ToKen: Task Decomposition and Knowledge Infusion for Few-Shot Hate Speech Detection

Badr AlKhamissi* **Faisal Ladhak***

Srini Iyer **Ves Stoyanov** **Zornitsa Kozareva** **Xian Li** **Pascale Fung**

Lambert Mathias **Asli Celikyilmaz** **Mona Diab**

Meta AI

Abstract

Hate speech detection is complex; it relies on commonsense reasoning, knowledge of stereotypes, and an understanding of social nuance that differs from one culture to the next. It is also difficult to collect a large-scale hate speech annotated dataset. In this work, we frame this problem as a few-shot learning task, and show significant gains with decomposing the task into its "constituent" parts. In addition, we see that infusing knowledge from reasoning datasets (e.g. ATOMIC$^{20}_{20}$) improves the performance even further. Moreover, we observe that the trained models generalize to out-of-distribution datasets, showing the superiority of task decomposition and knowledge infusion compared to previously used methods. Concretely, our method outperforms the baseline by 17.83% absolute gain in the 16-shot case.

1 Introduction

Disclaimer: *Due to the nature of this work, some examples contain offensive text and hate speech. This does not reflect authors' values, however our aim is to help detect and prevent the spread of such harmful content.*

The task of automatically detecting *Hate Speech* (HS) is becoming increasingly important given the rapid growth of social media platforms, and the severe social harms associated with the spread of hateful content. However, building good systems for automated HS detection is challenging due to the complex nature of the task. It requires the system to understand social nuance, such as which groups are being targeted by the hateful content. Prior work has shown that even humans cannot achieve a high agreement on whether or not a social media post constitutes HS (Rahman et al., 2021).

In this work, we explore whether decomposing HS detection into subtasks that correspond to the definitional criteria of what constitutes HS (i.e. the offensiveness of a post and whether it targets a group or an individual) (Davidson et al., 2017; Mollas et al., 2020a) would lead to systems that are more accurate and robust. In particular, we show that task decomposition leads to more sample-efficient systems for HS detection, by showing improved results in the few-shot setting. Moreover, we demonstrate that infusing commonsense knowledge by fine-tuning on the ATOMIC$^{20}_{20}$ (Hwang et al., 2021) and *StereoSet* (Nadeem et al., 2021) datasets improve performance even further. Specifically, we observe an absolute improvement of 17.83% in the 16-shot case over the baseline model (§5.2). Further, we show that the resulting models are more robust, and are able to achieve better performance than baseline methods in out-of-distribution settings (§5.3).

Explainability is an important aspect for being able to identify and fix failure modes of HS systems (Attanasio et al., 2022). To that end, we show that task decomposition of HS detection moves us a step closer towards explainable systems, allowing us to identify the problematic subtasks that may be

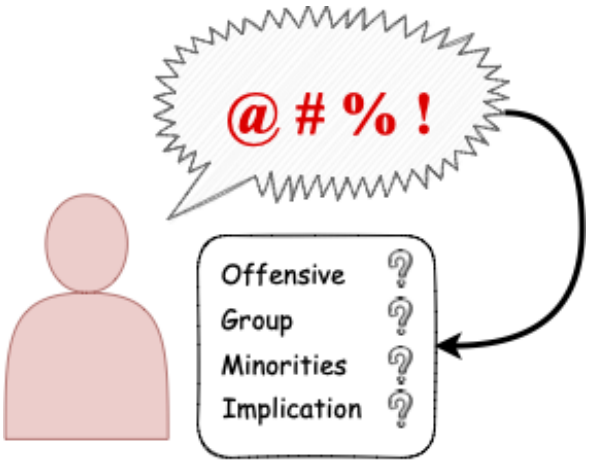


Figure 1: *Hate Speech* decomposed into several subtasks leading to better results.

* Equal Contribution

the bottleneck for improving overall performance for HS detection. We then show that explicitly targeting to improve the problematic subtask leads to improved overall performance (§5.1.1).

The remainder of the paper is structured along four axes. We first present the importance of (1) **Task Decomposition** and (2) **Knowledge Infusion** for training better few-shot HS detection models. Specifically, we compare our method against a baseline that only outputs a binary prediction for HS. Thus showing the significance of decomposing the prediction into several subtasks and pre-finetuning the model on two different reasoning datasets that instill in the model a degree of commonsense reasoning and knowledge of stereotypes. We evaluate our experiments across 10 seeds each of which uses a different sampled dataset with sizes ranging from 16 to 1024 samples. Our method, TOKEN, show significant improvements[1] over the baseline. The trained models can also (3) **Generalize** better to three out-of-distribution datasets, and are more (4) **Robust** to training data and hyperparameters. We demonstrate this by measuring the variance across different seeds, data partitions and hyperparameters and show that it is significantly smaller for the TOKEN models compared to the baseline.

2 Few-Shot Hate Speech Detection

Collecting a high quality large-scale dataset for HS is difficult since it is a relatively rare phenomenon, which makes it hard to sample social media posts containing HS without relying on keywords that may be indicative of it (Rahman et al., 2021). However, despite being rare, its effects are of significant harm. Further, since HS is a complex phenomenon, relying on keywords may result in datasets with low coverage that are not effective in capturing more subtle forms of HS (ElSherief et al., 2021). This further results in building models that are less generalizable and can exhibit racial biases (Davidson et al., 2019; Sap et al., 2019).

