Natural Language Processing (2025), **31**, pp. 1323–1349
doi:10.1017/nlp.2024.60

CAMBRIDGE UNIVERSITY PRESS

ARTICLE

Focal inferential infusion coupled with tractable density discrimination for implicit hate detection

Sarah Masud[1,*], Ashutosh Bajpai[2,3,*] and Tanmoy Chakraborty[2]

[1]Indraprastha Institute of Information Technology Delhi, New Delhi, India, [2]Indian Institute of Technology Delhi, New Delhi, India, and [3]Wipro Research, Bengaluru, India
Corresponding author: Tanmoy Chakraborty; Email: tanchak@iitd.ac.in

(Received 9 February 2024; revised 28 October 2024; accepted 29 October 2024; first published online 13 December 2024)

Abstract

Although pretrained large language models (PLMs) have achieved state of the art on many natural language processing tasks, they lack an understanding of subtle expressions of *implicit* hate speech. Various attempts have been made to enhance the detection of implicit hate by augmenting external context or enforcing label separation via distance-based metrics. Combining these two approaches, we introduce `FiADD`, a novel focused inferential adaptive density discrimination framework. `FiADD` enhances the PLM finetuning pipeline by bringing the surface form/meaning of an implicit hate speech closer to its implied form while increasing the intercluster distance among various labels. We test `FiADD` on three implicit hate datasets and observe significant improvement in the two-way and three-way hate classification tasks. We further experiment on the generalizability of `FiADD` on three other tasks, detecting sarcasm, irony, and stance, in which surface and implied forms differ, and observe similar performance improvements. Consequently, we analyze the generated latent space to understand its evolution under `FiADD`, which corroborates the advantage of employing `FiADD` for implicit hate speech detection.

Keywords: Hate speech; Implicit hate speech; Distance metric; Hate detection

1. Introduction

The Internet has led to a proliferation of hateful content (Suler, 2004). However, what can be considered hate speech is subjective (Baucum, Cui, and John, 2020; Balayn *et al.* 2021). According to the United Nations,[a] *hate speech is any form of discriminatory content that targets or stereotypes a group or an individual based on identity traits.* In order to assist content moderators, practitioners are now looking into automated hate speech detection techniques. The paradigm that is currently being adopted is finetuning a pretrained language model (PLM) for hate speech detection. Akin to any supervised classification task, the first step is the curation of hateful instances. While instances of online hate speech have increased, they still form a small part of the overall content on the Web. For example, on platforms like Twitter, the ratio of hate/non-hate posts curated from the data stream is 1:10 (Kulkarni *et al.* 2023). Thus, data curators often employ lexicons and identity slurs to increase the coverage of hateful content.[b] While this increases the number of explicit samples, it comes at the cost of capturing fewer instances of implied/non-explicit hatred (Davidson *et al.* 2017; Silva *et al.* 2021). This skewness in the number of implicit samples contributes to less

*Equal Contribution.
[a]https://www.un.org/en/hate-speech/understanding-hate-speech/what-is-hate-speech
[b]The paper contains samples of hate speech, which are only included for contextual understanding.

© The Author(s), 2024. Published by Cambridge University Press. This is an Open Access article, distributed under the terms of the Creative Commons Attribution licence (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted re-use, distribution and reproduction, provided the original article is properly cited.

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

information being available for the models to learn from. Among the myriad datasets on hate speech (Vidgen and Derczynski, 2020; Poletto *et al.* 2021) in English, only a few (Caselli *et al.* 2020; ElSherief *et al.* 2021; Kennedy *et al.* 2022) have annotations for "implicit" hate.

Why is implicit hate hard to detect?

It has been observed that classifiers can work effectively with direct markers of hate (Lin, 2022; Muralikumar, Yang, and McDonald, 2023), *a.k.a* explicit hate. The behavior stems from the data distribution since slurs are more likely to occur in hateful samples than in neutral ones. On the other hand, implicit hate on the surface appears lexically and semantically closer to statements that are non-hate/neutral. Inferring the underlying stereotype and implied hatefulness in an implicit post requires a combination of multi-hop reasoning with sufficient cultural reference and world knowledge. Existing research has established that even the most sophisticated systems like ChatGPT perform poorly in case of implicit hate detection (Yadav *et al.* 2024).

At the distribution level, the aim is to bring the surface meaning closer to its implied meaning, that is, what is said *versus* what is intended (ElSherief *et al.* 2021; Lin, 2022). One way to reduce the misclassification of implicit hate is to manipulate the intercluster latent space via contrastive or exemplar sampling (Kim, Park, and Han, 2022). Contrastive loss similar to cross-entropy operates in a per-sample setting (Chopra, Hadsell, and LeCun, 2005), leading to suboptimal separation among classes (Liu *et al.* 2016). Another technique is to infuse external knowledge. However, without explicit hate markers, providing external knowledge increases the noise in the input signal (Lin, 2022; Yadav *et al.* 2024).

Proposed framework

In this work, we examine a framework for overcoming these two drawbacks. As an alternative to the per-sample contrastive approach in computer vision tasks, adaptive density discrimination (ADD) *a.k.a* magnet loss (Rippel *et al.* 2016) has been proposed. ADD does not employ the most positively and negatively matching sample; instead, it exploits the local neighborhood to balance the interclass similarity and variability. Extensive literature (Rippel *et al.* 2016; Snell, Swersky, and Zemel, 2017; Deng *et al.* 2019) has established the efficacy and superiority of ADD for computer vision tasks over contrastive settings. We hypothesize that its advantage can be extended to natural language processing (NLP) and attempt to establish the same in this work.

For our use case, ADD can help improve the regional boundaries around implicit and non-hate samples that lie close. However, simply employing ADD in a three-way classification of implicit, explicit, and non-hate will not yield the desired results due to the semantic and lexical similarity of implicit with non-hate. We, thus, introduce external context for implicit hate samples to bring them closer to their intended meaning (Kim *et al.* 2022), facilitating them to be sufficiently discriminated. To this end, we employ implied/descriptive phrases instead of knowledge tuples or Wikipedia summaries based on empirical findings (Lin, 2022; Yadav *et al.* 2024) that the latter tend to be noisy if the tuples are not directly aligned with the entities in the input statement. As outlined in Figure 1, our proposed pipeline focused inferential adaptive density discrimination (`FiADD`) improves the detection of implicit hate by employing distance-metric learning to set apart the class distributions in conjunction with reducing the latent space between the implied context and implicit hate. The dual nature of the loss function is aided by nonuniform weightage, with a focus on penalizing samples near the discriminant boundary.

Through extensive experiments, we observe that `FiADD` variants improve overall as well as implicit class macro-F1 for `LatentHatred` (ElSherief *et al.* 2021), `ImpGab` (Kennedy *et al.* 2022), and `AbuseEval` (Caselli *et al.* 2020) datasets. Our experimental results further suggest that our framework can generalize to other tasks where surface and implied meanings differ, such as humor (Labadie Tamayo, Chulvi, and Rosso, 2023), sarcasm (Abu Farha *et al.* 2022; Frenda, Patti, and Rosso, 2023), irony (Van Hee, Lefever, and Hoste, 2018), stance (Mohammad *et al.* 2016), etc.

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

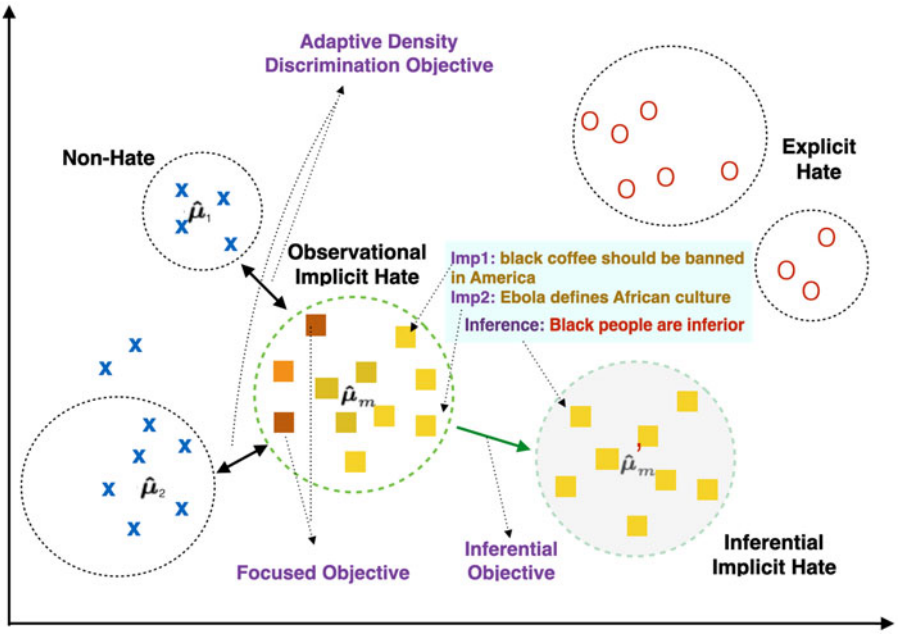


Figure 1. The three objectives of FiADD as applied to implicit hate detection are (a) adaptive density discrimination, (b) higher penalty on boundary samples, and (c) bringing the surface and semantic form of the implicit hate closer.

To establish that our results are not BERT (Devlin *et al.* 2019) specific, we also experiment with HateBERT (Caselli *et al.* 2021a), XLM (Chi *et al.* 2022), and LSTM (Hochreiter and Schmidhuber, 1997).

Contributions

In short, we make the following contributions through this study:[c]

- We perform a thorough literature survey of the implicit hate speech datasets. For the datasets employed in this study, we establish the closeness of implicitly hateful samples from non-hateful ones (Section 3) and use it to motivate our model design (Figure 1).
- We adopt ADD for the NLP setting and employ it to propose FiADD). The variants of the proposed setup allow it to be used as a pluggable unit for the PLM finetuning pipeline for the task of hate speech detection as well as other implicit text-based tasks (Section 4).
- We manually generate implied explanations/descriptions for 798 and 404 implicit hate samples for AbuseEval and ImpGab, respectively. These annotations contribute to corpora of unmasking implicit hate (Section 5).
- Our exhaustive experiments, analyses, and ablations highlight how FiADD compares with the cross-entropy loss on three hate speech datasets. We also extend our analysis to three other SemEval tasks to demonstrate the model's generalizability (Section 6).
- We perform an analysis to assess how the latent space evolves under FiADD (Sections 7).

[c]**Reproducibility.** Sample code and dataset are available at https://github.com/LCS2-IIITD/FIADD.

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

Research scope and social impact

Early detection of implicit hate will help reduce the psychological burden on the target groups, prevent conversation threads from turning more intense, and also assist in counter-hate speech generation. It is imperative to note the limitations of PLMs in understanding implicit hate speech. We attempt to overcome this by incorporating latent space alignment of surface and implied context. However, PLMs cannot replace human content moderators and can only be assistive.

2. Related work

Given that this study proposes a distance-based objective function primarily for implicit hate detection, the literature survey focuses on three main aspects—(i) implicit hate datasets, (ii) implicit hate detection, and (ii) improvement in the classification tasks via distance-based metrics. To determine the relevant literature for implicit hate within the vast hate speech literature, we make use of the up-to-date hate speech corpus[d] (Vidgen and Derczynski, 2020) as well ACL Anthology. The keywords used to search for relevant literature on the two corpora were "implicit" and "implicit hate," respectively.

Implicit hate datasets

The task of classifying hateful texts has led to an avalanche of hate detection datasets and models (Schmidt and Wiegand, 2017; Tahmasbi and Rastegari, 2018; Vidgen and Derczynski, 2020). Before discussing the literature, it is imperative to point out that issues with generalizable (Yin and Zubiaga, 2021), biasing (Balayn *et al.* 2021; Garg *et al.* 2023), adversary (Masud *et al.* 2024b), and outdated benchmarks (Masud *et al.* 2024a) are prevalent for hate speech detection at large and forms an active area of research.

