# Learning from Disagreement: Entropy-Guided Few-Shot Selection for Toxic Language Detection

**Tommaso Caselli**
CLCG
University of Groningen, The Netherlands
t.caselli@rug.nl

**Flor Miriam Plaza-del-Arco**
LIACS
Leiden University, The Netherlands
f.m.plaza.del.arco
@liacs.leidenuniv.nl

## Abstract

In-context learning (ICL) has shown significant benefits, particularly in scenarios where large amounts of labeled data are unavailable. However, its effectiveness for highly subjective tasks, such as toxic language detection, remains an open question. A key challenge in this setting is to select shots to maximize performance. Although previous work has focused on enhancing variety and representativeness, the role of annotator disagreement in shot selection has received less attention. In this paper, we conduct an in-depth analysis of ICL using two families of open-source LLMs (`Llama-3*` and `Qwen2.5`) of varying sizes, evaluating their performance in five prominent English datasets covering multiple toxic language phenomena. We use disaggregated annotations and categorize different types of training examples to assess their impact on model predictions. We specifically investigate whether selecting shots based on annotators' entropy – focusing on ambiguous or difficult examples – can improve generalization in LLMs. Additionally, we examine the extent to which the order of examples in prompts influences model behavior. Our results show that selecting shots based on entropy from annotator disagreement can enhance ICL performance. Specifically, ambiguous shots with a median entropy value generally lead to the best results for our selected LLMs in the few-shot setting. However, ICL often underperforms when compared to fine-tuned encoders.

## 1 Introduction

In-context learning (ICL) is becoming a dominant paradigm in NLP, mainly due to its numerous advantages over fine-tuning methods. First, it facilitates task-specific learning in a dynamic way from a limited set of examples directly provided in a prompt, without the need to update the model's weights. This allows rapid adaptation to new tasks while significantly reducing storage and computational costs. Additionally, ICL requires fewer labeled examples compared to full fine-tuning, reducing the need for extensive training sets and making it an efficient and scalable alternative, especially for low-resource languages and cognitively demanding tasks such as hate speech annotation (Plaza-del-Arco et al., 2024a; Poletto et al., 2021; Vidgen and Derczynski, 2020; Davidson et al., 2017). However, as Dong et al. (2024) highlight, multiple factors affect ICL performance, such as the prompt template and wording, the selected examples (henceforth, shots), the order in which the shots are presented to the models, and the model size, among others (Wang et al., 2023; Shi et al., 2024). Finding optimal shots, that is, instances of a linguistic phenomenon that are representative and that could be used to learn to generalize its identification, is a challenging task whose solution could help boost the LLM results in ICL (Zhang et al., 2022; Yang et al., 2023).

This paper presents an in-depth study on ICL functionalities of two families of open-source LLMs (`LlamA-3*` and `Qwen2.5`) with different sizes on five prominent datasets in English covering multiple toxic language phenomena. In particular, we have used the disaggregated annotations of the datasets to identify different types of training examples to test their impact on LLMs. We have also analyzed to what extent the order of presentations of the examples in the prompts impacts the models. Our contributions can be summarized as follows:

1. For the first time, we provide insights into the distribution of annotators' judgments through entropy, by examining how examples are distributed along a complexity axis across five major English-language datasets;
2. We select shots based on annotators' entropy to identify difficult and ambiguous examples to test whether they could represent an advantage in boosting generalization functionalities of LLMs;



*Proceedings of the The 9th Workshop on Online Abuse and Harms (WOAH)*, pages 53–66

3. We benchmark six open-source instruction-tuned LLMs with sizes ranging between 3B up to 72B parameters, showing that a principled selection of shot types (and in some cases of the order of the labels) can help in boosting ICL performance;

4. ICL often underperforms when compared to fine-tuned encoders, suggesting that ICL is suboptimal in scenarios where training data is available.

Our code is publicly available at the following link https://github.com/tommasoc80/woah_2025_shot_selection.

