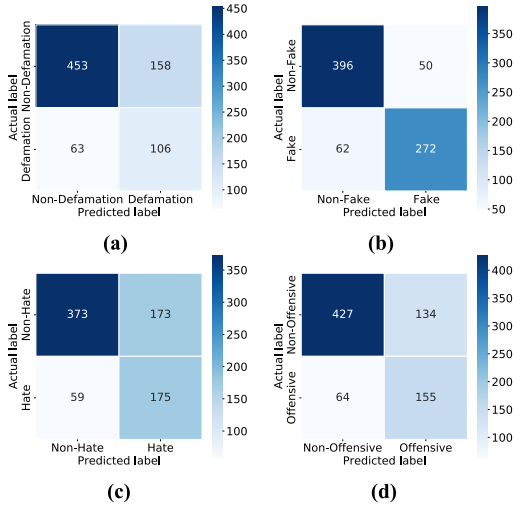

Fig. 2. Confusion matrix plot across four hostile dimensions. (a) Defamation. (b) Fake. (c) Hate. (d) Offensive.

Table I)—*defamation* has the least number of samples (810), whereas the *fake* samples are the highest (1638).

G. Qualitative Error Analysis

In Table VI, we investigate some error cases of HostileNet and the best baseline system—IREL IIIT-H [16] for the coarse-grained analysis (samples 1 and 2) and Zeus [17] for the fine-grained analysis (samples 3 and 4). The first example is *nonhostile*; however, both HostileNet and the best baseline misclassify it as *hostile*. The possible reason might be the presence of the term "बलिदान" (Balidaan | Oblation) in the post. To get deeper insights, we obtain a multilabel lexicon vector for "बलिदान" and analyze its values. The vector hence obtained is [0.0, 0.9998, 0.00004, 0.0, 0.00006], where each index denotes one hostile dimension—*defamation*, *fake*, *hate*, *offensive*, and *nonhostile*, respectively. The lexicon vector clearly shows that the term is primarily associated with the "*fake*" dimension, and thus the post was misclassified by HostileNet. Similarly, we observe misclassifications by both the systems in the second example as well—both tag the post as *nonhostile*. These two examples show that both systems fail to understand the *hostility* in the posts.

As expected, the predictions in the fine-grained setup are much more complex than the coarse-grained setup due to the multilabel classification. In the majority of the cases, we observe that HostileNet makes at least one correct prediction. In the third example, HostileNet takes a conservative approach and predicts partially correct class *hate*; however, it fails to recognize the *offensiveness* in the post. Similarly, HostileNet makes one correct (*defamation*) and one incorrect (*hate*) prediction in the fourth example, whereas Zeus predicts two incorrect (*hate* and *offensive*) labels and fails to identify the *defamation* class. In both cases, HostileNet predicts one extra class—*hate* in the third example and *defamation* in the fourth example.

H. Human Evaluation In-the-Wild

To further establish the efficacy of HostileNet, we assess its performance against real-world posts on the web. We collect 71 random samples from various online social media platforms and manually annotate them for the hostility detection task.[10] It is worth noting that we gathered these instances in-the-wild. As expected, most of the coarse-grained labels are correctly identified by HostileNet (65 out of 71) and yield a weighted F1-score of 91.58 (refer Table VII). On the other hand, HostileNet reports a 72.46 weighted F1-score in the fine-grained setup. Our observations from the in-the-wild evaluation reveal that HostileNet adapts swiftly and precisely with confidence to unseen posts. Further details of the in-the-wild evaluation are elucidated in the Supplementary material.

VI. USER SURVEY

To evaluate the robustness of HostileNet at the user level, we conduct a user survey where we ask 15 respondents to test random posts in Hindi against our proposed HostileNet model. The respondents are between the ages of 21 and 30, frequently use social media, and speak Hindi as their first language. We discuss the labels with participants before asking them to rate the labels predicted by HostileNet on a five-point Likert scale, with five indicating highly acceptable output and one indicating unacceptable

[10]We obtain an interannotator score (*Cohen kappa*) of 0.88.