Focusing on implicit hate datasets, we searched the hate speech database (Vidgen and Derczynski, 2020) with the keyword "implicit" as an indicator of whether the label set contains "implicit" labels and obtained 4 results. DALC (Caselli *et al.* 2021b) is a Dutch dataset consisting of 8*k* tweets curated from Twitter, labeled for the level of explicitness as well as the target of hate. Meanwhile, ConvAbuse consists of 4k English samples obtained from in-the-wild human-AI conversations with AI chatbots. Each conversation is marked for the degree of abuse (1 to -3) and directness (explicit or implicit). The other two datasets are also in English. `AbuseEval` (Caselli *et al.* 2020) is 14*k*, Twitter labeled for "abusiveness" and "explicitness." On the other hand, `ImpGab` (Kennedy *et al.* 2022) consists of 27k posts from Gab, which contain a hierarchy of annotations about the type and target of hate.

Meanwhile, from the ACL Anthology (we looked at the results from the first two pages out of 10), we discovered four more datasets. `LatentHatred` is the most extensive and most widely used implicit hate speech dataset. It consists of 21*k* Twitter samples labeled for implicit hate as well as 6 additional sub-categories of implicitness. It also contains free-text human annotations explaining the implied meaning behind the implicit posts. Along similar lines, SBIC (Ocampo *et al.* 2023c) is also a collection of 44*k* implicit posts curated from online platforms with human-annotated explainations. However, unlike complete sentences in `LatentHatred` SBIC focuses on single-phrased explainations. Further, SBIC does not have a direct marker for the explicitness of the post, and by default, all posts are implicit. For specific target groups and types of hate speech, such as sexism (Kirk *et al.* 2023) or xenophobia against immigrants (Sánchez-Junquera *et al.* 2021), researchers have also explored imploying multiple-level annotations as a means of obtaining granular label spans as explainations for the hateful instance. It serves as an alternative to free-text annotations, allowing for more structured and linguistic analysis (Merlo *et al.* 2023)

[d] https://hatespeechdata.com/

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

of implicitness. Further, building upon the multimodal hate meme dataset MMHS150K (Gomez *et al.* 2020), proposed a multimodal implied hate dataset (Botelho, Hale, and Vidgen, 2021) with the different types of implicitness occurring as a combination of the text and image.

More recently, the ISHate (Ocampo *et al.* 2023c) dataset has been curated by combining existing hate speech and counter-hate speech datasets and relabeling the samples for explicit–implicit markers, consisting of 30*k* samples labeled as explicit, implicit, subtle, or non-hate. It is interesting to note that in their analysis, the authors do not showcase how the different datasets interact with each other in the latent space. We hypothesize that the performance improvements in hate detection are obtained not as a result of modeling but due to the fact that these samples are obtained from distinct datasets, that is, distinct distributions. For example, counter-hate datasets do not contribute to the non-hate class. Meanwhile, the majority of implicit hate samples come from `LatentHatred` and ToxiGen (Hartvigsen *et al.* 2022). The latter is a curation of around 1 M toxic and implicit statements obtained by controlled generation.

Modeling implicit hate speech in NLP

Despite a large body of hate speech benchmarks, the majority of datasets fail to demark implicit hate. Even during the annotation process, fine-grained variants of offensiveness Founta *et al.* (2018); Kulkarni *et al.* (2023); Kirk *et al.* (2023) like abuse, provocation, and sexism are favored over the nature of hate, that is, explicit vs implicit. As the annotation schemas have a direct impact on the downstream tasks (Rottger *et al.* 2022), the common vouge of binary hate speech classification, while easier to annotate and model, focuses on explicit forms of hate. It also comes at the cost of not analyzing the erroneous cases where implicit hate is classified as neutral content. This further motivates us to examine the role of PLMs in three-way classification in this work.

Given the skewness in the number of implicit hate samples in a three-way classification setup, data augmentation techniques have been explored. For example, (Ocampo *et al.* 2023c) employed multiple data augmentation like substitution and back translation and observed that only when multiple techniques are combined did they surpass the finetuned HateBERT in performance. Adversarial data collection (Ocampo, Cabrio, and Villata, 2023a) and LLM-prompting (Kim *et al.* 2023) have also been explored for augmenting and improving implicit hate detection.

Language models are being employed not only to augment the implicit hate corpora but also to detect hate (Ghosh and Senapati, 2022; Plaza-del Arco, Nozza, and Hovy, 2023). With the recent trend of prompting generative large language models (LLMs), hate speech detection is now being evaluated under zero-shot (Nozza, 2021; Plaza-del Arco *et al.* 2023; Masud *et al.* 2024) and few-shot settings as well. An examination of the hate detection techniques under fine-grained hate speech detection has revealed that traditional models, either statistical (Waseem and Hovy, 2016; Davidson *et al.* 2017) or deep learning-based (Badjatiya *et al.* 2017; Founta *et al.* 2019), are characterized by a low recall for hateful samples (Kulkarni *et al.* 2023). To increase the information gained from the implicit samples, researchers are now leveraging external context.

Studies have mainly explored the infusion of external context in the form of knowledge entities, either in the form of knowledge-graph (KG) tuples (ElSherief *et al.* 2021) or Wikipedia summaries (Lin, 2022). However, both works have observed that knowledge infusion at the input level lowered the performance on fine-grained implicit categories. An examination of the quality of knowledge tuple (Yadav *et al.* 2024) infusion for implicit hate reveals that KG tuples fail to enlist information that directly connects with the implicit entities, acting more as noise than information. Apart from textual features, social media platform-specific features like user metadata, user network, and conversation thread/timeline can also be employed to improve the detection of hate and capture implicitness in long-range contexts (Ghosh *et al.* 2023). However, such features are platform-specific, complex to curate, and resource-intensive to operate (in terms of storage and memory to train network embeddings). From the latent space perspective, researchers have

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

explored how the infusion of a common target group can bring explicit and implicit samples closer (Ocampo, Cabrio, and Villata, 2023b), aiding in the detection of the latter. While the idea is intuitive since implicit hate and explicit slurs are specific to a target group, here, the extent of overlap in the case of multiple target groups or intersectional identities is not adequately addressed.

Distance-metric learning

Akin to supervised classification task in NLP, all the setups reviewed so far use an encoder-only BERT-based + cross entropy (CE) for finetuning. Therefore, in our study, BERT + CE acts as a baseline. Despite its popularity, CE's impact on the inter/intra-class clusters is suboptimal (Liu *et al.* 2016). Since classification tasks can be modeled as obtaining distant clusters per class, one can exploit clustering and distance-metric approaches to enhance the boundary among the labels, leading to improved classification performance. Distance-metric learning-based methods employ either deep divergence via distribution (Cilingir, Manzelli, and Kulis, 2020) or point-wise norm (Chopra *et al.* 2005). The most popular deep metric learning is the contrastive loss family (Chopra *et al.* 2005; Schroff, Kalenichenko, and Philbin, 2015; Chen *et al.* 2017). In order to improve upon the CE loss and benefit from the one-to-one mapping of the implicit hate and its implied meaning, contrastive learning has been explored (Kim *et al.* 2022), which has only provided slight improvement.

However, like cross-entropy, contrastive loss operates on a per-sample basis; even when considering positive and negative exemplars, they are curated on a per-sample basis. Clustering-inspired methods (Rippel *et al.* 2016; Song *et al.* 2017) have sought to overcome this issue by focusing on subclusters per class. ADD *a.k.a* magnet loss (Rippel *et al.* 2016) specifically lends a good starting point to operate the shift in the intercluster distance to extend to our use case. Based on the fact that ADD has surpassed contrastive losses in other tasks, we use ADD as a starting point and improve upon its formulation for implicit detection. As the current ADD setup fails to account for the implied meaning, we infuse external information into the latent space as an implied/inferential cluster.

3. Intuition and background

This section attempts to establish the need for distance-metric-based learning for the task of hate speech detection. Inspired by our initial experiments, we provide an intuition for ADD.

Hypothesis

A manual inspection of the hate speech datasets reveals that non-hate is closer to implicit hate than explicit hate. We, thus, measure the intercluster distance between non-hate and implicit hate compared to non-hate and explicit hate.

Setup

For three implicit hate speech datasets, that is, `LatentHatred` `ImpGab` and `AbuseEval`, we embed all the samples for a dataset in the latent space using 768 dimensional CLS embedding from BERT. The embeddings are not finetuned on any dataset or task related to hate speech so as to reduce the impact of confounding variables. We then consider 3 clusters directly adopted from the implicit, explicit, and non-hate classes and record the pairwise average linkage distance (ALD) and average centroid linkage distance (ACLD) among these clusters.

As the name suggests, for ACLD, we first obtain the embedding for the center of each cluster as a central tendency (mean or median) of all its representative samples and then compute

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

Table 1. The L1 intercluster distances between neutral (N) and explicit hate (E)), as well as non-hate and implicit hate (I) samples based on ALD and ACLD

	ALD		ACLD	
Dataset	N–E	N–I	N–E	N–I
`AbuseEval`	132.73	128.81	96.69	94.94
`ImpGab`	132.32	131.62	97.62	97.08
`LatentHatred`	125.12	122.59	90.86	90.69

the distance between the centers. This distance indicates the overall closeness of the two centers, which, in our case, measures the extent of similarity between the two classes. We also assess the latent space more granularly via ALD. In ALD, the distance between two clusters is obtained as the average distance between all possible pairs of samples where each element of the pair comes from a distinct group. It allows for a more fine-grained evaluation of the latent space, as not all data points are equidistant from each other or their respective centers. Formally, consider a system with E ($\mathbb{R} \in d$) points, where each point (e_i) belongs to one of the N clusters $c^n(e_i)$ and $\mu^n = \frac{1}{|c^n|} \sum_{e_i \in c^n} e_i$ is the cluster center. For clusters a and b, $ACLD^{a,b} = dist(\mu^a, \mu^b)$. Meanwhile, $ALD^{a,b} = \frac{1}{|c^a| * |c^b|} \sum_{e_i \in c^a \& e_j \in c^b} dist(e^i, e^j)$ where $(c^a(e_i) \neq c^b(e_j))$.

The intuition behind using both ACLD and ALD stems from the fact that online hate speech is part of the larger discourse on the Web. Thus, it is possible that at the level of individual datapoint labeling an isolated instance as hateful is hard. Furthermore, some implicit samples may be closer to the explicit hate samples in terms of lexicon or semantics. On the other hand, it is also possible for some non-hate samples to contain slurs that are commonplace and context-specific but not objectionable within the community (Diaz *et al.* 2022; Röttger *et al.* 2021). ACLD and ALD allow us to capture these dynamics at a macroscopic and a microscopic level.

Observation

From Table 1, we observe that under both ALD and ACLD, non-hate is closer to implicit samples. As expected, ALD shows more variability than ACLD. It follows from the fact that the mere presence of a keyword/lexicon does not render a sample as hateful.

Stemming from these observations, we see a clear advantage of employing a distance-metric approach that can exploit the granular variability in the latent space. `Adaptive density discrimination` (ADD) based clustering loss, which optimizes the inter and intra-clustering around the local neighborhood, directly maps to our problem of regional variability among the hateful and non-hateful samples. Further, our observations motivate the penalization of samples closer to the boundary responsible for increasing variability. The proposed model, as motivated by our empirical observations, is outlined in Figures 1 and 2.

3.1 Background on adaptive density discrimination

Here, we briefly outline ADD, which forms the backbone of our proposed framework. ADD is a clustering-based distance-metric. It evaluates the local neighborhood or clusters among the samples after each training iteration. At each epoch, after the training samples have been encoded into vector space, ADD clusters all data points within a class into K representative local groups via K-means clustering. The subclusters within a class help capture the inter/intra-label similarity

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

around the local neighborhood. If there are N classes, then each training sample will belong to one of the $N * K$ subclusters.