## 2 Datasets

A recent trend in the creation of language resources for highly subjective tasks, such as toxic language, is to collect multiple judgments for every message and subsequently release the annotations both in an aggregated and a disaggregated format. Releasing data in a disaggregated form allows to analyze disagreements, different perspectives, and develop systems that account for different social and cultural viewpoints (Cabitza et al., 2023). For this paper, we select five datasets that either have been conceived to be released in a disaggregated format or that have preserved the disaggregated annotations while offering aggregated labels. The datasets contain English texts, and they are all based on social media messages from different platforms. The spectrum of toxic language phenomena includes sexism, hate speech, and offensive language. Table 1 presents the message distribution across positive and negative classes.

| Dataset | Train | | Dev | | Test | |
|---|---|---|---|---|---|---|
| | POS | NEG | POS | NEG | POS | NEG |
| **EDOS** | 3,398 | 10,602 | 486 | 1,514 | 970 | 3,030 |
| **Brexit** | 72 | 712 | 19 | 149 | 18 | 150 |
| **GAB** | 1,941 | 20,095 | – | – | 509 | 5,001 |
| **MD** | 1,962 | 4,630 | 388 | 716 | 1,018 | 2,039 |
| **SBIC** | 18,726 | 16,698 | 2,612 | 2,054 | 2,710 | 1,981 |

Table 1: Class distribution of the selected datasets. POS refers to the positive class (sexism for EDOS; hate speech for Brexit and GAB; offensive language for MD and SBIC), while NEG represents the negative class.

**EDOS** The Explainable Detection of Online Sexism (EDOS) (Kirk et al., 2023), released in the context of SemEval 2023, consists of 20k messages and it is structured along a three-layered, hierarchical annotations for detecting sexism. The first layer determines whether a message is sexist or not; the second layer identifies four distinct categories of sexism, and, the last layer distinguishes between 11 fine-grained sexism types. We select only the first annotation layer. Messages for EDOS were retrieved from Reddit and GAB. The dataset creators implemented a collection method based on a mix of community-based sampling (for Reddit) and an ensemble of sampling methods, rather than using a set of keywords, to guarantee a better diversity of the data. Sexism is defined as "any implicit or explicit abuse directed at women based on gender or intersecting identities" (Kirk et al., 2023, pg 2194). Nineteen trained women annotators followed strict guidelines, each labeling three messages. Data were manually curated, with expert adjudication resolving disagreements. Table 1 shows a 3:1 skew toward the negative class (non-sexism). The authors do not report an IAA study.

**Brexit** The Hate Speech on Brexit dataset (Akhtar et al., 2021) was also published in the context of SemEval 2023 (Leonardelli et al., 2023). The dataset is composed by 1,120 tweets collected with keywords related to immigration and Brexit. It was annotated with four categories, namely hate speech (in particular xenophobia and islamophobia), aggressiveness, offensiveness, and stereotype, following the annotation scheme and definitions in Sanguinetti et al. (2018). Six annotators, divided into target (Muslim immigrants/students) and control (Western researchers) groups, provided binary annotations, negatively skewed (Table 1). Class ratios vary across splits: train (70:30), dev (80:20), test (90:10). The SemEval task covers only hate speech, with dataset creators reporting Fleiss' kappa of 0.35. Agreement was higher within groups (0.58 target, 0.43 control), with all full disagreements marked hateful by the target group but never by the control. We use the aggregated SemEval labels.

**GAB** The GAB Hate Speech Corpus (Kennedy et al., 2022) is composed by 27,665 posts from the social network platform GAB. The coding typology of GAB is mostly grounded in sociology research. Hate speech is defined as "[l]anguage that intends to [...] attack the dignity of a group of people, either through an incitement to violence, encouragement of the incitement to violence, or

the incitement to hatred"(Kennedy et al., 2022, pg 92). This translates in two different annotation categories: assault on human dignity (HD) and calls for violence (CV). Posts where randomly sampled from GAB considering that users on the platform tend to produce a high amount of hateful and dangerous speech thanks to the platform's relaxed policies on free speech (Matsakis, 2018). The dataset creators recruited and trained with prescriptive annotation guidelines 18 annotators. Only posts with at least three annotations were kept. Annotators labeled between 288 to 13,543 each. Like the other datasets, GAB is also unbalanced with the HD and CV categories corresponding to only 9% of the entire corpus. For our experiments, we merge HD and CV into a single hate speech label (see Table 1).