Motivated by these, we frame HS detection as a few-shot learning task, where the model is given a limited number of examples to learn what constitutes HS, and explore whether we can build robust HS detection models that can generalize well in cases where we do not have a lot of training data. In particular, we show that our trained TOKEN models are more *generalizable* by measuring the performance on out-of-distribution HS datasets, and are more *robust* by measuring the variance in performance of HS detection across different randomly sampled few-shot datasets and hyperparameters.

[1]We use Welch's t-test on all experiments that were repeated 10 times, and consider $p < 0.05$ as significant.

3 Datasets

SBIC (Sap et al., 2020). We use the Social Bias Inference Corpus (SBIC) to construct few-shot training and validation sets. This corpus includes posts from several online social media platforms, such as Reddit, Twitter, etc., along with the annotations for the offensiveness, targeted group as well as the implications to further explain what stereotype is being implied by the post. While the dataset does not have explicit labels for whether or not a post is HS, we derive it using the annotations for offensiveness and group detection, i.e. a post is considered HS if it contains offensive/derogatory language that is expressed towards a targeted group (see Figure 2). This is consistent with the definition of HS used by prior work (Davidson et al., 2017; Mollas et al., 2020a).

To construct few-shot training sets, we perform a stratified sampling of data from the SBIC corpus, up to a target size n. We sample $\frac{n}{4}$ examples containing inoffensive posts, $\frac{n}{4}$ examples containing offensive, but non-HS posts. The remaining budget of $\frac{n}{2}$ samples is used for posts containing HS, spread evenly across different targeted groups to ensure diversity. We create datasets of varying sizes from 16 samples up to 1024, ensuring each smaller dataset is a proper subset of the larger datasets. We sample 10 different datasets for each target size n using 10 different random seeds.

3.1 Reasoning Datasets

ATOMIC$^{20}_{20}$ (Hwang et al., 2021) This is a commonsense knowledge graph containing 1.33M inferential knowledge tuples in textual format that are not readily available in pretrained LMs. ATOMIC$^{20}_{20}$ encodes different social and physical aspects of the everyday experience of human life. In this work, we find that training on human readable templates in-place of each tuple vastly improves the downstream SBIC few-shot HS performance. Examples of such templates are shown in the Appendix (§A.1).

StereoSet (Nadeem et al., 2021) This dataset was developed to measure stereotype bias in LMs. Specifically, it contains 17k sentences that measure

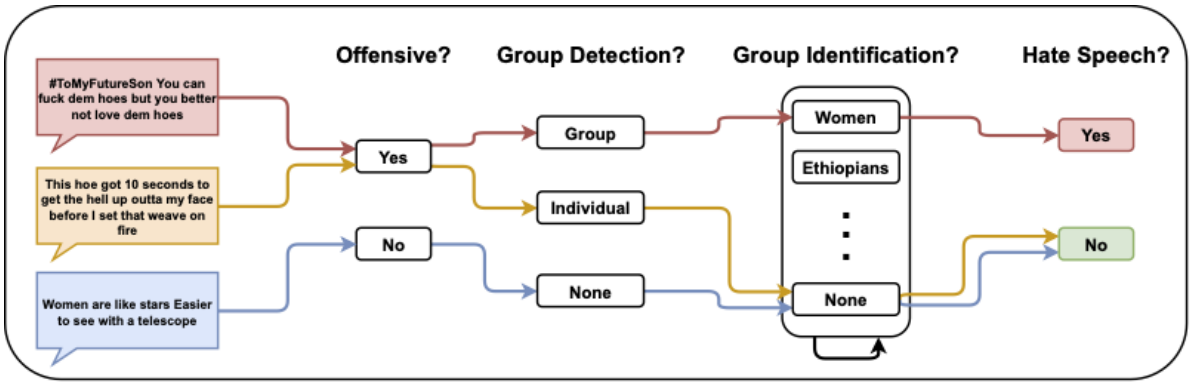


Figure 2: Examples from the SBIC dataset. The post is classified as HS if it is *Offensive* and a *Group* is referenced as the target of the offensive speech.

biases across four different domains: gender, profession, race and religion. In this work, we only finetune task-decomposed models on a subset of StereoSet. In particular we only use stereotypes that belong to the intersentence task since we found that it results in better HS detection models. More details on the StereoSet training can be found in the Appendix (§A.2).

3.2 Out-of-Distribution Datasets

In addition, to test the generalizability of our models, we use the following three corpora to evaluate out-of-distribution performance:

HateXplain (Mathew et al., 2021). This corpus includes posts from Twitter and Gab along with the HS labels (i.e. hate, offensive or normal), target community (i.e. victim of the HS or offensive speech) and the rationales (i.e. spans from the posts that affected the annotator's decision). In our work, we convert the HS labels into a binary HS/non-HS label, to measure performance for HS detection.

HS18 (de Gibert et al., 2018). This corpus consists of sentences from posts on Stormfront, a white supremacist forum, along with labels for HS.

Ethos (Mollas et al., 2020b). This corpus (compiled by researchers at the Aristotle University of Thessaloniki) consists of comments from social media platforms (YouTube and Reddit) with binary labels for HS, as well as a finer-grained categorization of the type of HS. We use the binary labels in our work to measure the performance for HS detection. Number of examples in test set of each evaluation dataset is shown in Table 1.

Dataset	# Examples
HateXplain	1,924
HS18	9,916
Ethos	998

Table 1: Evaluation dataset statistics.