Given that mapping and tracking distances among all N∗K groups are computationally expensive, ADD randomly selects a reference/seed cluster I_s^c representing class C and then picks M imposter clusters from local neighborhood $I_{s1}^{c'}, \ldots, I_{sm}^{c'}$ but from disparate classes ($c \neq c'$) based on their proximity to seed cluster. To understand the concept of seed and imposter cluster better, consider the three-way hate speech classification task with implicit, explicit, and non-hate labels. As we aim to distinguish implicit hate speech better, we select one of the implicit hate subclusters as the seed. Consequently, the imposter clusters will be from explicit hate or non-hate, where implicit hate can be misclassified. ADD then samples D points uniformly at random from each sample cluster. For the d^{th} data point in m^{th} cluster, r_d^m is its encoded vector representation, with $C(.)$ representing the class for the sample under consideration. Subsequently, $\mu^m = \frac{1}{D}\sum_{d=1}^{D} r_d^m$ acts the mean representation of m^{th} cluster. Here, ADD applies Equation 1 to discriminate the local distribution around a point:

$$p^{ADD}(r_d^m) = \frac{e^{-\frac{1}{2\sigma^2}\left\|r_d^m - \mu^m\right\|_2^2 - \alpha}}{\sum_{\mu^o : C(\mu^o) \neq C(r_m^d)} -\frac{1}{2\sigma^2}\left\|r_d^m - \mu^o\right\|_2^2} \tag{1}$$

Here, α is a scalar margin for the cluster separation gap. The variance of all samples away from their respective centers is approximated via $\sigma^2 = \frac{1}{MD-1}\sum_{m=1}^{M}\sum_{d=1}^{D}\left\|r_d^m - \mu^m\right\|_2^2$.

After each iteration, as the embedding space gets updated, so does each of the subclusters; this lends to a dynamic nature to ADD. It allows for the selection of random subclusters and data points after each iteration. The overall loss is computed via Equation 2.

$$\ell(\Theta) = \frac{1}{MD}\sum_{m=1}^{M}\sum_{d=1}^{D} -\log p^{ADD}(r_d^m) \tag{2}$$

4. Proposed method

The proposed `FiADD` framework consists of a standard finetuning pipeline with encoder-only PLM followed by a projection layer R_h and a classification head (CH). To reduce the distance between the implicit hate (imp) and implied clusters (inf), `FiADD` measures, the average distance of implicit points from implied meaning as a ratio of its distance to explicit and non-hate subspaces. During the PLM finetuning, our setup combines with cross-entropy loss to improve the detection of hate. An overview of `FiADD`'s architecture is reflected in Figure 2. For each training instance $(x_d, y_d) \in X$, with x_d input and y_d label, $x_p = PLM(x_d)$ is the encoded representation obtained from the PLM. The encodings are projected to obtain $r_d = R_h(x_p)$. Here, $x_p \in \mathbb{R}^{768}$ and $r_d \in \mathbb{R}^{128}$ as $r_d \ll x_d$ allows for faster clustering.

Novel component: inferential infusion

As each output label y_d belongs to one of the distinct classes ($c_i \in C$), we employ the respective embeddings r_d and offline K-means algorithm to obtain K subclusters per class. For implicit hate samples, the latent representation of their implied/inferential counterparts $\tilde{x_d}$ is denoted as $\tilde{r_d} = R_h(\tilde{x_d})$. If $r_1^m, \ldots, r_d^m$ are representations for D samples of m^{th} implicit cluster, then $\tilde{r}_1^m, \ldots, \tilde{r}_d^m$ represent their respective inferential forms. The updated inferential adaptive density discrimination (ADD^{inf}) helps reduce the distance between $(r_d, \tilde{r_d})$ for implicit hate samples via Equation 3.

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

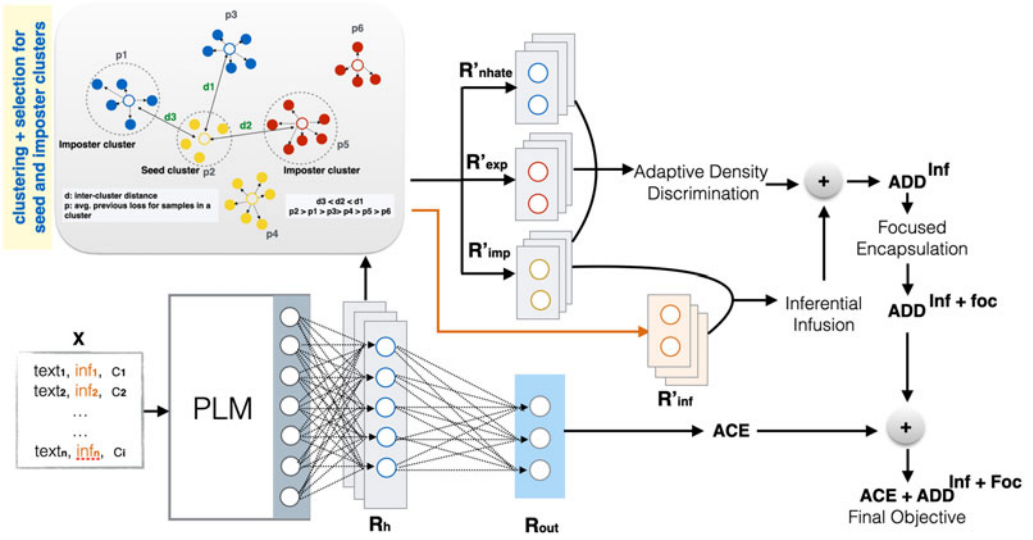


Figure 2. The architecture of FiADD. Input X is a set of texts, implied annotations (only for implicit class), and class labels. PLM: pretrained language model (frozen). R'_{nhate}, R'_{exp}, and R'_{imp} are the representatives for seed and imposter clusters of non-hate, explicit, and implicit, respectively. R'_{inf} represents inferential meaning for corresponding R'_{imp}. ACE is alpha cross-entropy, and $ADD^{inf+foc}$ is the adaptive density discriminator with inferential + focal objective.

$$p^{ADD^{inf}}(r_d^m) = \frac{e^{-\frac{1}{2\sigma^2}\|r_d^m - \mu^m\|_2^2 - \alpha} + e^{-\frac{1}{2\tilde{\sigma^2}}\|r_d^m - \tilde{\mu^m}\|_2^2 - \alpha}}{\sum_{\mu^o : C(\mu^o) \neq C(r_m^d)} -\frac{1}{2\sigma^2}\left\|r_d^m - \mu^o\right\|_2^2} \tag{3}$$

Here, μ^m (σ^2) and $\tilde{\mu^m}$ ($\tilde{\sigma^2}$) are the mean (variance) representations of the implicit and inferential/implied form for m^{th} implicit cluster, respectively.

The above equation can be broken into two parts. The first part is equivalent to ADD thus focusing on reducing the intra-cluster distance within the implicit class. The second part brings the implicit class closer to its implied meaning. Meanwhile, in the case of explicit or non-hate clusters, there is no mapping to the inferential/implied cluster, and ADD^{inf} in Equation 3 reduces to ADD in Equation 1.

Novel component: focal weight

Both ADD^{inf} and ADD assign uniform weight to all samples under consideration. In contrast, we have established that some instances are closer to the boundary of the imposter clusters and harder to classify (i.e., contribute more to the loss). Inspired by the concept of focal cross-entropy (Lin *et al.* 2017), we improve the ADD^{inf} objective by introducing $ADD^{inf+foc}$. Under $ADD^{inf+foc}$, the loss on each sample is multiplied by a factor called the *focused term* $(1 - p^{ADD^{inf}}(r_d^m))^{\gamma}$. γ, a hyperparameter, acts as a magnifier. The formulation assigns uniform weight as $\gamma \to 0$, reducing to ADD^{inf}. Analogously, the focal term is paying "more attention" to specific data points. Even without inferential infusion, our novel focal term can be incorporated as ADD^{foc} as enlisted in Equation 4.

$$\ell^{ADD^*}(\Theta) = \frac{1}{MD}\sum_{m=1}^{M}\sum_{d=1}^{D}\left[-(1 - p^{ADD^*}(r_d^m))^{\gamma} log(p^{ADD^*}(r_d^m))\right] \tag{4}$$

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

Here, $\ell^{ADD^{inf+foc}}$ ($\ell^{ADD^{foc}}$) captures the setup with (without) inferential objective. We utilize $p^{ADD^{inf}}$ (Equation 3) for the former and p^{ADD} (Equation 1) for the latter. Despite ADD^{foc} being a minor update on ADD, we empirically observe that focal infusion improves ADD.

Training pipeline

It should be noted that selecting the seed cluster and its subsequent imposter clusters is a random process for initial iterations. We assign the label with the highest loss margin for later iterations as the seed. Here, $ADD^{inf+foc}$ operates for implicit hate and overcomes the drawback of existing literature where implicit detection fails to account for implied context. It is also essential to point out that this evaluation is carried out in the local neighborhood, which is aided by the focal loss (Equation 4).

Overall loss

Apart from employing r_d in ℓ^{ADD^*}, it is also passed through a classification head $CH(r_d)$. We combine CE with the focal inference to obtain the final loss of `FiADD` with β controlling the contribution of the two losses as Equation 5.

$$\ell(\Theta) = \beta \ell^{CE}(\Theta) + (1 - \beta)\ell^{ADD^*}(\Theta) \tag{5}$$

Inference

During inference, the system does not have access to implied meaning. Once the PLM is trained via `FiADD`, the CH performs classification similar to any finetuned PLM. Here, we rely on the latent space being modified so that the implicit statements are closer to their semantic or implied form and sufficiently separated from other classes.

Note on K-mean

As a clustering algorithm, K-means is the most generic as it does not assume any dataset property (like hierarchy) except for the semantic similarity of the samples. Further, the K-means computation happens offline in each epoch, that is, it does not consume GPU resources. In the future, we aim to employ faster versions of K-means to improve training latency. Meanwhile, the computational complexity of `FiADD` during inference is the same as the finetuned PLM.

5. Experimental setup

`FiADD` provides an opportunity to improve the detection of implicit context. In the first set of experiments, we focus on the task of hate speech classification with datasets that consist of implicit hate labels. In the second set of experiments, we establish the generalizability of the proposed framework via SemEval[e] datasets on three separate tasks. Table 2 provides both sets' label-wise distribution. In all the tasks, the surface form of the text varies contextually from its semantic structure. Besides introducing the datasets and annotation schema, this section also outlines the hyperparameters and baselines curated for our evaluation.

[e] https://semeval.github.io/

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

Table 2. Datasets employed in evaluating the FiADD framework. The statistics enlist the class-wise distributions for (a) Hate Speech and (b) SemEval datasets

(a)					(b)			
Datasets	**source**	**Labels**			**Task**	**Labels**		
		N-OFF	EXP	IMP		N-SAR	SAR	
AbuseEval	Twitter	11173	2129	798	Sarcasm	3801	1067	
		N-OFF	EXP	IMP		N-IRO	IRO	
ImpGab	Gab	25102	2159	404	Irony	1890	1756	
		N-OFF	EXP	IMP		NEU	ANG	FAV
LatentHatred	Twitter	13291	1089	7100	Stance	918	1969	982

Implicit hate classification datasets

Based on our literature survey of implicit hate datasets, we discard the ones that are either multilingual (DALC) or multimodal (ConvAbuse, MMHS150K), as modeling them is out of the scope of current work. Further, SBIC and ToxiGen do not offer 3-way labels; hence, they are discarded, too. From among the English datasets left for assessment, we drop ISHate as it is an aggregated dataset, and its implicit samples are already covered by LatentHatred. Finally, we have LatentHatred AbuseEval, and ImpGab as English-based text-only datasets with explicit, implicit, and non-hate labels that suit our task. For LatentHatred, we employ the first level of annotation and the existing manual annotations of implied hatred for implicit samples. Meanwhile, AbuseEval and ImpGab do not have the implied descriptions. We manually annotate the implicit samples of these datasets with their implied meaning generated as free text.