**MD** The Multi-Domain Agreement dataset (Leonardelli et al., 2021) was originally designed as a "disagreement" dataset, allowing for the creation of different train and test data based on annotators' agreement. The dataset is composed by 10,753 tweets covering three topics (Black Lives Matter, the 2020 US Presidential Elections, and the COVID-19 pandemic). Messages were annotated on Amazon Mechanical Turk (AMT), with annotators judging if the message was offensive based solely on its content. Each message received five judgments, with an unbalanced class distribution similar to the other datasets (31% offensive vs. 69% non-offensive tweets (Table 1). Each tweet was annotated by five AMT workers, totaling 670 unique workers. The dataset creators discarded annotators with less than 70% accuracy on a gold standard. Overall, 43% of messages had full agreement, 29.35% mild agreement (4/5 annotators), and 28.28% weak agreement (3/5 annotators). We used the SemEval 2023 dataset version (Leonardelli et al., 2023).

**SBIC** Social Bias Inference Corpus (Sap et al., 2020) is composed by 44,000 posts from different social media platforms, including Reddit, Twitter, and hate speech forums (e.g., GAB, Stormfront, and other banned subreddits). The annotation framework combines categorical labels (offensiveness, intent, lewdness, group targeting) with free-text explanations of implied stereotypes. Posts were annotated via the AMT platform, with each post receiving three annotations from 264 unique workers. We select only the annotation for the offensive category. Similarly to MD, annotators were asked to judge whether a post can be offensive. The main difference is that the offensive status of the message has to be determined based on the text, and by considering whether it can be perceived as such by someone or everyone, that is eliciting their subjective interpretations rather than having strict annotation guidelines. As reported by the dataset's creators, the IAA for SBIC is of relatively good quality, with an overall pairwise agreement of 84% and an average Krippendorf's $\alpha$, which equals 0.45. Concerning the offensive category, the Krippendorf's $\alpha$ is higher, reaching 0.51, although the pairwise agreement is a bit lower (76%). This is the only dataset with a slightly skewed distribution in favor of the positive class, with 54.7% of the messages marked as offensive (see Table 1).

### 2.1 Entropy as a Measure of Data Complexity

All the datasets we review have implemented various data quality checks, such as directly recruiting and training annotators (Brexit, EDOS, GAB), removing crowd workers who failed to meet minimum annotation accuracy standards (MD), or reporting the IAA scores (SBIC). However, none of these measures provide any insights on two critical issues of dataset creation for NLP models: (i) the representativeness of the target phenomenon and (ii) the complexity of the examples. Although not trivial, a relatively good proxy for the representativeness of target phenomena can be obtained through system portability to out-of-domain distributions (Ettinger et al., 2017). On the other hand, assessing the complexity of the examples is a more difficult task. IAA captures this dimension only partially. Low IAA scores may result from factors such as poor annotation guidelines, task complexity (rather than the difficulty of specific examples), or low annotator quality, nor are IAA scores reliable indicators of a system's future performance (Artstein and Poesio, 2008).

To explore whether selecting shots based on complexity improves models, we use the Multi-Annotator Competence Estimation (MACE) tool (Hovy et al., 2013). MACE is an unsupervised tool based on Variational Bayes inference, designed to identify trustworthy annotators in crowdsourcing tasks. It can also estimate the difficulty of each dataset item and the overall task. In both cases, MACE learns competence estimates of the annotators. Item difficulty is expressed through entropy, with the most challenging instances receiving the highest scores. For each dataset, we identify the unique annotators. Brexit is the only dataset where

the same set of annotators evaluated every data point. In all other datasets, annotators were randomly assigned to different portions of the data. To prevent confusion between annotations from different individuals being grouped under a single annotator label (e.g., A1), we left the values empty for data points that were not annotated by each specific annotator subset. After this fix, we run MACE to compute the entropy scores. To avoid data risk of data leakage or contamination, entropy scores have been run only on the training distributions of each dataset. The results are reported on Table 2.

| Dataset | Mean | Median | Max. | Min. |
|---|---|---|---|---|
| **EDOS** | 4.33e-1 | 6.51e-1 | 6.93e-1 | 4.84e-5 |
| **Brexit** | 7.42e-2 | 7.94e-4 | 6.92e-1 | 5.80e-5 |
| **GAB** | 9.57e-2 | 2.56e-2 | 6.93e-1 | 1.90e-5 |
| **MD** | 1.24e-1 | 1.60e-2 | 6.93e-1 | 9.02e-8 |
| **SBIC** | 1.78e-1 | 7.38e-2 | 6.93e-1 | 5.11e-6 |

Table 2: Summary of average, median, max. and min. values of the entropy scores of each dataset obtained with MACE.