4 Experimental Setup

4.1 Task Decomposition

HS detection is a complex and subjective task, and prior work has shown that it is hard to get high agreements between humans about whether or not a post constitutes HS (Sanguinetti et al., 2018; Assimakopoulos et al., 2020b). Therefore recent efforts on hate speech annotation have turned to more fine-grained, hierarchical annotation schemes that break HS detection into subtasks that correspond to the definitional criteria of what constitutes HS, leading to higher agreement scores than reported by prior work (Assimakopoulos et al., 2020b; Mathew et al., 2021; Sap et al., 2020; Rahman et al., 2021).

Motivated by these findings, we treat HS detection as a conditional generation task, since that allows us to represent classification and generation subtasks in a unified framework. The model is given the set of tokens in a post as input and is tasked with generating the inferences related to the HS subtasks. Table 2 shows the linearization scheme that we use to train our models. The baseline model is tasked with generating a binary prediction for HS, whereas the task-decomposed model has to generate the predictions for the subtasks, in order, before making the prediction for HS. Specifically, the model predicts first if the post is offensive, then whether it is targeting a group or an individual or neither, and following that it

	Input	Output
Baseline	Post: {POST} Hate speech?	{HS}
TOKEN	Post: {POST} Offensive?	{OFF} Target implication? {GD} Targeted minorities? {GI_1, ..., GI_N} Hate speech? {HS}

Table 2: Linearization scheme for the **Baseline** and the **TOKEN** models. Given the post, the *Baseline* predicts whether it is HS or not; whereas the task-decomposed model does the prediction for *Offensiveness*, *Group Detection* and *Group Identification* before predicting the HS label. HS and OFF are binary labels (i.e. either Yes or No). GD can be one of {Group, Individual, None}. Finally, GI_I is a group identity (e.g. Women).

predicts the identities of the targeted groups (e.g. disabled people). This forces the task decomposed model to reason about the subtasks before deciding whether or not a post constitutes HS. For all sets of experiments, we finetune the pre-trained $\text{BART}_{\text{LARGE}}$ model (Lewis et al., 2020) provided by the HuggingFace library (Wolf et al., 2019) for the task of HS detection. The results are shown in Table 3.

4.2 Knowledge Infusion

In the following set of experiments we ask whether incorporating commonsense knowledge and stereotypes into the model show significant improvements on the few-shot HS detection task.

ATOMIC$^{20}_{20}$ To answer this question, we first finetune BART on the ATOMIC$^{20}_{20}$ dataset, where each tuple is converted into a natural language statement using human readable templates (see §A.1). The resulting model achieves similar performance on the held-out ATOMIC$^{20}_{20}$ test set as the one reported in Hwang et al. (2021). Following that, we further finetune the resulting model on both the baseline and the task decomposed data from SBIC as described in §4.1. Results are shown in Table 4.

StereoSet Here, we finetune both BART and our model finetuned on ATOMIC$^{20}_{20}$ on stereotypical sentences from the StereoSet dataset. Specifically, we similarly treat it as a conditional generation task, where we predict the context based on the sentence, bias type and target group (see Appendix §A.2).

5 Results

In this section, we report the mean binary F1-scores on the testing set across 10 different seeds for each dataset size for the HS detection task. Specifically, we compare the **Baseline** model, which directly predicts a binary label of whether the post is HS or not, with the **TOKEN** model that employ task decomposition and knowledge infusion as described in the previous section.

5.1 Task Decomposition

Table 3 shows the effect of task decomposition on the HS detection task. In particular, we compare the baseline model with a model that uses only the *Offensiveness* and *Group Detection* tasks as it's subtasks before predicting the HS label. This is referred to as the **Minimal Decomposition** model since it uses the minimal constituents that we used to derive the HS label. Following that, we add the **Group Identification** to the subtasks and observe significant improvements over the *Baseline* in the few-shot setting. However, as the dataset size increases beyond 512 samples, the observed differences in the mean are no longer significant (see Figure 3c).

5.1.1 Fine-Grained Error Analysis

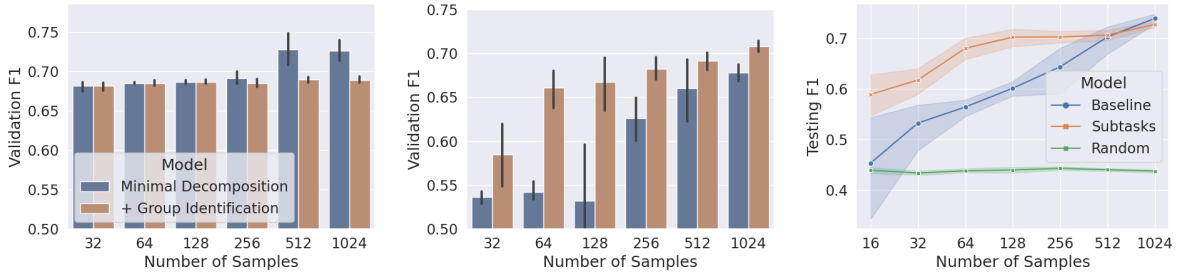
Task decomposition allows us to perform finer grained error analysis to identify failure modes of the HS model. Specifically, we analyze whether *Offensiveness* classification or *Group Detection* is more challenging for the model to learn. Figure 3a shows the performance of the model for the *Offensiveness* subtask, while Figure 3b shows the performance for *Group Detection* subtask with varying number of training samples.