Annotation for implied hate

Implied contexts are succinct statements that make explicit the underlying stereotype. Note that the implied context cannot be considered a comprehensive explanation for implicit hate but rather a more explicit understanding of the underlying subtle connotations. For AbuseEval and ImpGab, two expert annotators (one male and one female social media expert; age range between 29 to 35) perform the annotations based on the following guidelines:

- Implied meaning should consider the post's author's perspective.
- Implied meaning should emphasize on the post's content only.
- Annotations must be explicitly associated with the target entity.
- Annotations must contain a broader abusive context for the given post.
- Annotations should balance lexical diversity and uniformity w.r.t abuse toward a target group.

Annotation agreement

For our use case, annotation agreement scores help establish how well-aligned and coherent the explicit connotations are. To carry out the assessment, annotators A and B exchange a random sample of 30 annotation pairs. They score the pairs on a 5-point Likert scale (Likert, 1932), with 5 being the highest agreement. For AbuseEval, we obtain a mean agreement of 4.13 ± 1.13 and

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

Table 3. Some sample posts from `AbuseEval` and `ImpGab` along with their implied annotations. We also provide the cross-annotator scores and the cross-annotator remarks

Dataset	Implicit Post	Implied Annotation	Score	Remarks
`AbuseEval`	@USER How dare you tell me what Antifa is doing is right. They are not fighting fascism. They are fighting for their own stupid cause. They are further from fixing fascism than the Republicans themselves,	Anti-fascists are stupid and good for nothing.	5	Captured explicit target entity and the broader abuse in context
`AbuseEval`	@USER And this traitor thinks he is?	He thinks highly of himself.	2	Captured explicit target entity but fails to note broader abuse (traitor or anti-national) in context
`ImpGab`	I said it, and I stand by it! Blacks get almost all of their cultural cues and information from rap and sports. It's not like they're reading books or anything. URL	Black people don't read books. Black people are less intellectual	5	Target is identified correctly. An alternate implication can be: Black people are illiterate.
`ImpGab`	You missed out on his post saying there are still many Jews on #Gab! And didn't see his followers cuss out kike?! Good that you didn't see he claims himself a "racist capitalist"! I'm glad you missed ALL of them so you can continue to support him. AND excuse me, but he doesn't have much intelligence based on some of his posts. I'm not kike & not stand w/ any sides!	Christian are anti-Jews	3	Could be referring to a specific right-winged politician. Alternative meaning can be: You are as racist and anti-semitic as the leaders you follow.

4.07 ± 1.41 for `ImpGab`. Table 3 lists some sample annotations and their agreement scores. Further, a third expert (a 24-year-old male) conducts an independent survey using the above metric on the other set of random 30 samples. As per annotator C, for `AbuseEval`, we obtain a mean agreement of 4.55 ± 1.09 and 4.41 ± 1.15 for `ImpGab`. This independent assessment corroborates the annotation process, as annotator C did not participate in the initial annotations yet observed similar alignment scores.

Generalizibility testing

We further consider three SemEval tasks for our generalizability analysis. Sarcasm detection (Abu Farha *et al.* 2022) and irony detection (Van Hee *et al.* 2018) are two-way classification datasets. Meanwhile, stance detection (Mohammad *et al.* 2016) is a three-way classification. While we have implied annotations for sarcasm, they are missing for the other two datasets. Here, no additional annotations are performed.

Hyperparameters

We run all experiments on two Nvidia V100 GPUs. Three random seeds (1, 4, 7) are used per setup. We report each setup's best performance based on overall macro-F1 out of three random seeds, where the best seed for a setup may vary. We follow an 80-20 split for the dataset across experiments (specific to the seed). In initial experiments, we observe that $ADD^{inf+foc}$ has a stronger influence on the later iterations, whereas CE influences the initial ones. Thus, to balance

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

them throughout the training process, we put equal weightage on both using $\beta = 0.5$. We consider $K = 3$ with $M = 2$ imposters for all experiments. We leave the experiments for β and M for future work. We set 100 as the maximum K-means iterations in each training step. During finetuning, each training cycle is executed for a maximum of 5000 epochs with all layers of PLM frozen.

PLMs

We begin our assessment with BERT (Devlin *et al.* 2019). For hate speech detection, we also employ a domain-specific HateBERT (Caselli *et al.* 2021a) model to establish generalizability beyond BERT embedding. HateBERT is built upon the concepts of continued pretraining on top of BERT. Here, the corpus for performing another round of unsupervised masking language modeling is obtained from potentially offensive subreddits. For the SemEval tasks, we consider BERT and XLM (Chi *et al.* 2022) for evaluation based on their popularity in the SemEval. The PLMs variants are "bert-base-uncased" for BERT, "xlm-roberta-large" for XLM, and "GroNLP/hateBERT" for HateBERT.

Baselines

First, we assess the improvement in the performance of ADD^{foc} over vanilla ADD (Equation 4) without the influence of cross-entropy. We follow the same prediction setup adopted in ADD (Rippel *et al.* 2016), where a sample gets assigned the label based on the nearest cluster in trained latent space during inference. We choose a simple Long short-term memory (LSTM) based (Hochreiter and Schmidhuber, 1997) model for quicker experimentation and compare the original ADD formulation with class weighted ADD (α-ADD) and our proposed ADD^{foc}. Table 4(a) shows a significant performance improvement of 8.2-10.8% in overall macro-F1 using ADD^{foc} across all three hate datasets. We thus recommend using our ADD^{foc} variant instead of vanilla ADD for future works. Interestingly, we note that α-ADD does not outperform ADD^{foc}. Hence, it is not employed in further experiments. Further, we perform a three-way classification using BERT to compare standalone alpha cross-entropy (ACE) against standalone ADD^{foc}. The results are presented in Table 4(b). We observe that ACE outperforms the standalone ADD^{foc} by a substantial margin of 7.6%, 1.9%, and 2.6% for `LatentHatred`, `ImpGab`, and `AbuseEval`, respectively. Based on the above two experiments, we employ ACE as our baseline. As the proposed model introduces an additional loss complimenting ACE, we thus use ACE+ADD^{foc} variants as comparative systems.

6. Results and abaltions

In this section, we enlist the performance of `FiADD` for classifying implicit hate and discuss its robustness under different tasks and ablation setups. In both two- and three-way hate classifications, clustering is performed w.r.t. the three classes; however, the CH is determined by the specific setup, either two or three-way. For two-way hate classification, explicit (EXP) and implicit (IMP) labels are consolidated under the *Hate* class.

Two-way hate classification

From Table 5, we note that `FiADD` variants improve overall macro-F1 by 0.58 (↑ 0.83%), 2.47 (↑ 3.68%), and 0.56 (↑ 0.79%) in `LatentHatred`, `ImpGab`, and `AbuseEval`, respectively, using BERT. However, except for maximizing hate macro-F1, the inferential objective does not significantly impact the final macro-F1 in the case of a two-way classification. It can be explained by

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

Table 4. Baseline selection based on comparison of ADD^{foc} over: (a) vanilla ADD for two-way hate speech classification via LSTM. (b) ACE for three-way hate speech classification via BERT

(a)					(b)			
		LSTM						
Dataset	**Metric (F1)**	ADD	*α-ADD*	ADD^{foc}	**Dataset**	**Metric (F1)**	ADD^{foc}	*ACE*
LatentHatred	Macro	0.557	0.601	**0.665**	LatentHatred	Macro	0.457	**0.533**
	N-Hate	0.646	0.695	0.782		N-Hate	0.690	0.772
	Hate	0.469	0.509	0.629		EXP	0.221	0.268
						IMP	0.462	0.558
ImpGab	Macro	0.554	0.566	**0.639**	ImpGab	Macro	0.451	**0.470**
	N-Hate	0.919	0.911	0.927		N-Hate	0.926	0.924
	Hate	0.189	0.222	0.351		EXP	0.347	0.390
						IMP	0.080	0.097
AbuseEval	Macro	0.552	0.554	**0.634**	AbuseEval	Macro	0.504	**0.530**
	N-Hate	0.820	0.780	0.851		N-Hate	0.817	0.854
	Hate	0.284	0.327	0.417		EXP	0.491	0.503
						IMP	0.203	0.233

Table 5. Results for two-way hate classification on BERT and HateBERT. We also highlight the highest Hate class macro-F1 that the respective model can achieve

		BERT			HateBERT		
Dataset	Metric (F1)	**ACE**	**$ACE+ADD^{foc}$**	**$ACE+ADD^{inf+foc}$**	**ACE**	**$ACE+ADD^{foc}$**	**$ACE+ADD^{inf+foc}$**
LatentHatred	Macro	0.6991	**0.7049**	0.7039	0.7121	**0.7135**	0.7121
	N-Hate	0.7599	0.7603	0.7638	0.7843	0.7787	0.7791
	Hate	0.6383	**0.6495**	0.6439	0.6400	**0.6484**	0.6450
	Highest Hate F1	0.6478	0.6491	**0.6548**	0.6615	0.6572	**0.6630**
ImpGab	Macro	0.6709	0.6886	**0.6956**	0.6889	**0.7027**	0.6987
	N-Hate	0.9171	0.9313	0.9355	0.9239	0.9398	0.9344
	Hate	0.4247	0.4459	**0.4556**	0.4538	**0.4657**	0.4631
	Highest Hate F1	0.4281	0.4494	**0.4562**	0.4538	0.4663	**0.4631**
AbuseEval	Macro	0.7075	**0.7131**	0.7124	0.7189	0.7176	**0.7202**
	N-Hate	0.8729	0.8779	0.8675	0.8767	0.8711	0.8808
	Hate	0.5420	0.5483	**0.5574**	0.5610	**0.5641**	0.5595
	Highest Hate F1	0.5502	0.5583	**0.5625**	0.5642	**0.5710**	0.5691

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

Table 6. Results for three-way hate classification on BERT and HateBERT

		BERT			HateBERT		
Dataset	Metric (F1)	**ACE**	**ACE+ADD^{foc}**	**ACE+ADD$^{inf+foc}$**	**ACE**	**ACE+ADD^{foc}**	**ACE+ADD$^{inf+foc}$**
LatentHatred	Macro	0.5327	0.5331	**0.5336**	0.5537	**0.5607**	0.5571
	N-Hate	0.7722	0.7609	0.7474	0.7480	0.7654	0.7829
	EXP	0.2682	0.2775	0.2776	0.3403	0.3300	0.3111
	IMP	0.5577	0.5610	**0.5759**	0.5728	**0.5866**	0.5774
ImpGab	Macro	0.4621	0.4575	**0.4668**	0.4797	0.4772	**0.4813**
	N-Hate	0.9198	0.9197	0.9258	0.9230	0.9434	0.9339
	EXP	0.3775	0.3850	0.3819	0.4055	0.4076	0.4190
	IMP	0.0889	0.0678	**0.0928**	**0.1105**	0.0806	0.0909
AbuseEval	Macro	0.5300	0.5328	**0.5398**	0.5309	0.5311	**0.5313**
	N-Hate	0.8541	0.8370	0.8867	0.8532	0.8611	0.8606
	EXP	0.5027	0.5256	0.5234	0.5038	0.5138	0.5111
	IMP	0.2333	**0.2359**	0.2094	**0.2357**	0.2185	0.2222

the partially conflicting objectives between the final two-way result and $ADD^{inf+foc}$'s three-way objective, leading to higher misclassification.