If we focus on the average entropy score, we observe that the majority of these datasets (4 out 5) present low values. This suggests that the data are likely to contain easy instances. In our case, an "easy instance" refers to a data item that clearly expresses the target phenomenon of the dataset. EDOS is the only dataset with relatively high entropy scores, indicating that its data items are more challenging. This is further supported by the median value, which is higher than the average entropy and closer to the maximum entropy score. Notably, all datasets show similar maximum entropy scores, but differ in their minimum values. The distribution of entropy scores is usually negatively skewed and is comparable between positive and negative classes. There is a general tendency of the positive class instances to have higher values than their negative counterparts. The only dataset that presents a homogeneous behavior between the two classes is SBIC. EDOS, on the other hand, is the only dataset that shows a bimodal distribution. See distribution plots in Appendix A.

## 3 Experiments

For each dataset, we design seven experiment settings. First, models are tested in zero-shot settings as a baseline. The other six settings are based on few-shot ICL and can be grouped into three blocks according to the types of shots. The first block uses examples deemed as "difficult" (`diff`), the second block employs shots considered as "ambiguous" (`amg`), and the third randomly selects the shots (`random`). In each block, the shots are presented either in a fixed order per class label or in a randomized one. In particular, for the fixed order (`ordered`), all shots belonging to the negative class are presented first, followed by those for the positive class. For the randomized version (`shuffled`), we shuffle the order of the examples per dataset. This means that there are no two datasets whose order of the shot labels is the same. For all ICL settings, we select 20 shots, 10 for the positive class and 10 for the negative class. Table 3 summarizes the combinations of the data points per entropy bin and presentation of the shots.

| Shot Order | Entropy Bin: Ambiguous | Entropy Bin: Difficult | Entropy Bin: Random |
|---|---|---|---|
| By label | `amb-ordered` | `diff-ordered` | `random-ordered` |
| Randomized | `amb-shuffled` | `diff-shuffled` | `random-shuffled` |

Table 3: Combinations of shot order and entropy bin used in the ICL experiments.

The idea of selecting different types of shots to identify representative samples of a target phenomenon is inspired by the Dataset Cartography (DC) method (Swayamdipta et al., 2020). The method relies on a model's confidence in the true class and its fluctuations across training epochs to pinpoint reliable and informative data points. This makes it possible to train with less data while maintaining or even improving performance. Mapping these training dynamics reveals a spectrum of data points: *easy* (high confidence, low variability), *hard* (low confidence, low variability), and *ambiguous* (mid-range confidence, high variability). Since we have access to disaggregated annotations for all our datasets, we opt to use the annotators' entropy scores to select the shots. The "difficult" instances are selected by ranking the data points of each dataset and each class in decreasing order of entropy score. The top ten data points per class correspond to our "difficult" shots. For the "ambiguous" cases we use the median score and the median absolute deviation. Ambiguity has been set as the range between the median plus the median absolute deviation and the median minus the median absolute deviation. The median and the median absolute deviation have been determined per class in each dataset.

### 3.1 LLMs and Prompt Settings

We identify three groups of models belonging to two different collections, `LlaMa-3*` and `Qwen-2.5`, comparable for their sizes. This allows for a fair comparison between models that have been obtained using different pre-training data and optimization techniques. While both collections rely on the basic decoder-only Transformer architecture and support a context window of up to 128K tokens, they present some differences. `LlaMa-3*` has been trained on a 15T token corpus collected from publicly available sources. The models use grouped query attention (GQA) to improve inference efficiency and, at post-training, they combine supervised fine-tuning (SFT), rejection sampling, proximal policy optimization (PPO), and direct preference optimization (DPO).[1] As for `Qwen-2.5`, the models have been trained on a collection of 18T tokens, including synthetic data, and support 29 languages (Yang et al., 2024). Standard feed-forward network (FFN) layers have been replaced with Mixture of Experts (MoE) layers. For the post-training steps, `Qwen-2.5` uses 1 million examples across SFT, DPO and group relative policy optimization (GRPO). All selected models for both collections are text-only and instruction-tuned. The selected models are the following: `LlaMa-3.2-3B`, `LlaMa-3-8B`; `LlaMa-3-70B`; `Qwen-2.5-3B`, `Qwen-2.5-7B`; and `Qwen-2.5-72B`.