TABLE VI

ERROR ANALYSIS FOR HOSTILE POST CLASSIFICATION USING MISCLASSIFIED EXAMPLES BY HOSTILENET. SAMPLES 1 AND 2 ANALYZE HOSTILENET AND IREL IIIT-H [16] IN THE COARSE-GRAINED SETUP, WHEREAS SAMPLES 3 AND 4 COMPARE HOSTILENET AND ZEUS [17] IN THE FINE-GRAINED SETUP. IREL IIIT-H AND ZEUS ARE THE BEST COARSE-GRAINED AND FINE-GRAINED HOSTILE POST DETECTION BASELINES, RESPECTIVELY, IN THE CONSTRAINT-2021 [12] HINDI SHARED TASK

	Example	Ground Truth	Prediction	
			HostileNet	Best Baseline
1	आज हज़रत इमाम हुसैन साहब की बहादुरी और बलिदान को याद करते हुए हम सच्चाई और इंसाफ़ की राह पर चलने का संकल्प लेते हैं। Today, in the remembrance of bravery and sacrifice of Late Hazrat Imam Hussain, we take resolution to walk on the path of truth and justice.	Non-Hostile	Hostile	Hostile
2	मुहर्रम के लिए छूट है पर गणेश उत्सव में कुछ लोग मूर्ति विसर्जन को भी जाय तो पाबंदी । तेलंगाना सरकार को शर्म आनी चाहिए। @Username @Username @Username @Username URL There is leniency for Muharram but not for immersion of statues in river during Ganesh Chaturthi. Telegana government should be ashamed. @Username @Username @Username @Username URL	Hostile (Offensive)	Non-Hostile	Non-Hostile
3	facebook.com/pram####46 ये प्रमोद सिंह जी की फेसबुक आईडी है जिस पर एक विडिओ डाला गया है जिसमे अबु आजमी और मुंबई पुलिस व साजिद भाई के समर्थन में नारे लगाए जा रहे है जिसे इन्होंने #पाकिस्तान जिंदाबाद बताया है जिससे गलत संदेश जाता है @Username इस पर कार्यवाही करे...@Username facebook.com/pram####46 This is the Facebook ID of Pramod Singh ji, on which a video has been put in which slogans are being raised in support of Abu Azmi and Mumbai Police and Sajid Bhai, which he described as #PakistanZindabad Wrong message goes @Username take action on this ... @ Username	Hostile (Hate, Offensive)	Hate	Non-Hostile
4	देखो देखो यह है मोदी दिखाई कुछ नहीं दे रहा है क्यों विकास गायब है Look look this is Modi nothing is visible because development is missing	Hostile (Defamation)	Defamation, Hate	Hate, Offensive

TABLE VII

RESULTS OF OUR HOSTILENET ARCHITECTURE FOR IN-THE-WILD EVALUATION

Fine-Grained					Coarse-Grained
Def F1	Fake F1	Hate F1	Off F1	w-F1	w-F1
14.29	87.50	86.67	75.86	72.46	91.58

output. In 50 of the 75 instances, the score was greater than or equal to 3. Nearly 42.66% of the predictions are rated as highly acceptable. According to the user survey results, we observe that HostileNet can be used to label unseen posts accurately and efficiently. We report our results in theSupplementary material.

VII. CONCLUSION

In this work, we presented a unified neural network architecture, HostileNet, for hostile post detection in Hindi across four dimensions—*fake*, *hate*, *offensive*, and *defamation*. Experiments illustrated the superiority of our proposed model HostileNet against various existing state-of-the-art systems. We experimentally showed an improvement of 0.36% and 1.92% in the weighted F1-score for the coarse-grained and fine-grained hostile post detection tasks, respectively, over the best-performing baseline systems. Furthermore, we visualized and illustrated the robustness and explainability of HostileNet through attention heatmap analysis and the token's association score for each dimension. We observed that HostileNet with attention fine-tuning attends to relevant tokens corresponding to the associated hostile dimension. We conducted an exhaustive error analysis and compared the outcome against state-of-the-art systems. Finally, we demonstrated HostileNet's robustness by conducting a qualitative human evaluation and a user survey on random samples. Consequently, we provided empirical evidence that HostileNet can efficiently classify unseen social media posts into different hostile dimensions. Our analysis showed that HostileNet performed comparatively well for the majority (*fake*) class than the minority (*defamation*) class. Therefore, our future work would involve improving the performance of the minority class as well as increasing the size of the dataset. We used HindiBERT, which was pretrained on structured datasets (OSCAR common crawl and Wikipedia) and may not be well suited for social media posts, to obtain contextual representations. However, a major bottleneck is the unavailability of large-scale pretrained language models in Hindi for noisy platforms like Twitter. In the future, we intend to expand our work utilizing contextual representations from noisy datasets. Also, we plan to extend hostile post detection for other low-resource languages such as Bengali and Marathi.