Three-way hate classification

Inferential infusion reasonably impacts the outcome of the three-way classification task (Table 6). Overall, in three-way classification, $ADD^{inf+foc}$ provide an improvement of 0.09 (↑ 0.17%), 0.47 (↑ 1.02%), and 0.98 (↑ 1.85%) in macro-F1 for LatentHatred, ImpGab, and AbuseEval, respectively, on BERT. *It is noteworthy that we observe an even higher level of improvement for implicit hate class than overall.* Compared to ACE in three-way classification, ADD^{foc} helps AbuseEval with an improvement of 0.26 macro-F1 (↑ 1.11%) in implicit hate. Meanwhile, $ADD^{inf+foc}$ helps LatentHatred and ImpGab with an improvement of 1.82 (↑ 3.26%) and 0.39 (↑ 4.39%), macro-F1, respectively, in implicit hate.

Generalizability test

The availability of implied annotations in the sarcasm dataset enables us to test FiADD's $ADD^{Inf+foc}$ variant. The unavailability of such annotations in the other two tasks limits our experiments with only ADD^{foc} variant. Table 7 (a) and (b) present the results for sarcasm detection and the other two tasks (irony and stance detection), respectively. Barring one setup, we observe reasonable improvements in macro-F1 (0.41–2.37) across all three tasks using both PLMs. Further, for the minority class considering the best of the BERT and XLM, we observe FiADD variants report an improvement of 6.06 (↑ 23.96)%, 1.35 (↑ 2.65%), and 3.14 (↑ 5.42%) for the respective minority class in sarcasm, stance, and irony detection.

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

Table 7. Comparative performance for sarcasm, irony, and stance detection

(a)					(b)				
Tasks	**Model**	**Metric (F1)**	**Objective**		**Tasks**	**Model**	**Metric (F1)**	**Objective**	
			ACE	**ACE+ADD$^{inf+foc}$**				**ACE**	**ACE+ADD^{foc}**
Sarcasm	BERT	Macro	**0.5651**	0.5610	Stance	BERT	Macro	0.5735	**0.5776**
		N-SAR	0.8329	0.8501			NEU	0.5009	0.4910
		SAR	0.2974	0.2720			ANG	0.6964	0.7090
							FAV	0.5231	0.5327
	XLM	Macro	0.5577	**0.5814**		XLM	Macro	0.5560	**0.5692**
		N-SAR	0.8626	0.8493			NEU	0.4494	0.4637
		SAR	0.2529	0.3135			ANG	0.7084	0.7199
							FAV	0.5103	0.5238
					Irony	BERT	Macro	0.6886	**0.7099**
							N-IRO	0.7379	0.7658
							IRO	0.6394	0.6540
						XLM	Macro	0.6637	**0.6740**
							N-IRO	0.7482	0.7372
							IRO	0.5792	0.6106

Impact of domain-specific PLM

Under HateBERT, `FiADD` variants improve two-way classification by overall 0.14 (↑ 0.20%), 1.38 (↑ 2.00%), and 0.13 (↑ 0.18%) for `LatentHatred`, `ImpGab`, and `AbuseEval`, respectively. Similarly, `FiADD` variants improve three-way classification by overall 0.7 (↑ 1.26%), 0.16 (↑ 0.34%), 0.04 (↑ 0.08%) for `LatentHatred`, `ImpGab`, and `AbuseEval`, respectively. However, the results with HateBERT show more variability. While all datasets in two-classification via HateBERT benefit from `FiADD` implicitness of `AbuseEval`, and `ImpGab` suffers under three-way classification. This variation can be attributed to a lot more offensive and slur terms in HateBERT's training than BERT. Through this analysis, we are able to comment on the domain-specific (HateBERT) vs general-purpose (BERT) systems and their role in finetuning. Interestingly, this has been noted in other research in hate speech as well (Masud *et al.* 2024a).

On the other hand, under generalization testing, which utilizes only general-purpose encoders (BERT and XLM), a high-performance improvement is observed in all minority classes.

Significance of hyperparameters

We further experiment with the hyperparameters of the `FiADD`. The experiments are performed on a two-way hate classification task on the `AbuseEval` dataset using BERT. The limited range for the probe is heuristically defined based on the sample size of categories. We recommend determining the values on a case-to-case basis for optimized performance. Figure 3(a) represents the significance of the number of subclusters per class (k) in the range of [2-4]. We observe comparable performance for $K = 3$ or 4. For our experiments, since four of the six datasets contain three classes, we use $K = 3$. The intuition is that within a subcluster of a class, the three subclusters represent a case of one of them having a high affinity to the class itself and two others being closer

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

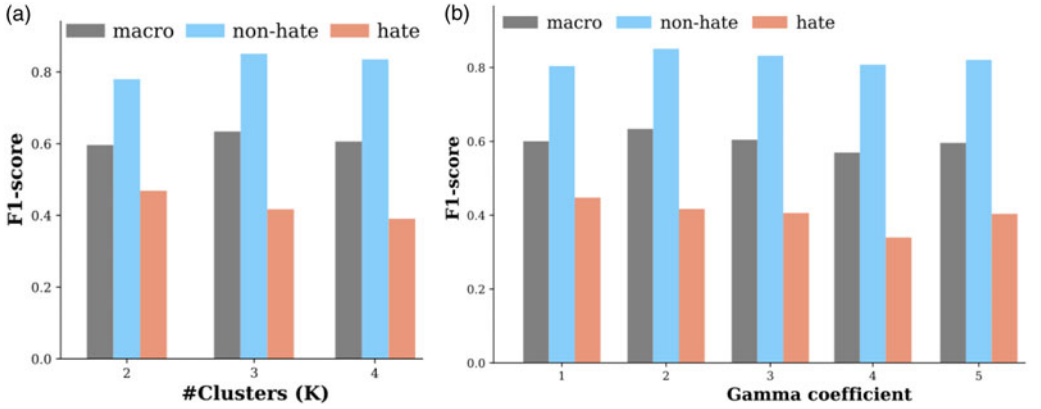


Figure 3. The variation in performance with changing values of (a) number of clusters (k) and (b) focal parameter (γ). We employ BERT on `AbuseEval` with ADD^{foc} in the two-way classification.

to their imposter classes. For example, within the implicit hate class, we assume at least one sub-cluster is easy to label as implicit, while there will likely be at least one cluster each that is closer to explicit and non-hate classes. Consequently, the setup leads to an imposter cluster value of $M = 2$. Meanwhile, the significance of the γ coefficient used in the focused objective is presented in Figure 3(b). The probe is limited to [1-5] with a unit interval as followed in existing literature (Lin *et al.* 2017). We observe the best outcome with $\gamma = 2$, which incidentally aligns with the best value identified by (Lin *et al.* 2017).

7. Does `FiADD` really improve implicit hate detection?

Given the overall macro-F1 results on hate speech detection vary in a narrow range, significance testing will be inconclusive. We thus perform a granular analysis of the results across all seeds and assess how well `FiADD` modifies the latent space. We also conduct an error analysis of cases where implicit hate is easy and hard to classify.

Seed-wise analysis

Across three random seeds, two PLMs, and three datasets, we record the performance for 18 setups, each in two-way and three-way hate speech detection. We note from Tables 8 and 9 that out of the 36 combinations, only four instances register a drop in performance. It corroborates that `FiADD`'s improvements are not limited to a specific initialization setup. Interestingly, the setups that register failure are all under HateBERT. The results further contribute to the discussion on domain-specific PLMs in Section 6.

Error analysis

The motivation for `FiADD` is that implicit is closer to non-hate than explicit hate. Employing `FiADD` should correct the misclassified implicit labels if this hypothesis holds. On the other hand, a false positive may occur if the example is already close to explicit subspace. Further, moving it toward explicit space can cause misclassification. We, thus, consider a positive/negative case where the predicted label for an implicit sample is correctly/incorrectly classified. To explain these two scenarios, we estimate the relative distance of the implicit sample from explicit and non-hate clusters. First, we perform K-means clustering on non-hate and explicit latent space to identify

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

Table 8. Results for two-way classification task across all three hate-speech datasets using two pretrained language models, BERT and HateBERT. Highlighted with green color are the outcomes where one of the variants of FiADD outperforms baseline ACE

			BERT			HateBERT		
Dataset	Seed	Metric (F1)	**ACE**	**ACE+ADD^{foc}**	**ACE+ADD$^{inf+foc}$**	**ACE**	**ACE+ADD^{foc}**	**ACE+ADD$^{inf+foc}$**
LatentHatred	1	Macro	0.69658	0.70291	0.69780	0.70843	0.69983	0.70143
		N-Hate	0.76353	0.76010	0.76069	0.77514	0.75034	0.76016
		Hate	0.62962	0.64571	0.63492	0.64171	0.64931	0.64271
	4	Macro	0.69388	0.68951	0.69611	0.70720	0.70423	0.70709
		N-Hate	0.75595	0.75246	0.77109	0.76370	0.75511	0.76389
		Hate	0.63182	0.62656	0.62113	0.65070	0.65335	0.65030
	7	Macro	0.69907	0.70490	0.70387	0.71212	0.71353	0.71205
		N-Hate	0.75987	0.76033	0.76384	0.78429	0.77868	0.77907
		Hate	0.63827	0.64946	0.64389	0.63995	0.64838	0.64502
ImpGab	1	Macro	0.67087	0.68862	0.69557	0.68886	0.69790	0.69872
		N-Hate	0.91708	0.93134	0.93551	0.92391	0.92951	0.93438
		Hate	0.42467	0.44590	0.45563	0.45380	0.46630	0.46307
	4	Macro	0.66411	0.67941	0.67591	0.68248	0.67406	0.68307
		N-Hate	0.91361	0.92403	0.91952	0.92069	0.91105	0.92318
		Hate	0.41461	0.43478	0.43231	0.44428	0.43708	0.44295
	7	Macro	0.66402	0.68237	0.68206	0.68720	0.70271	0.68962
		N-Hate	0.91019	0.92933	0.92782	0.92266	0.93976	0.93049
		Hate	0.41784	0.43541	0.43630	0.45175	0.46565	0.44874
AbuseEval	1	Macro	0.69253	0.70627	0.70502	0.70633	0.71030	0.70699
		N-Hate	0.85193	0.86709	0.87395	0.87774	0.87652	0.88362
		Hate	0.53313	0.54545	0.53609	0.53492	0.54409	0.53035
	4	Macro	0.70748	0.71313	0.71244	0.71885	0.71761	0.72018
		N-Hate	0.87293	0.87792	0.86754	0.87667	0.87112	0.88082
		Hate	0.54204	0.54833	0.55735	0.56103	0.56410	0.55953
	7	Macro	0.67794	0.68587	0.68944	0.69350	0.70146	0.69910
		N-Hate	0.85393	0.86020	0.85694	0.85240	0.85052	0.86129
		Hate	0.67794	0.68587	0.68944	0.69350	0.70146	0.69910

their centers. We then calculate the average Manhattan distance between the implicit samples and these local density centers. Finally, we obtain the relative score from explicit space by normalizing between 0 and 1 the average explicit distance by the sum of average distances from non-hate and explicit spaces. For example, if the sample has a distance of 3 from explicit and 6 from non-hate centers, then the normalized distance will be $3/(3+6) = 0.33$.