We observe that the overall performance for *Group Detection* subtask consistently lags behind *Offensiveness* prediction, especially when we have fewer examples in the training data. We note that the model is able to achieve a reasonable performance ($\sim 68\%$) for *Offensiveness* prediction even in the few-shot regime. This suggests that *Group Detection* subtask is the bottleneck in improving the performance for HS classification, and in order to improve further we need our model to be more accurate for this subtask.

Given these findings, we further explore whether

Model	16	32	64	128	256	512	1024
Baseline	45.31	53.23	56.41	60.12	64.37	70.29	**73.95**
Minimal Decomposition	50.79	56.12	56.46	59.78	64.94	67.83	69.95
+ Group Identification	**58.89**	**61.77**	**68.03**	**70.25**	**70.28**	**70.65**	72.76

Table 3: Results of the Task Decomposed Model on the *Hate Speech* detection task (Binary F1-score). **Baseline** predicts only whether the input post is HS or not. **Minimal Decomposition** additionally predicts whether the post is offensive or not and the group detection. **+ Group Identification** additionally predicts the minority groups the post is targeting if any.



(a) Offensiveness Performance (b) Group Detection Performance (c) Hate Speech Performance

Figure 3: **(a)** The validation F1-score of the *Offensiveness* subtask for the minimal decomposition and task-decomposed models. **(b)** Similarly, this is the validation F1-score for the *Group Detection* subtask. It can be seen that adding the *Group Identification* subtask improves the performance dramatically. **(c)** Testing Performance of *Baseline* vs the *Task Decomposed* model. Random performance plotted for reference.

adding more fine-grained information related to groups would help improve the *Group Detection* subtask. In particular, we additionally require the task-decomposed model to generate the group that is being targeted by the offensive content in the post. Figure 3b shows the performance of the model in predicting *Group Detection* when we require the subtask model to identify the groups that are being targeted. We see that incorporating this subtask significantly improves the performance for *Group Detection* in the few-shot setting. Figure 3c shows that this further translates to improved HS detection in the same regime.

5.2 Knowledge Infusion

Table 4 show the binary F1-scores across 10 seeds on the SBIC testing-set for the HS detection task using different knowledge infused models for both the baseline and task-decomposed datasets (referred as Subtasks). The first row show the BART model without any knowledge infusion (same as the one reported in Table 3). The following row show the results when we finetune the pre-trained BART on StereoSet. It can be seen that this result is the best performance in the 32-shot regime. In the third row we finetune BART on the natural language version of the ATOMIC$^{20}_{20}$ dataset. This increases performance most noticeably in the 16-shot regime. The final row shows the results when we further finetune the model finetuned on ATOMIC$^{20}_{20}$ on StereoSet, it consistently improves the performance even further from the 64-shot setting to the 512-shot setting. In all models, the difference between the baseline and subtasks model is significant in most cases until we reach 512 training examples. It can be seen that knowledge infusion alone does not seem to consistently improve performance over the BART baseline model, however when combined with subtask decomposition, it leads to the best results overall. This would imply that the reasoning knowledge helps the model to better understand relationships among subtasks.

5.3 Generalizability

Figure 4 compares the OOD performance of the baseline model with subtasks models that were trained with different degrees of knowledge infusion. Note that all models were trained on the SBIC data and evaluated for each of the three datasets in a zero-shot manner, i.e. we do not perform any further dataset specific finetuning. Similar to above, each point represents the mean F1 scores across

Model		16	32	64	128	256	512	1024
BART	**Baseline**	45.31	53.23	56.41	60.12	64.37	70.29	73.95
	Subtasks	58.89	61.77	68.03	70.25	70.28	70.65	72.76
+ StereoSet	**Baseline**	53.30	54.68	54.17	61.41	67.69	71.25	73.68
	Subtasks	42.86	**66.17**	69.01	70.06	70.14	72.14	72.64
+ ATOMIC$^{20}_{20}$	**Baseline**	44.76	49.60	64.89	69.38	70.09	72.32	**73.97**
	Subtasks	**63.14**	62.01	67.96	70.94	70.16	72.29	72.96
+ StereoSet	**Baseline**	44.75	47.47	56.18	62.88	66.38	69.77	71.50
	Subtasks	59.74	63.28	**70.08**	**70.99**	**70.57**	**72.36**	73.80

Table 4: **Knowledge Infusion Results** Here we report binary F1-score on the SBIC testing set for the HS detection task using models with different degrees of knowledge infusion. In each row we compare the corresponding baseline and subtasks models. Results in **bold** show the best overall model in each few-shot setting. See Section 5.2 for more details.

10 different runs for the given dataset size. We see that for all three datasets, the TOKEN models consistently outperforms the baseline, and shows significantly better zero-shot generalizability in the few-shot setting. However, we note though that HateXplain has a slight distribution shift (since it's also built based on Twitter as SBIC), and interestingly it can be seen that only the results from Ethos and HS18 (in Figure 4) are significantly different from the binary (baseline) prediction model. This implies that our model is indeed better at generalizing to out-of-distribution data.

5.4 Robustness

Here, we show that the TOKEN models are less sensitive to changes in hyperparameters and training dataset selection. Specifically, Table 5 show the average standard deviation across different hyperparameters on the validation set as a function of the number of training samples. It can be seen that the subtasks model is significantly more robust than the baseline model.