VIII. ACKNOWLEDGMENT

The authors would also like to thanks the Infosys Centre for AI (CAI), IIIT Delhi for the valuable support.

REFERENCES

[1] D. Devakumar, G. Shannon, S. S. Bhopal, and I. Abubakar, "Racism and discrimination in COVID-19 responses," *Lancet*, vol. 395, no. 10231, p. 1194, Apr. 2020. [Online]. Available: https://www.thelancet.com/pdfs/journals/lancet/PIIS0140-6736(20)30792-3.pdf

[2] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, and T. Mikolov, "Learning word vectors for 157 languages," 2018, *arXiv:1802.06893*.

[3] I. Ahmed and J. A. Manik. (2012). *Attacks on Buddhist Templates a Hazy Picture Appears*. [Online]. Available: https://www.thedailystar.net/news-detail-252212

[4] E. Spertus, "Smokey: Automatic recognition of hostile messages," in *Proc. IAAI 9th Conf. Innov. Appl. Artif. Intell.*, 1997, pp. 1058–1065.

[5] I. Kwok and Y. Wang, "Locate the hate: Detecting tweets against blacks," in *Proc. AAAI*, vol. 27, no. 1, 2013, pp. 1–2.

[6] Z. Waseem and D. Hovy, "Hateful symbols or hateful people? Predictive features for hate speech detection on Twitter," in *Proc. NAACL Student Res. Workshop*, 2016, pp. 88–93.

[7] T. Davidson, D. Warmsley, M. Macy, and I. Weber, "Automated hate speech detection and the problem of offensive language," in *Proc. ICWSM*, vol. 11, no. 1, 2017, pp. 1–4.

[8] P. Badjatiya, S. Gupta, M. Gupta, and V. Varma, "Deep learning for hate speech detection in tweets," in *Proc. 26th Int. Conf. World Wide Web Companion (WWW Companion)*, 2017, pp. 1–2.

[9] T. Tran et al., "HABERTOR: An efficient and effective deep hatespeech detector," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2020, pp. 7486–7502.

[10] T. Mandl et al., "Overview of the HASOC track at FIRE 2019: Hate speech and offensive content identification in indo-European languages," in *Proc. 11th Forum for Inf. Retr. Eval.*, Dec. 2019, pp. 14–17.

[11] D. Kar, M. Bhardwaj, S. Samanta, and A. P. Azad, "No rumours please! A multi-indic-lingual approach for COVID fake-tweet detection," in *Proc. Grace Hopper Celebration India (GHCI)*, Feb. 2021, pp. 1–6.

[12] P. Patwa et al., "Overview of constraint 2021 shared tasks: Detecting English COVID-19 fake news and Hindi hostile posts," in *Proc. CONSTRAINT Worskhop*. Cham, Switzerland: Springer, 2021, pp. 42–53.

[13] V. Bhatnagar, P. Kumar, S. Moghili, and P. Bhattacharyya, "Divide and conquer: An ensemble approach for hostile post detection in Hindi," in *Proc. CONSTRAINT Workshop*. Cham, Switzerland: Springer, 2021, pp. 244–255.

[14] Sarthak, S. Shukla, and K. V. Arya, "Detecting hostile posts using relational graph convolutional network," 2021, *arXiv:2101.03485*.

[15] O. Kamal, A. Kumar, and T. Vaidhya, "Hostility detection in Hindi leveraging pre-trained language models," in *Proc. CONSTRAINT Workshop*. Cham, Switzerland: Springer, 2021, pp. 213–223.