We highlight a positive and a negative case in Figures 4 (a) and (b), respectively. In the positive case, the implicit sample is closer to non-hate space (Point A) under the ACE objective. After employing the FiADD, its relative position moves away from non-hate and closer to explicit

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

Table 9. Results for three-way classification tasks across all three hate-speech datasets using two pretrained language models, BERT and HateBERT. Highlighted with green color are the outcomes where one of the variants of FiADD outperforms baseline ACE

			BERT			HateBERT		
Dataset	Seed	Metric (F1)	**ACE**	**ACE+ADD^{foc}**	**ACE+ADD$^{inf+foc}$**	**ACE**	**ACE+ADD^{foc}**	**ACE+ADD$^{inf+foc}$**
LatentHatred	1	Macro	0.50683	0.51872	0.51727	0.53277	0.53023	0.52951
		N-Hate	0.76214	0.76201	0.74872	0.76216	0.75547	0.76122
		EXP	0.20159	0.25344	0.25296	0.25575	0.26912	0.26109
		IMP	0.55678	0.54072	0.55013	0.58041	0.56609	0.56622
	4	Macro	0.52340	0.52746	0.52819	0.55372	0.56065	0.54937
		N-Hate	0.77192	0.75288	0.76089	0.74803	0.76535	0.75695
		EXP	0.26631	0.25862	0.28870	0.34032	0.33000	0.32524
		IMP	0.53196	0.57088	0.53499	0.57281	0.58660	0.56591
	7	Macro	0.53267	0.53312	0.53361	0.54265	0.54197	0.55713
		N-Hate	0.77215	0.76094	0.74736	0.76769	0.77002	0.78290
		EXP	0.26815	0.27748	0.27762	0.29891	0.28428	0.31111
		IMP	0.55772	0.56095	0.57587	0.56136	0.57162	0.57738
ImpGab	1	Macro	0.46284	0.45740	0.47635	0.47448	0.47417	0.47325
		N-Hate	0.90876	0.91715	0.92309	0.92432	0.92950	0.91882
		EXP	0.36140	0.38783	0.37983	0.39767	0.40897	0.40336
		IMP	0.11834	0.06722	0.12612	0.10144	0.08403	0.09756
	4	Macro	0.45289	0.45772	0.43606	0.46788	0.47415	0.44959
		N-Hate	0.92661	0.91054	0.91243	0.92351	0.92469	0.91982
		EXP	0.38078	0.36646	0.37195	0.38563	0.42696	0.40542
		IMP	0.05128	0.09615	0.02380	0.09448	0.07079	0.02352
	7	Macro	0.47048	0.45741	0.46351	0.47969	0.47721	0.48128
		N-Hate	0.92416	0.93149	0.93998	0.92302	0.94337	0.93388
		EXP	0.39033	0.40073	0.38159	0.40554	0.40761	0.41904
		IMP	0.09696	0.04000	0.06896	0.11049	0.08064	0.09090
AbuseEval	1	Macro	0.51761	0.52082	0.52249	0.52462	0.52638	0.52559
		N-Hate	0.84885	0.84862	0.849518	0.86772	0.87497	0.87748
		EXP	0.49009	0.51483	0.51033	0.53555	0.52680	0.52727
		IMP	0.21390	0.19900	0.20765	0.17058	0.17737	0.17204
	4	Macro	0.53002	0.53283	0.53982	0.52609	0.52095	0.52764
		N-Hate	0.85409	0.83699	0.88670	0.86536	0.86110	0.87867
		EXP	0.50272	0.52564	0.52338	0.52133	0.50485	0.51759
		IMP	0.23325	0.23587	0.20938	0.19158	0.19689	0.18666
	7	Macro	0.51914	0.52951	0.52534	0.53091	0.53110	0.53129
		N-Hate	0.84480	0.85799	0.85942	0.85322	0.86106	0.86060
		EXP	0.50420	0.51487	0.51041	0.50383	0.51378	0.51106
		IMP	0.20843	0.21568	0.20618	0.23569	0.21848	0.22222

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

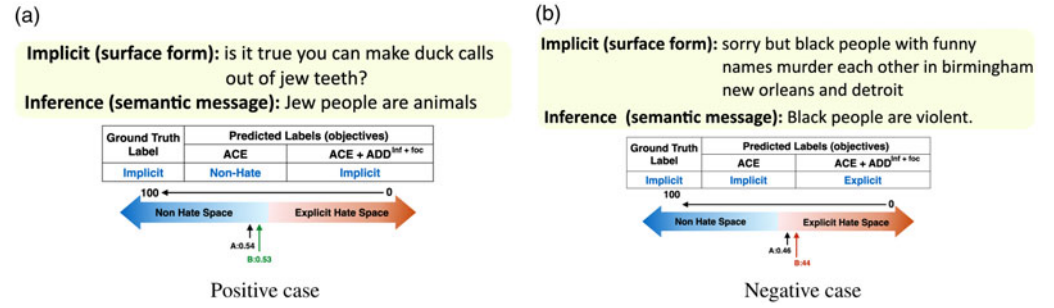


Figure 4. Error analysis with (a) correctly and (b) incorrectly classified samples in three-way classification on `LatentHatred`. Here, scores A and B are the relative positions of implicit sample w.r.t non-hate and explicit space finetuned with ACE and $ADD^{inf+foc}$, respectively.

(point B). In contrast, for the negative case, where the implicit sample is initially close to explicit hate (point A), our objective leads to misclassification. In the future, this problem can be reduced by introducing a constraint on the distance between implicit and explicit intact.

7.1 Latent space analysis

Building upon the cluster assessment in the error analysis, where we examined only a single positive and negative sample, we now perform an overall evaluation of how $ADD^{inf+foc}$ manipulates the embedding space. Inspired by the existing literature examining the latent space under hate speech datasets (Fortuna, Soler, and Wanner, 2020) and models (Kim *et al.* 2022; Ocampo *et al.* 2023c), we attempt to quantify the intercluster separation via Silhouette scores.

Silhouette score

It is a metric to measure the "goodness" of the clustering technique. It is calculated as a trade-off between within-cluster similarity and intercluster dissimilarity. Consider a system with E points (e_i), each point belonging to one of the N clusters $c^j(e_i)$. For $e_i \in c^a$, its Silhouette score $SS_i = \frac{max(p_i,q_i)}{q_i - p_i}$. p_i captures the intra-cluster distance of e_i to all the points within the cluster it belongs; $p_i = \frac{1}{|c^a|-1} \sum_{e^j \in c^a} dist(e_i, e_j)$. q_i captures the intercluster distance of $e_i \in c^a$ to all the points in the nearest cluster to c^a; $q_i = \frac{1}{|c^b|} \sum_{e^j \in c^b} dist(e_i, e_j)$. The Silhouette score of a setup is, thus, $SS = \frac{1}{|E|} \sum_{e_i \in E} SS_i$. Silhouette scores are measured on a scale of -1 to 1, with -1 being the worst set of cluster assignments.

Subclustering objective

After applying the $ADD^{inf+foc}$ objective, we expect not only the per-class clusters to be sufficiently separated but also the subclusters in each class to be better segregated to match their local neighborhood better. Figure 5 shows the implicit embedding space of `AbuseEval`, `ImpGab`, and `LatentHatred` after applying K-means on the default BERT embedding (a, d, g), BERT finetuned with the ACE (b, e, h), and `FiADD` (c, f, i) on three-way hate classification. The higher the Silhouette score, the better the subclusters are separated. 0.34, 0.31, and 0.51 are the scores for cases (a), (b), and (c), respectively, in `AbuseEval`. 0.38, 0.24, and 0.52 are the scores for cases (d), (e), and (f), respectively, in `ImpGab`. 0.32, 0.29, and 0.32 are the scores for cases (d), (e), and (f), respectively, in `LatentHatred`.

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

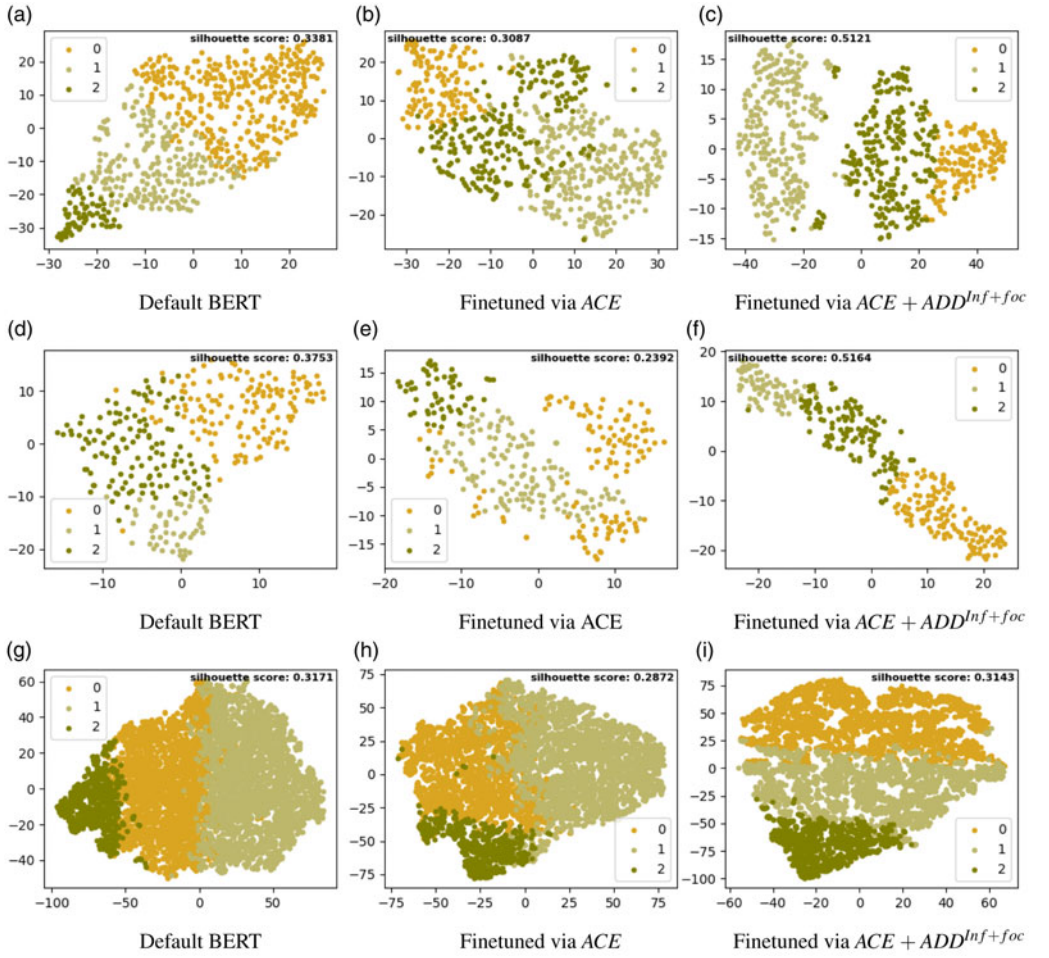


Figure 5. 2D t-SNE plots of the last hidden representations after applying K-means (K = 3) on the implicit class for `AbuseEval` (a, b, c), `ImpGab` (d, e, f), and `LatentHatred` (g, h, i). {0, 1, 2} are the subcluster ids. The higher the Silhouette score, the better discriminated the clusters.

Consequently, an increase of 0.20, 0.28, and 0.03 scores is observed when comparing `FiADD` with ACE for `AbuseEval`, `ImpGab`, and `LatentHatred`, respectively. This increase in scores validates that the local densities within a class get further refined under $ADD^{inf+foc}$ objective. As expected, ACE suboptimally treats the implicit class as a single homogeneous cluster. Interestingly, for `LatentHatred` the score does not improve over the default BERT, even though it improves over ACE. A deeper analysis with multiple K values might help here.