To minimize safe guard mechanisms which may results in the models refusing to answer the prompt, we have specified in the prompt preamble that the system's role is that of AI expert in text classification and content moderation. The same prompt has been used across all datasets. The only variation concerned the task at hand by making explicit which toxic language phenomenon is targeted (i.e., sexism, hate speech, or offensive language). Instances of the prompts for each experiment settings are reported in Figures F, G and H in Appendix B. Actual prompts and shots are publicly available.[2]

## 4 Results and Discussion

We present out findings in three blocks: first, we discuss the results *per dataset* (§ 4.1); then we discuss the results *per model* (§ 4.2), and finally we compare the results of the LLMs against fine-tuned versions of an encoder-based model, HateBERT (Caselli et al., 2021) (§ 4.3).

[1] `https://ai.meta.com/blog/meta-llama-3/`

[2] Link to repository publicly available upon acceptance

### 4.1 Results by Dataset

Table 4 presents the results of our experiments by ICL setting and datasets across the different models. In the following, we discuss the results by dataset.

**EDOS** EDOS focuses on sexism detection. We observe that zero-shot achieves a low performance across models with an average of 0.505 as macro-F1. However, few-shot settings improve over zero-shot across models. The best result (0.752) is achieved by `Llama3-70B` in the `amb-shuffled` setting – with a $\Delta$ = -0.122 against the best model in the SemEval evaluation (Kirk et al., 2023), which employs an ensemble of encoders.. Shuffling the shots yields better performance than presenting them in fixed order, i.e., `amb-ordered` setting (0.719). With the exception of two models, `Llama3.2-3B` and `Qwen2.5-72B` where choosing shots randomly improves the results over zero-shot. we find across the several models we test that whether ordered or shuffled, instances reflecting a wider spectrum of complexity (`amb` setting) improve sexism detection.

**Brexit** This dataset, which involves political discourse, also benefits from the few-shot setup, especially when samples are chosen based on the highest entropy, i.e., those that are more challenging (`diff_setting`). The best result is achieved by `Qwen2.5-7B` in the `diff-ordered` setting, with an F1 score of 0.812, a notable improvement over the zero-shot baseline score of 0.596 for this model. The corresponding SemEval task (Leonardelli et al., 2023) adopts micro-F1 scores for evaluating the aggregated labels. Our best ICL model achieves competitive results against the best model based on a multi-task learning and encoder fine-tuning (0.915 for`Qwen2.5-7B` *vs*). 0.932).

**GAB** This dataset follows a similar pattern to EDOS, with zero-shot showing relatively low performance across models with `Llama3-70B` achieving the best result, a macro-F1 score of 0.666. However, in contrast to previous datasets, the few-shot setting that works the best is based on random selection with mixed results across whether they are ordered (`random-ordered`, 0.713) or shuffled (`random-shuffled`, 0.711). The best result is achieved by `Qwen2.5-7B` and the `random-ordered` setting with a macro-F1 score of 0.713, a 4.7% of improvement over the best zero-shot result, and a positive $\Delta$ = 0.08 for the F1-score on the positive class against Kennedy et al. (2022).