6 Analysis

In this section, we explore whether the order of the subtasks matter or not, the scale of the model and if adding an additional subtask that requires the model to generate natural language explanations as to why the post is considered HS helps improve the overall HS performance. The results reported here were done across only 5 seeds. In addition, we show a few examples that the baseline model was not able to capture but the TOKEN model got correctly.

# of Samples	Baseline	TOKEN
16	15.75	2.44
32	14.15	0.55
64	13.71	0.87
128	16.44	1.66
256	21.20	2.69
512	17.29	3.36
1024	15.72	3.20

Table 5: **Robustness Results** Here we report the average standard deviation across the 10 runs for a given training set size across the different hyperparameters used in our experiments. It can be seen that the task-decomposed model is more robust to training configuration.

6.1 Does Order Matter?

Table 6 shows the importance of the order of the HS label in the sequence of subtasks. It can be seen that placing HS at the end gets the best result when the number of training samples are 64 or less.

6.2 Do Implications Matter?

Here, we add an additional subtask with the goal of generating an implied stereotype in natural language for a HS post or 'None' otherwise. For example, given the following post: "*How do you make a Muslim's phone explode? Set it to airplane mode.*" The implied stereotype would be: `Muslims are terrorists.` Therefore, the model is tasked with generating such a sentence before predicting the HS label. Table 7 shows the HS detection performance before and after adding the implications subtask in comparison with the

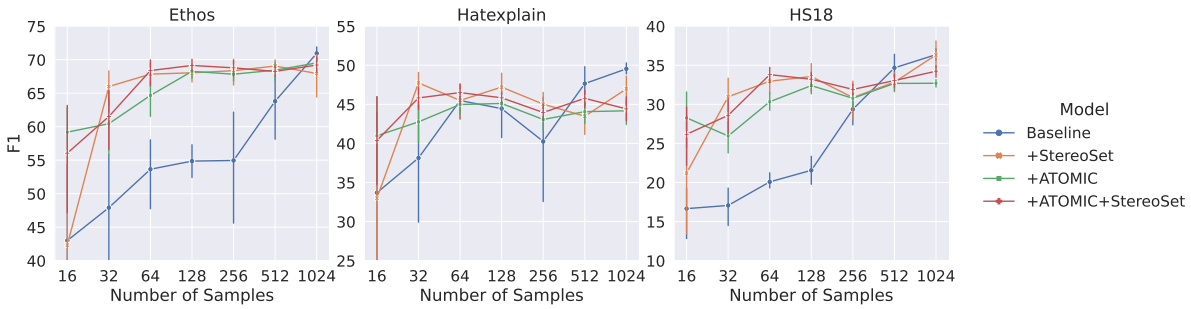


Figure 4: **OOD Results** Performance of Baseline versus TOKEN models that employ different degrees of knowledge infusion on OOD datasets HS18, Ethos, Hatexplain.

Order	16	32	64	128	256	512	1024
OFF GD GI **HS**	**55.60**	**62.31**	**68.47**	69.22	69.64	70.49	71.69
OFF GD **HS** GI	54.67	60.36	68.02	67.65	68.66	70.13	70.92
OFF **HS** GD GI	54.53	56.14	67.02	68.59	**71.37**	**72.47**	**72.39**
HS OFF GD GI	51.64	62.04	64.48	**69.45**	70.33	71.33	72.20
GD GI **OFF** HS	38.28	27.11	31.32	53.98	52.12	60.69	67.20

Table 6: The validation performance of the best model on the HS detection task as a function of the position of the HS label in the sequence of subtasks across different number of training samples.

# of Samples	Baseline	TOKEN	+Impl
16	52.67	58.21	57.02
32	52.71	64.47	59.98
64	57.60	70.93	65.50
128	60.01	71.25	67.89
256	66.81	71.62	69.22
512	69.41	72.59	70.12
1024	74.72	74.09	71.45

Table 7: **Implications Results** The HS detection performance of the **Baseline** in comparison with the **Subtasks** models before and after adding the implication to the subtasks across 5 runs for a given training set size.

baseline across 5 seeds for a given training set size. It can be seen that although adding the implied stereotype to the list of subtasks pushes the model to performing better than the baseline when the number of samples is less than 512, it still falls short to the TOKEN model without the implication. The reason behind this might be because the implications were noisy and sometimes too generic, which is why it might have resulted in performance degradation. Further, we believe that scaling up the model to be an order or two magnitude larger will enable a better utilization of the implications.

6.3 Does Scale Matter?

We train $\text{BART}_{\text{BASE}}$ using the same task decomposition and knowledge infusion methods reported earlier. We find that the results do not fully transfer to smaller models. Specifically, the results are only better when the training examples are 16 to 64, otherwise the baseline model surpasses the TOKEN model. Showing that scale does matter. The results are shown in Table 9 in the Appendix (§A.3).