Inferential infusion

Given that $ADD^{inf+foc}$ brings the surface and semantic forms of implicit hate closer, we expect a significant drop in Silhouette scores between these clusters under `FiADD`. Figure 6 visualizes the embedding space of default BERT (a, d, g), BERT finetuned with the ACE (b, e, h), and `FiADD` (c, f, i) on three-way classification for `AbuseEval`, `ImpGab`, and `LatentHatred`. 0.18, 0.18, and 0.03 are the scores for cases (a), (b), and (c), respectively, in `AbuseEval`. 0.18, 0.23, and 0.07 are the scores for cases (d), (e), and (f), respectively, in `ImpGab`. 0.14, 0.13, and 0.01 are the scores for

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

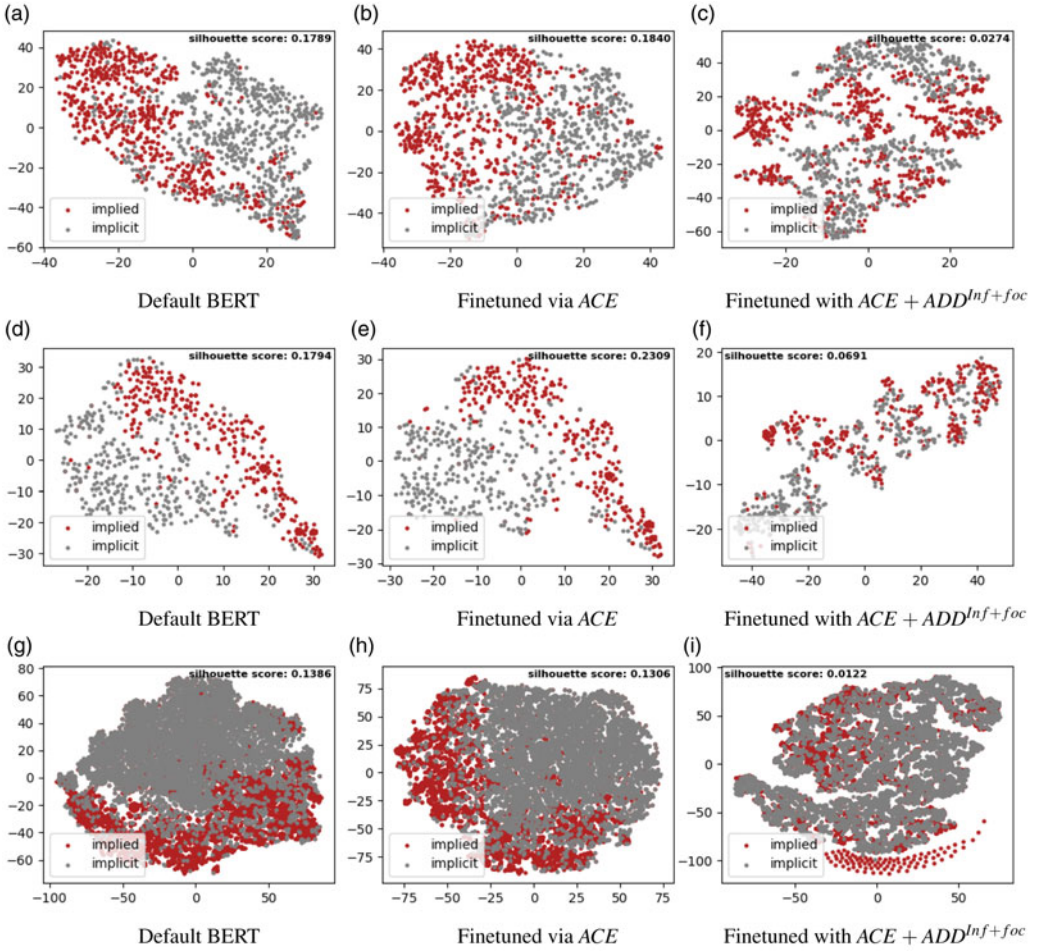


Figure 6. 2D t-SNE plots of the last hidden representations obtained for the implicit class and its respective inferential (implied) set for `AbuseEval` (a, b, c), `ImpGab` (d, e, f), and `LatentHatred` (g, h, i). The lower the Silhouette score, the closer the surface and implied forms of hate.

cases (g), (h), and (i), respectively, in `LatentHatred`. It is important to highlight that for both BERT and BERT + ACE, there is no explicit objective to bring the implicit and implied clusters together. Hence, they act as a baseline for comparing how well the $ADD^{inf+foc}$ objective brings the two spaces closer.

A drop of 0.15, 0.16, and 0.12 in the Silhouette score is observed when comparing BERT + ACE with `FiADD` for `AbuseEval`, `ImpGab`, and `LatentHatred`, respectively. It corroborates that the implicit and implied meaning representations are brought significantly closer to each other by employing our model. In addition to Tables 5 and 6, the latent space analysis also quantifies our manual annotations for `AbuseEval` and `ImpGab`, as inferential infusion (supported by the manual annotations) is improving the detection of implicit hate.

8. Conclusion

An increase in hate speech on the Web has necessitated the involvement of automated hate speech detection systems. To this end, we do not recommend completely removing human moderators;

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

instead, we recommend employing machine learning-based systems to perform the first level of filtering. Following the rise of PLMs for text classification, they are now defacto for hate speech detection, too. However, PLM-based systems still suffer from understanding nuanced concepts, such as implicitness, and require external contextualization.

To this end, FiADD presents a generalized framework for semantic classification tasks in which the surface form of the source text differs from its inference form. For any system modeling this setup, the aim is to bring the two embedding spaces closer. In this work, the objective is achieved by optimizing for adaptive density discrimination via inferential infusion. Clustering accounts for variation in local neighborhoods beyond a single sample or a single positive/negative pairing; the inferential infusion assures that while we look into the local neighborhood, the implicit clusters are mapped to the apt semantic latent spaces. Further, this work introduces the focal penalty that pays more attention to the sample near the classification boundary. Even by itself, the ADD^{foc} objective provides a considerable improvement over a standard loss function and can be applied as a substitute.

Overall, our inferential-infused focal $ADD^{inf+foc}$ provides a novel augmentation to the PLM finetuning pipeline. The efficacy of the FiADD's variants is analyzed over three implicit hate detection datasets (with two of them being manually annotated by us for inferential context), three implicit semantic tasks (sarcasm, irony, and stance detection), and three PLMs (BERT, HateBERT, and XLM). By design, the $ADD^{inf+foc}$ objective does help improve the detection of hate in both two-way and three-way classifications. Our results call into question the role of domain-specific models like HateBERT against BERT as we observe that once finetuned, both of them perform comparably. It calls into question the role of domain-specific models in NLP.

A more granular examination of FiADD over the latent space for hate speech detection is performed via the analysis of—seed-wise performance measurement, latent space analysis of the embedding space clusters, and error analysis of positive and negative use cases. Over multiple seeds and 36 experimental setups, we observe the FiADD variants improve over ACE in 32 instances. Meanwhile, a closer look at the later space further highlights the significant improvement that FiADD has on the implicit clusters in bringing them near their implied meaning.

9. Limitations and future work

First, the current setup utilizes manual annotations of implicit meaning to be available for inferential clustering, requiring manual effort. Second, the proposed setup, being a novel approach in the direction of implicit detection, works on the de facto K-means and uses the same number of subclusters for all datasets.

In the future, we expect an infusion of generative models to pseudo-annotate the implied meaning, which can be paraphrased and rectified by human annotators on a need basis. Further, the proposed setup can be employed as an external loss to nudge the LLMs to generate better-quality adversarial examples. Meanwhile, to overcome performing K-means on the entire training set after each epoch, consider representations only for the given batch, starting with stratified sampling so that the batch is representative of the overall dataset. Recent advancements in hashing and dictionary techniques can improve computational efficiency. In the future, we aim to make the system more computationally efficient and extend its application to other tasks. It would be fascinating to review how focal infusion impacts the classification tasks in computer vision in comparison to the ADD setup.

Ethical concerns. This work focuses on textual features and does not incorporate personally identifiable or user-specific signals. For annotations, the annotators were sensitized about the task at hand and given sufficient compensation for their expert involvement. The annotators worked on $\approx$ 250 samples per day over four days to avoid feeling fatigued. Further, the annotators had access to the Web; while annotating, they referred to multiple news sources to understand the context. The dataset of inferential statements for AbuseEval and ImpGab will be available to researchers on request.

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

Acknowledgments. Sarah Masud acknowledges the support of the Prime Minister Doctoral Fellowship in association with Wipro AI and Google India PhD Fellowship. Tanmoy Chakraborty acknowledges the financial support of Anusandhan National Research Foundation (CRG/2023/001351) and Rajiv Khemani Young Faculty Chair Professorship in Artificial Intelligence.

References

Abu Farha I., **Oprea S. V.**, **Wilson S. and Magdy W.** (2022). SemEval-2022 task 6: iSarcasmEval, intended sarcasm detection in English and Arabic. In Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), Seattle, United States. Association for Computational Linguistics, pp. 802–814.

Badjatiya P., **Gupta S.**, **Gupta M. and Varma V.** (2017). Deep learning for hate speech detection in tweets. In *WWW*, pp. 759–760.

Balayn A., **Yang J.**, **Szlavik Z. and Bozzon A.** (2021). Automatic identification of harmful, aggressive, abusive, and offensive language on the web: A survey of technical biases informed by psychology literature. *ACM Transactions on Social Computing* **4**, 1–56.

Baucum M., **Cui J. and John R. S.** (2020). Temporal and geospatial gradients of fear and anger in social media responses to terrorism. *ACM Transactions on Social Computing* **2**, 1–16.

Botelho A., **Hale S. and Vidgen B.** (2021). Deciphering implicit hate: Evaluating automated detection algorithms for multimodal hate. In *Findings of the Association for Computational Linguistics*. Online. Association for Computational Linguistics. 1896–1907.

Caselli T., **Basile V.**, **Mitrović J. and Granitzer M.** (2021a). HateBERT: Retraining BERT for abusive language detection in English. In Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021), Online. Association for Computational Linguistics, pp. 17–25.

Caselli T., **Basile V.**, **Mitrović J.**, **Kartoziya I. and Granitzer M.** (2020). I feel offended, don't be abusive! implicit/explicit messages in offensive and abusive language. In Proceedings of the Twelfth Language Resources and Evaluation Conference, Marseille, France: European Language Resources Association, pp. 6193–6202.

Caselli T., **Schelhaas A.**, **Weultjes M.**, **Leistra F.**, **van der Veen H.**, **Timmerman G. and Nissim M.** (2021b). DALC: the Dutch abusive language corpus. In Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021), Online. Association for Computational Linguistics, pp. 54–66.

Chen W., **Chen X.**, **Zhang J. and Huang K.** (2017). Beyond triplet loss: A deep quadruplet network for person re-identification. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Los Alamitos, CA, USA: IEEE Computer Society, pp. 1320–1329.

Chi Z., **Huang S.**, **Dong L.**, **Ma S.**, **Zheng B.**, **Singhal S.**, **Bajaj P.**, **Song X.**, **Mao X.-L.**, **Huang H. and Wei F.** (2022). XLM-E: Cross-lingual language model pre-training via ELECTRA. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Dublin, Ireland: Association for Computational Linguistics, pp. 6170–6182.

Chopra S., **Hadsell R. and LeCun Y.** (2005). Learning a similarity metric discriminatively, with application to face verification. In 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. **1**, pp. 539–546.

Cilingir H. K., **Manzelli R. and Kulis B.** (2020). Deep divergence learning. In Proceedings of the 37th International Conference on Machine Learning, PMLR, vol. **119** of Proceedings of Machine Learning Research, pp. 2027–2037.

Davidson T., **Warmsley D.**, **Macy M. and Weber I.** (2017). Automated hate speech detection and the problem of offensive language. *Proceedings of the International AAAI Conference On Web and Social Media* **11**, 512–515.

Deng J., **Guo J.**, **Xue N. and Zafeiriou S.** (2019). Arcface: Additive angular margin loss for deep face recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).

Devlin J., **Chang M.-W.**, **Lee K. and Toutanova K.** (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, Minnesota: Association for Computational Linguistics, vol. **1** (Long and Short Papers), pp. 4171–4186.

Diaz M., **Amironesei R.**, **Weidinger L. and Gabriel I.** (2022). Accounting for offensive speech as a practice of resistance. In Proceedings of the Sixth Workshop on Online Abuse and Harms (WOAH), Seattle, Washington (Hybrid): Association for Computational Linguistics, pp. 192–202.