| Dataset | ICL Setting | Model | | | | | |
|---|---|---|---|---|---|---|---|
| | | Llama3.2-3B | Qwen2.5-3B | Llama3-8B | Qwen2.5-7B | Llama3-70B | Qwen2.5-72B |
| **EDOS** | 0-shot | 0.457 (0.461) | 0.423 (0.450) | 0.477 (0.460) | 0.503 (0.484) | 0.672 (0.596) | 0.495 (0.136) |
| | amb-ordered | 0.591 (0.504) | 0.619 (0.518) | **0.675** (0.572) | 0.716 (0.589) | 0.719 (0.636) | 0.666 (0.590) |
| | amb-shuffled | 0.606 (0.504) | **0.654** (0.539) | 0.641 (0.561) | **0.719** (0.596) | **0.752** (0.664) | 0.652 (0.581) |
| | diff-ordered | 0.596 (0.501) | 0.566 (0.498) | 0.671 (0.553) | 0.702 (0.589) | 0.714 (0.632) | 0.590 (0.531) |
| | diff-shuffled | 0.615 (0.494) | 0.578 (0.498) | 0.649 (0.544) | 0.688 (0.593) | 0.708 (0.626) | 0.610 (0.552) |
| | random-ordered | 0.572 (0.493) | 0.563 (0.495) | 0.674 (0.574) | 0.563 (0.495) | 0.703 (0.627) | **0.673** (0.594) |
| | random-shuffled | **0.628** (0.496) | 0.575 (0.502) | 0.598 (0.534) | 0.694 (0.598) | 0.729 (0.646) | 0.650 (0.578) |
| **Brexit** | 0-shot | 0.486 (0.318) | 0.433 (0.278) | 0.508 (0.323) | 0.596 (0.390) | 0.599 (0.409) | 0.512 (0.121) |
| | amb-ordered | 0.590 (0.400) | 0.675 (0.466) | 0.722 (0.542) | 0.736 (0.566) | 0.653 (0.459) | 0.664 (0.472) |
| | amb-shuffled | 0.595 (0.404) | 0.662 (0.463) | 0.716 (0.539) | 0.723 (0.548) | **0.698** (0.515) | 0.648 (0.453) |
| | diff-ordered | 0.639 (0.450) | 0.613 (0.400) | **0.780** (0.627) | **0.812** (0.680) | 0.673 (0.477) | 0.488 (0.453) |
| | diff-shuffled | **0.699** (0.521) | **0.757** (0.596) | 0.749 (0.586) | 0.757 (0.596) | 0.691 (0.500) | **0.692** (0.507) |
| | random-ordered | 0.590 (0.400) | 0.592 (0.385) | 0.722 (0.542) | 0.771 (0.618) | 0.642 (0.433) | 0.669 (0.478) |
| | random-shuffled | 0.595 (0.404) | 0.596 (0.373) | 0.705 (0.529) | 0.675 (0.485) | 0.659 (0.473) | 0.649 (0.461) |
| **GAB** | 0-shot | 0.538 (0.322) | 0.558 (0.332) | 0.612 (0.380) | 0.646 (0.416) | 0.666 (0.444) | 0.570 (0.201) |
| | amb-ordered | 0.582 (0.354) | 0.618 (0.381) | 0.665 (0.439) | 0.686 (0.457) | 0.648 (0.420) | 0.640 (0.411) |
| | amb-shuffled | 0.587 (0.357) | 0.618 (0.383) | 0.646 (0.419) | 0.668 (0.435) | 0.647 (0.420) | 0.634 (0.407) |
| | diff-ordered | 0.514 (0.297) | 0.601 (0.366) | 0.681 (0.458) | 0.681 (0.450) | 0.637 (0.405) | 0.620 (0.357) |
| | diff-shuffled | 0.592 (0.358) | 0.635 (0.403) | 0.622 (0.394) | 0.679 (0.449) | 0.607 (0.380) | 0.659 (0.432) |
| | random-ordered | 0.575 (0.338) | 0.662 (0.431) | **0.697** (0.477) | **0.713** (0.486) | 0.670 (0.440) | **0.680** (0.455) |
| | random-shuffled | **0.670** (0.430) | **0.680** (0.451) | 0.670 (0.445) | 0.711 (0.486) | **0.670** (0.441) | 0.674 (0.448) |
| **MD** | 0-shot | 0.587 (0.602) | 0.410 (0.539) | 0.552 (0.586) | 0.635 (0.620) | 0.747 (0.697) | 0.478 (0.170) |
| | amb-ordered | 0.650 (0.616) | 0.555 (0.475) | 0.690 (0.647) | 0.678 (0.576) | **0.761** (0.718) | **0.732** (0.700) |
| | amb-shuffled | 0.666 (0.594) | 0.562 (0.415) | 0.546 (0.589) | 0.671 (0.595) | 0.717 (0.683) | 0.664 (0.658) |
| | diff-ordered | 0.645 (0.571) | 0.569 (0.417) | 0.669 (0.567) | 0.657 (0.536) | 0.716 (0.646) | 0.699 (0.658) |
| | diff-shuffled | 0.663 (0.550) | **0.595** (0.477) | 0.676 (0.586) | 0.661 (0.558) | 0.719 (0.645) | 0.709 (0.665) |
| | random-ordered | 0.673 (0.616) | 0.554 (0.397) | **0.696** (0.656) | **0.687** (0.622) | 0.700 (0.675) | 0.674 (0.663) |
| | random-shuffled | **0.689** (0.599) | 0.555 (0.429) | 0.587 (0.607) | 0.670 (0.620) | 0.692 (0.666) | 0.691 (0.672) |
| **SBIC** | 0-shot | 0.734 (0.793) | 0.696 (0.795) | 0.696 (0.746) | 0.757 (0.814) | 0.781 (0.814) | 0.350 (0.108) |
| | amb-ordered | 0.733 (0.784) | 0.718 (0.765) | **0.762** (0.792) | 0.758 (0.775) | **0.803** (0.837) | 0.794 (0.836) |
| | amb-shuffled | 0.724 (0.762) | **0.739** (0.792) | 0.756 (0.792) | **0.763** (0.783) | 0.791 (0.827) | **0.801** (0.843) |
| | diff-ordered | 0.735 (0.766) | 0.720 (0.770) | 0.744 (0.762) | 0.726 (0.727) | 0.771 (0.809) | 0.771 (0.813) |
| | diff-shuffled | 0.718 (0.741) | 0.708 (0.740) | 0.734 (0.790) | 0.699 (0.700) | 0.779 (0.813) | 0.779 (0.824) |
| | random-ordered | 0.702 (0.794) | 0.704 (0.796) | 0.754 (0.805) | 0.758 (0.794) | 0.787 (0.831) | 0.778 (0.837) |
| | random-shuffled | **0.751** (0.781) | 0.725 (0.782) | 0.740 (0.805) | 0.752 (0.778) | 0.785 (0.825) | 0.774 (0.835) |
| *Mean* | | 0.628 (0.525) | 0.614 (0.508) | 0.671 (0.578) | 0.695 (0.586) | **0.706 (0.604)** | 0.652 (0.529) |