6.4 Error Analysis

Figure 5 shows the percentage of false positives (FPs) and false negatives (FNs) across 10 runs on the testing set for the 16 to 128-shot case. It can be seen that the difference in performance between both models is largely due to a considerable decrease in the percentage of FPs for the TOKEN model as we increase the number of samples, while the percentage of FNs stay almost constant. However, the percentage of FNs are always less than the FPs, which is a desirable property for HS models as FPs can be more tolerated (i.e. over enforcement) than FNs (i.e. under enforcement), since the latter will lead to more propagation of hateful content. As discussed in Section 5.1.1, predicting the group identity in the TOKEN model enables the model to correctly detect whether the post is targeting a

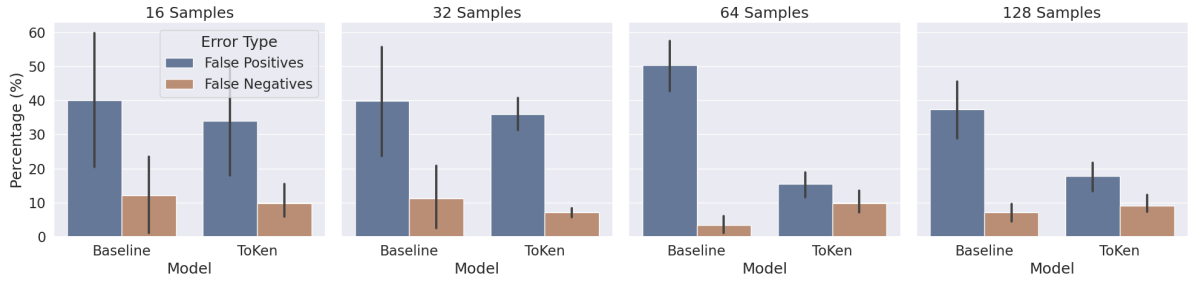


Figure 5: The percentage of *False Positives* and *False Negatives* across 10 runs on the testing set.

group or an individual, and we can see that clearly in posts where the baseline model got a FN and the TOKEN model got the correct result. For example, the baseline model wasn't able to predict this post as HS: "*Your approval is so worthless you should pay people to take it @user Call me sexist but I do not approve of female football presenters*", while the subtasks model identified correctly that it targets `Women` leading to a correct prediction. On the other hand, this is an example that the baseline predicted correctly while the TOKEN model did not: "*What do you get when an epileptic falls into a garden? A seizure salad.*" The reason for this is the TOKEN model predicted that this post targets an `Individual` and not a `Group`.

7 Related Work

Social media provides a platform for users to connect with people all over the world and engage in ways that were not previously possible. Recent surveys show that 41% of internet users experienced some form of harassment online, with a third of these cases being identity-related (i.e. race, gender, sexual orientation, etc.) (Vogels, 2021; League, 2020). The sheer scale of content shared on social media platforms makes manual moderation untenable and necessitates automated methods for detecting hateful content (Halevy et al., 2022). This has led to an increased interest in automated hate speech detection, both in terms of collecting corpora (Poletto et al., 2021) as well as improved methods for hate speech detection (Schmidt and Wiegand, 2017; Fortuna and Nunes, 2018).

Early work in hate speech detection has treated the problem as a binary classification task, requiring annotators to simply indicate whether or not a given post constitutes hate speech (Waseem and Hovy, 2016; Davidson et al., 2017; Founta et al., 2018). However, recent work has shown that eliciting binary judgments for hate speech is unreliable and leads to poor inter-annotator agreement (Sanguinetti et al., 2018; Assimakopoulos et al., 2020a). This has lead to increased work in collecting hate speech annotation with more complex annotation schemas. Zampieri et al. (2019) propose a three-level annotation schema that identifies both the type and target of offensiveness in social media posts. Another line of work proposes a hierarchical annotation schema where the task of determining hate speech is broken down into subtasks, in an effort to eliminate some of the subjectivity (Assimakopoulos et al., 2020a; Sap et al., 2020). Rahman et al. (2021) combine established information retrieval techniques with task decomposition and annotator rationale, in order to create a higher quality dataset for hate speech detection. While the aforementioned studies explore the idea of task decomposition in improving annotation consistency, our work instead looks at the role of task decomposition in building more robust, generalizable models for few-shot hate speech detection.

The focus of the limited prior work on few-shot hate speech detection has been to explore zero-shot/few-shot crosslingual transfer from a source language (such as English) with sufficient hate speech data to a target language with limited data (Stappen et al., 2020; Nozza, 2021). In contrast, our work explores how task decomposition and knowledge infusion can help even when there is not sufficient hate speech data in English.

8 Conclusion

In this work, we propose TOKEN , a method to train language models for detecting HS in the few-shot setting. We show that it significantly outperforms comparable baseline models that predicts the HS label directly instead of decomposing it into its constituent parts. We further show that task decomposition not only improves the performance, but also allows for fine-grained inspection of the model's

behavior. Since HS is a complex phenomenon that requires a set of reasoning skills not readily available in such pre-trained models, we pre-finetune the $\text{BART}_{\text{LARGE}}$ on both the ATOMIC^{20}_{20} and StereoSet datasets to equip the model with commonsense reasoning and knowledge of stereotypes that we show leads to further improvement in HS detection performance. We show that the TOKEN models generalize better to three out-of-distribution datasets in the few-shot setting as well as being significantly more robust to training setups. We further analyze the model's behavior in terms of the order in which the HS labels appears, the scale of the model and the performance when adding an additional subtask that explains the implied stereotype of the post.

In future work, we plan on investigating the role of task decomposition, knowledge infusion and the additional subtask of explaining the implication behind the post in large language models as well as explore the TOKEN method in low-resource languages, where it is expected to be most beneficial.