ElSherief M., **Ziems C.**, **Muchlinski D.**, **Anupindi V.**, **Seybolt J.**, **Choudhury M. D. and Yang D.** (2021). Latent hatred: a benchmark for understanding implicit hate speech. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp. 345–363.

Fortuna P., **Soler J. and Wanner L.** (2020). Toxic, hateful, offensive or abusive? what are we really classifying? an empirical analysis of hate speech datasets. In Proceedings of the Twelfth Language Resources and Evaluation Conference, Marseille, France: European Language Resources Association, pp. 6786–6794.

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

Founta A., **Djouvas C.**, **Chatzakou D.**, **Leontiadis I.**, **Blackburn J.**, **Stringhini G.**, **Vakali A.**, **Sirivianos M. and Kourtellis N.** (2018). Large scale crowdsourcing and characterization of twitter abusive behavior. *Proceedings of the International AAAI Conference On Web and Social Media* **12**, 491–500.

Founta A. M., **Chatzakou D.**, **Kourtellis N.**, **Blackburn J.**, **Vakali A. and Leontiadis I.** (2019). A unified deep learning architecture for abuse detection. In *WebSci*, pp. 105–114.

Frenda S., **Patti V. and Rosso P.** (2023). When sarcasm hurts: Irony-aware models for abusive language detection. In Experimental IR Meets Multilinguality, Multimodality, and Interaction: 14th International Conference of the CLEF Association, CLEF. 2023, Thessaloniki, Greece, Berlin, Heidelberg: Springer-Verlag, pp. 34–47.

Garg T., **Masud S.**, **Suresh T. and Chakraborty T.** (2023). Handling bias in toxic speech detection: A survey. *ACM Computing Surveys* **55**, 1–32.

Ghosh K. and Senapati D. A. (2022). Hate speech detection: a comparison of mono and multilingual transformer model with cross-language evaluation. In Proceedings of the 36th Pacific Asia Conference on Language, Information and Computation, Manila, Philippines: De La Salle University, pp. 853–865.

Ghosh S., **Suri M.**, **Chiniya P.**, **Tyagi U.**, **Kumar S. and Manocha D.** (2023). CoSyn: Detecting implicit hate speech in online conversations using a context synergized hyperbolic network. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Singapore: Association for Computational Linguistics, pp. 6159–6173.

Gomez R., **Gibert J.**, **Gomez L. and Karatzas D.** (2020). Exploring hate speech detection in multimodal publications. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pp. 1459–1467.

Hartvigsen T., **Gabriel S.**, **Palangi H.**, **Sap M.**, **Ray D. and Kamar E.** (2022). ToxiGen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Dublin, Ireland: Association for Computational Linguistics, pp. 3309–3326.

Hochreiter S. and Schmidhuber J. (1997). Long short-term memory. *Neural Computation* **9**, 1735–1780.

Kennedy B., **Atari M.**, **Davani A. M.**, **Yeh L.**, **Omrani A.**, **Kim Y.**, **Coombs K.**, **Havaldar S.**, **Portillo-Wightman G.**, **Gonzalez E.**, **Hoover J.**, **Azatian A.**, **Hussain A.**, **Lara A.**, **Cardenas G.**, **Omary A.**, **Park C.**, **Wang X.**, **Wijaya C.**, **Zhang Y.**, **Meyerowitz B. and Dehghani M.** (2022). Introducing the gab hate corpus: defining and applying hate-based rhetoric to social media posts at scale. *Language Resources and Evaluation* **56**, 79–108.

Kim Y., **Park S. and Han Y.-S.** (2022). Generalizable implicit hate speech detection using contrastive learning. In Proceedings of the 29th International Conference on Computational Linguistics, Gyeongju, Republic of Korea: International Committee on Computational Linguistics, 6667–6679.

Kim Y., **Park S.**, **Namgoong Y. and Han Y.-S.** (2023). ConPrompt: pre-training a language model with machine-generated data for implicit hate speech detection. In *Findings of the Association for Computational Linguistics: EMNLP*. Singapore: Association for Computational Linguistics, pp. 10964–10980.

Kirk H., **Yin W.**, **Vidgen B. and Röttger P.** (2023). SemEval-2023 task 10: Explainable detection of online sexism. In Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023), Toronto, Canada: Association for Computational Linguistics, pp. 2193–2210.

Kulkarni A., **Masud S.**, **Goyal V. and Chakraborty T.** (2023). Revisiting hate speech benchmarks: From data curation to system deployment. In Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD, New York, NY, USA: Association for Computing Machinery, vol **23**, pp. 4333–4345.

Labadie Tamayo R., **Chulvi B. and Rosso P.** (2023). Everybody hurts, sometimes overview of hurtful humour at iberlef 2023: Detection of humour spreading prejudice in twitter. Procesamiento del Lenguaje Natural (Spanish Society of Natural Language Processing), [Sl], vol. **71**, pp. 383–395.

Likert R. (1932). A technique for the measurement of attitudes. *Archives of Psychology* **22**, 55.

Lin J. (2022). Leveraging world knowledge in implicit hate speech detection. In Proceedings of the Second Workshop on NLP for Positive Impact (NLP4PI), Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics, pp. 31–39.

Lin T.-Y., **Goyal P.**, **Girshick R.**, **He K. and Dollár P.** (2017). Focal loss for dense object detection. In 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 2999–3007.

Liu W., **Wen Y.**, **Yu Z. and Yang M.** (2016). Large-margin softmax loss for convolutional neural networks. In Proceedings of The 33rd International Conference on Machine Learning, New York, New York, USA: PMLR, vol. **48** of Proceedings of Machine Learning Research, pp. 507–516.

Masud S., **Khan M. A.**, **Goyal V.**, **Akhtar M. S. and Chakraborty T.** (2024a). Probing critical learning dynamics of PLMs for hate speech detection. In *Findings of the Association for Computational Linguistics* EACL 2024, St. Julian's, Malta: Association for Computational Linguistics. 826–845.

Masud S., **Singh S.**, **Hangya V.**, **Fraser A. and Chakraborty T.** (2024b). Hate personified: investigating the role of llms in content moderation. In Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pp. 15847–15863.

Merlo L. I., **Chulvi B.**, **Ortega-Bueno R. and Rosso P.** (2023). When humour hurts: linguistic features to foster explainability. *Procesamiento Del Lenguaje Natural* **70**, 85–98.

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

Mohammad S., **Kiritchenko S.**, **Sobhani P.**, **Zhu X. and Cherry C.** (2016). SemEval-2016 task 6: Detecting stance in tweets. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016), San Diego, California: Association for Computational Linguistics, pp. 31–41.

Muralikumar M. D., **Yang Y. S. and McDonald D. W.** (2023). A human-centered evaluation of a toxicity detection api: testing transferability and unpacking latent attributes. *ACM Transactions on Social Computing* **6**, 1–38.

Nozza D. (2021). Exposing the limits of zero-shot cross-lingual hate speech detection. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), Online. Association for Computational Linguistics, pp. 907–914.

Ocampo N. B., **Cabrio E. and Villata S.** (2023a). Playing the part of the sharp bully: generating adversarial examples for implicit hate speech detection. In *Findings of the Association for Computational Linguistics: ACL*. Toronto, Canada: Association for Computational Linguistics, pp. 2758–2772.

Ocampo N. B., **Cabrio E. and Villata S.** (2023). Unmasking the hidden meaning: bridging implicit and explicit hate speech embedding representations. In *Findings of the Association for Computational Linguistics: EMNLP*. Singapore: Association for Computational Linguistics, pp. 6626–6637.

Ocampo N. B., **Sviridova E.**, **Cabrio E. and Villata S.** (2023c). An in-depth analysis of implicit and subtle hate speech messages. In Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, Dubrovnik, Croatia: Association for Computational Linguistics, pp. 1997–2013.

Plaza-del Arco F. M., **Nozza D. and Hovy D.** (2023). Respectful or toxic? using zero-shot learning with language models to detect hate speech. In *the 7th Workshop On Online Abuse and Harms (WOAH)*. Toronto, Canada: Association for Computational Linguistics, pp. 60–68.

Poletto F., **Basile V.**, **Sanguinetti M.**, **Bosco C. and Patti V.** (2021). Resources and benchmark corpora for hate speech detection: a systematic review. *Language Resources and Evaluation* **55**, 477–523.

Rippel O., **Paluri M.**, **Dollár P. and Bourdev L. D.** (2016). Metric learning with adaptive density discrimination. In 4th International Conference on Learning Representations, ICLR. 2016, San Juan, Puerto Rico: Conference Track Proceedings.

Rottger P., **Vidgen B.**, **Hovy D. and Pierrehumbert J.** (2022). Two contrasting data annotation paradigms for subjective NLP tasks. In Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Seattle, United States: Association for Computational Linguistics, pp. 175–190.

Röttger P., **Vidgen B.**, **Nguyen D.**, **Waseem Z.**, **Margetts H. and Pierrehumbert J.** (2021). HateCheck: Functional tests for hate speech detection models. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Online. Association for Computational Linguistics, pp. 41–58.

Schmidt A. and Wiegand M. (2017). A survey on hate speech detection using natural language processing. In Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media, Valencia, Spain: Association for Computational Linguistics, pp. 1–10.

Schroff F., **Kalenichenko D. and Philbin J.** (2015). Facenet: A unified embedding for face recognition and clustering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, pp. 815–823.

Silva L., **Mondal M.**, **Correa D.**, **Benevenuto F. and Weber I.** (2021). Analyzing the targets of hate in online social media. *Proceedings of the International AAAI Conference on Web and Social Media* **10**, 687–690.

Sánchez-Junquera J., **Chulvi B.**, **Rosso P. and Ponzetto S. P.** (2021). How do you speak about immigrants? taxonomy and stereoimmigrants dataset for identifying stereotypes about immigrants. *Applied Sciences,* **11**, 3610.

Snell J., **Swersky K. and Zemel R.** (2017). Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, **30**, Curran Associates, Inc.

Song H. O., **Jegelka S.**, **Rathod V. and Murphy K.** (2017). Deep metric learning via facility location. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2206–2214.

Suler J. (2004). The online disinhibition effect. Cyberpsychology & behavior : the impact of the internet. *multimedia and virtual reality on behavior and society* **73**, 321–326.

Tahmasbi N. and Rastegari E. (2018). A socio-contextual approach in automated detection of public cyberbullying on twitter. *ACM Transactions on Social Computing* **1**, 1–22.

Van Hee C., **Lefever E. and Hoste V.** (2018). SemEval-2018 task 3: Irony detection in English tweets. In Proceedings of the 12th International Workshop on Semantic Evaluation, New Orleans, Louisiana: Association for Computational Linguistics, pp. 39–50.

Vidgen B. and Derczynski L. (2020). Directions in abusive language training data, a systematic review: garbage in, garbage out. *PLOS ONE* **15**, e0243300.

Waseem Z. and Hovy D. (2016). Hateful symbols or hateful people? predictive features for hate speech detection on Twitter. In Proceedings of the NAACL Student Research Workshop, San Diego, California: Association for Computational Linguistics, pp. 88–93.

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press

Yadav N., **Masud S.**, **Goyal V.**, **Akhtar M. S. and Chakraborty T.** (2024). Tox-BART: Leveraging toxicity attributes for explanation generation of implicit hate speech. In Ku L.-W., Martins A. and Srikumar V., (eds), *Findings of the Association for Computational Linguistics*: ACL 2024. Bangkok, Thailand: Association for Computational Linguistics, pp. 13967–13983.

Yin W. and Zubiaga A. (2021). Towards generalisable hate speech detection: a review on obstacles and solutions. *PeerJ Computer Science* **7**, e598.

Cite this article: Masud S, Bajpai A and Chakraborty T (2025). Focal inferential infusion coupled with tractable density discrimination for implicit hate detection. *Natural Language Processing* **31**, 1323–1349. https://doi.org/10.1017/nlp.2024.60

https://doi.org/10.1017/nlp.2024.60 Published online by Cambridge University Press