[16] T. Raha, S. Ghosh Roy, U. Narayan, Z. Abid, and V. Varma, "Task adaptive pretraining of transformers for hostility detection," in *Proc. CONSTRAINT Workshop*. Cham, Switzerland: Springer, 2021, pp. 236–243.

[17] S. Zhou, J. Li, and H. Ding, "Fake news and hostile posts detection using an ensemble learning model," in *Proc. CONSTRAINT Workshop*. Cham, Switzerland: Springer, 2021, pp. 74–82.

[18] M. Bhardwaj, M. Shad Akhtar, A. Ekbal, A. Das, and T. Chakraborty, "Hostility detection dataset in Hindi," 2020, *arXiv:2011.03588*.

[19] S. Malmasi and M. Zampieri, "Challenges in discriminating profanity from hate speech," *J. Experim. Theor. Artif. Intell.*, vol. 30, no. 2, pp. 187–202, Mar. 2018.

[20] M. Sajjad, F. Zulifqar, M. U. G. Khan, and M. Azeem, "Hate speech detection using fusion approach," in *Proc. Int. Conf. Appl. Eng. Math. (ICAEM)*, Aug. 2019, pp. 251–255.

[21] M. Zampieri, S. Malmasi, P. Nakov, S. Rosenthal, N. Farra, and R. Kumar, "Predicting the type and target of offensive posts in social media," in *Proc. NAACL-HLT*, 2019, pp. 1415–1420.

[22] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. NAACL-HLT*, Jun. 2019, pp. 4171–4186.

[23] P. Parikh et al., "Multi-label categorization of accounts of sexism using a neural framework," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, 2019, pp. 1642–1652.

[24] P. Mathur, R. Sawhney, M. Ayyar, and R. Shah, "Did you offend me? Classification of offensive tweets in hinglish language," in *Proc. 2nd Workshop Abusive Lang. Online (ALW)*, 2018, pp. 138–148.

[25] A. Sengupta, S. K. Bhattacharjee, M. S. Akhtar, and T. Chakraborty, "Does aggression lead to hate? Detecting and reasoning offensive traits in hinglish code-mixed texts," *Neurocomputing*, vol. 488, pp. 598–617, Jun. 2021.

[26] V. Bhatnagar, P. Kumar, and P. Bhattacharyya, "Investigating hostile post detection in Hindi," *Neurocomputing*, vol. 474, pp. 60–81, Feb. 2022.

[27] N. Doiron. (2020). *Monsoon NLP—Hindi Bert*. [Online]. Available: https://huggingface.co/monsoon-nlp/hindi-bert

[28] Y. Zou, T. Gui, Q. Zhang, and X.-J. Huang, "A lexicon-based supervised attention model for neural sentiment analysis," in *Proc. COLING*, 2018, pp. 868–877.

[29] K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning, "ELECTRA: Pre-training text encoders as discriminators rather than generators," 2020, *arXiv:2003.10555*.

[30] G. Guibon, M. Ochs, and P. Bellot, "From emojis to sentiment analysis," in *Proc. WACAI*, 2016, pp 1–8.

[31] R. Wang, W. Liu, and S. Gao, "Hashtags and information virality in networked social movement: Examining hashtag co-occurrence patterns," *Online Inf. Rev.*, vol. 40, no. 7, pp. 850–866, Nov. 2016.

[32] B. Eisner, T. Rocktäschel, I. Augenstein, M. Bosnjak, and S. Riedel, "emoji2vec: Learning emoji representations from their description," in *Proc. 4th Int. Workshop Natural Lang. Process. Social Media*, 2016, pp. 48–54.

[33] C. Baziotis, N. Pelekis, and C. Doulkeridis, "Datastories at SemEval-2017 task 4: Deep LSTM with attention for message-level and topic-based sentiment analysis," in *Proc. 11th Int. Workshop Semantic Eval. (SemEval)*, Aug. 2017, pp. 747–754.

[34] D. Kakwani et al., "IndicNLPSuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for Indian languages," in *Proc. Findings EMNLP*, Nov. 2020, pp. 4948–4961.