9 Limitations

We note a few limitations of our work: (1) in our experiments we compared our task-decomposed model to standard models as baselines. It will be valuable for future work to compare our models against other models of similar scale trained using multi-task learning in a similar manner to (AlKhamissi and Diab, 2022), where each classification head is subtask-specific and trained using categorical cross-entropy on the corresponding number of classes. However, that would require categorizing the group identities into a discrete number of classes. (2) To the best of our knowledge there is no literature that uses the SBIC dataset in a few-shot hate speech setting, therefore we resorted to the baseline with binary prediction using the same conditional generation framework. Future work should compare with such models. (3) The datasets used in this work are mostly looking at HS from a western perspective and are only in English. Different languages and societies may have subtleties which may affect the performance of HS systems. Even though we believe that our work is generalizable beyond the English language, we have not evaluated this, and we encourage future work to look beyond the settings we explored in this paper. (4) We follow prior work and determine hate-speech labels based on majority vote which might silence the voice of minority groups, which is especially problematic in this context. In the future we hope to model dissenting opinions between the annotators similar to recent work (Gordon et al., 2022).

References

Badr AlKhamissi and Mona T. Diab. 2022. Meta ai at arabic hate speech 2022: Multitask learning with self-correction for hate speech classification. *ArXiv*, abs/2205.07960.

Stavros Assimakopoulos, Rebecca Vella Muskat, Lonneke van der Plas, and Albert Gatt. 2020a. Annotating for hate speech: The maneco corpus and some input from critical discourse analysis. *arXiv preprint arXiv:2008.06222*.

Stavros Assimakopoulos, Rebecca Vella Muskat, Lonneke van der Plas, and Albert Gatt. 2020b. Annotating for hate speech: The MaNeCo corpus and some input from critical discourse analysis. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 5088–5097, Marseille, France. European Language Resources Association.

Giuseppe Attanasio, Debora Nozza, Eliana Pastor, and Dirk Hovy. 2022. Benchmarking post-hoc interpretability approaches for transformer-based misogyny detection. In *Proceedings of NLP Power! The First Workshop on Efficient Benchmarking in NLP*, pages 100–112, Dublin, Ireland. Association for Computational Linguistics.

Thomas Davidson, Debasmita Bhattacharya, and Ingmar Weber. 2019. Racial bias in hate speech and abusive language detection datasets. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 25–35, Florence, Italy. Association for Computational Linguistics.

Thomas Davidson, Dana Warmsley, Michael W. Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *ICWSM*.

Ona de Gibert, Naiara Perez, Aitor García-Pablos, and Montse Cuadros. 2018. Hate Speech Dataset from a White Supremacy Forum. In *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*, pages 11–20, Brussels, Belgium. Association for Computational Linguistics.

Mai ElSherief, Caleb Ziems, David Muchlinski, Vaishnavi Anupindi, Jordyn Seybolt, Munmun De Choudhury, and Diyi Yang. 2021. Latent hatred: A benchmark for understanding implicit hate speech. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 345–363, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Paula Fortuna and Sérgio Nunes. 2018. A survey on automatic detection of hate speech in text. *ACM Computing Surveys (CSUR)*, 51(4):1–30.

Antigoni Maria Founta, Constantinos Djouvas, Despoina Chatzakou, Ilias Leontiadis, Jeremy Blackburn, Gianluca Stringhini, Athena Vakali, Michael Sirivianos, and Nicolas Kourtellis. 2018. Large scale crowdsourcing and characterization of twitter abusive behavior. In *Twelfth International AAAI Conference on Web and Social Media*.

Mitchell L Gordon, Michelle S Lam, Joon Sung Park, Kayur Patel, Jeff Hancock, Tatsunori Hashimoto, and Michael S Bernstein. 2022. Jury learning: Integrating dissenting voices into machine learning models. In *CHI Conference on Human Factors in Computing Systems*, pages 1–19.

Alon Halevy, Cristian Canton-Ferrer, Hao Ma, Umut Ozertem, Patrick Pantel, Marzieh Saeidi, Fabrizio Silvestri, and Ves Stoyanov. 2022. Preserving integrity in online social networks. *Communications of the ACM*, 65(2):92–98.

Jena D. Hwang, Chandra Bhagavatula, Ronan Le Bras, Jeff Da, Keisuke Sakaguchi, Antoine Bosselut, and Yejin Choi. 2021. Comet-atomic 2020: On symbolic and neural commonsense knowledge graphs. In *AAAI*.

Anti-Defamation League. 2020. Online hate and harassment. the american experience 2021. *Center for Technology and Society. Retrieved from www. adl. org/media/14643/download*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Binny Mathew, Punyajoy Saha, Seid Muhie Yimam, Chris Biemann, Pawan Goyal, and Animesh Mukherjee. 2021. Hatexplain: A benchmark dataset for explainable hate speech detection. In *AAAI conference on artificial intelligence*.

Ioannis Mollas, Zoe Chrysopoulou, Stamatis Karlos, and Grigorios Tsoumakas. 2020a. Ethos: an online hate speech detection dataset. *ArXiv*, abs/2006.08328.

Ioannis Mollas, Zoe Chrysopoulou, Stamatis Karlos, and Grigorios Tsoumakas. 2020b. Ethos: an online hate speech detection dataset. *arXiv preprint arXiv:2006.08328*.

Moin Nadeem, Anna Bethke, and Siva Reddy. 2021. Stereoset: Measuring stereotypical bias in pretrained language models. In *ACL/IJCNLP*.

Debora Nozza. 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)*, pages 907–914, Online. Association for Computational Linguistics.

Fabio Poletto, Valerio Basile, Manuela Sanguinetti, Cristina Bosco, and Viviana Patti. 2021. Resources and benchmark corpora for hate speech detection: a systematic review. *Language Resources and Evaluation*, 55(2):477–523.

Md Mustafizur Rahman, Dinesh Balakrishnan, Dhiraj Murthy, Mucahid Kutlu, and Matthew Lease. 2021. An information retrieval approach to building datasets for hate speech detection. *arXiv preprint arXiv:2106.09775*.

Manuela Sanguinetti, Fabio Poletto, Cristina Bosco, Viviana Patti, and Marco Stranisci. 2018. An Italian Twitter corpus of hate speech against immigrants. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Maarten Sap, Dallas Card, Saadia Gabriel, Yejin Choi, and Noah A. Smith. 2019. The risk of racial bias in hate speech detection. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1668–1678, Florence, Italy. Association for Computational Linguistics.

Maarten Sap, Saadia Gabriel, Lianhui Qin, Dan Jurafsky, Noah A Smith, and Yejin Choi. 2020. Social bias frames: Reasoning about social and power implications of language. In *ACL*.

Anna Schmidt and Michael Wiegand. 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*, pages 1–10, Valencia, Spain. Association for Computational Linguistics.

Lukas Stappen, Fabian Brunn, and Björn W. Schuller. 2020. Cross-lingual zero- and few-shot hate speech detection utilising frozen transformer language models and axel. *ArXiv*, abs/2004.13850.

Emily A Vogels. 2021. The state of online harassment. *Pew Research Center*, 13.

Zeerak Waseem and Dirk Hovy. 2016. Hateful symbols or hateful people? predictive features for hate speech detection on Twitter. In *Proceedings of the NAACL Student Research Workshop*, pages 88–93, San Diego, California. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. Predicting the type and target of offensive posts in social media. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1415–1420, Minneapolis, Minnesota. Association for Computational Linguistics.

A Appendix

A.1 ATOMIC$^{20}_{20}$ Human Readable Templates

The ATOMIC$^{20}_{20}$ knowledge graph consists of tuples in the form of $\langle$ `Head`, `Relation`, `Tail` $\rangle$, where `Head` and `Tail` are provided in textual format. However, we found that training BART on this dataset directly led to sub-optimal results; where it is tasked to predict the `Tail` given the `Head` and `Relation`. Instead we replaced each relation with a human readable template as shown in Table 8, where the `Head` is injected in place of {0} and the model is tasked with predicting the `Tail`.

A.2 StereoSet Training

Nadeem et al. (2021) designed two types of *Context Association Tests* (CATs): intrasentence and intersentence to measure the degree of stereotypical bias in language models at the sentence level and discourse level respectively. In this work, we only consider the latter and only include stereotypical sentences (i.e. omit the anti-stereotype examples). In the pre-finetuning step of StereoSet, the BART model is tasked with predicting the explanation of a given stereotype conditioned on the targeted group, bias type and stereotype. The dataset is linearized according to the following format: **Input**: "Target: `{TARGET_GROUP}` Bias Type: `{BIAS_TYPE}` Sentence: `{STEREOTYPE}` Context?" and the **Output**: is "`{EXPLANATION}`". For example: **Input**: "Target: {Cape Verde} Bias Type: {race} Sentence: {They rely on the ports.} Context?"; **Output**: "{Cape Verde has a lot of water.}".

A.3 Scale Experiments

In this analysis, we investigate the role of scale in TOKEN . Table 9 shows that the results are only better from the 16 to the 64-shot case, suggesting that scale does play a role in the proposed method.

Relation	Human Readable Template
ObjectUse	{0} is used for {1}
AtLocation	You are likely to find {0} in {1}
MadeUpOf	{0} is made up of {1}
HasProperty	{0} is {1}
CapableOf	{0} can {1}
Desires	{0} wants {1}
NotDesires	{0} does not want {1}
isAfter	Something that happens after {0} is {1}
HasSubEvent	Something you might do while {0} is {1}
isBefore	Something that happens before {0} is {1}
HinderedBy	{0} is hindered by {1}
Causes	Sometimes {0} causes {1}
xReason	{0}. The reason for PersonX doing this is {1}
isFilledBy	{0} can be filled by {1}
xNeed	But before {0}, PersonX needed {1}
xAttr	{0} is seen as {1}
xEffect	As a result of {0}, PersonX will {1}
xReact	As a result of {0}, PersonX feels {1}
xWant	After {0}, PersonX would want {1}
xIntent	Because of {0}, PersonX wanted {1}
oEffect	as a result of {0}, others will {1}
oReact	as a result of {0}, others would feel {1}
oWant	as a result of {0}, others would want {1}

Table 8: Human readable templates for each relation used to train the BART model. The `Head` is injected in place of {0} and is tasked with predicting the `Tail` {1} in a conditional generation framework.

Model	16	32	64	128	256	512	1024
Baseline	53.49	51.47	58.28	**66.70**	**68.24**	**70.09**	**71.30**
TOKEN	**55.23**	**60.29**	**63.68**	65.10	67.52	68.67	69.66

Table 9: Results using $\text{BART}_{\text{BASE}}$ as the core model. Similar to the previous experiments the baseline model predicts the HS label directly, while the TOKEN model employs task decomposition and knowledge infusion using the ATOMIC^{20}_{20} and StereoSet datasets. (see §6.3 for more details)