Table 4: Experiments results: we report macro-F1 and, in brackets, the F1 score for the positive class. The best ICL setting per model and dataset is highlighted in **bold**.

**MD** MD, which contains instances annotated with offensive language shows more complex patterns. The best zero-shot performance is achieved by Llama3-70B with a macro-F1 score of 0.747, indicating that the model already performs fairly well without additional examples. Performance improves across the ICL settings, particularly in the amb-ordered and diff-shuffled settings. For example, in the amb-ordered setting, Llama3-70B achieves an F1 score of 0.761. Like for EDOS and Brexit, access to shots with varied levels of complexity has a positive effect. When compared to the best model in the SemEval task (Leonardelli et al., 2023), performance remains lower, with a micro-F1 score of 0.775 versus 0.846.

**SBIC** This dataset, which also targets offensive language, shows an interesting trend where the zero-shot performance is relatively high, especially for models like Llama3-70B (0.781). However, in this setup, other models like Qwen2.5-72 show very low performance with an F1-score of 0.350. For the few-shot settings, the amb-ordered, which includes examples of mixed complexity, outperforms other configurations, further supporting a principled selection of the shots to boost models' performance. Llama3-70B achieves the highest macro-F1 score of 0.803, representing the best few-shot result. SBIC is the dataset that obtains the best results across models and settings. In this case, ICL achieves better performance compared to the original paper (Sap et al., 2020), with a positive $\Delta$ F1-score of 0.049 on the offensive class.

### 4.2 Results by LLM

Building on the results presented in Table 4, we now discuss the key findings for the LLMs.

**Larger models perform better** Generally, `70/72B` models are those that almost always achieve the best results, with the exception for Brexit and GAB. For GAB, the scores are all in the same range across all models while for Brexit we observe that `3B` models have comparable results to the `70/72B` ones, with `Qwen2.5-3B` achieving better results.

**Few-shot is better than zero-shot** In every dataset and regardless of the shot type and their order of presentation, few-shot consistently obtains better results than the zero-shot, reinforcing findings from previous work (Shi et al., 2024; Dong et al., 2024). The additional insight we offer is that all models struggle to detect the positive class in zero-shot. `Qwen2.5-72B` is the most underperforming LLM - especially against its smaller versions.

**Performant shot ordering depends on model's size** An emerging pattern suggests a relationship between model size and shot order. In particular, it appears that `3B` models prefer labels in random order while `7/8B` ones perform best with the ordered format. The `70/72B` variants are less consistent, with the order of labels dependent on the specific model. For instance, `Llama3-70B` tends to perform better when labels are presented in a randomized order, while the opposite holds for `Qwen2.5-72B`. These findings go in a different direction when compared to Lu et al. (2022) where the authors claim that performant label ordering is not consistent across models.

**Varying the complexity of the shots helps** We have already seen that, with the exception of Brexit, using shots with varied complexities (i.e., ambiguous) improves models' performance. When looking at the average macro-F1 across all models by ICL setting, we observe that using ambiguous shots in an ordered format (`amb-ordered`) achieves the best score (0.683), immediately followed by difficult shots in a randomized order (0.681). Although the results show some variations across datasets and models, they also indicate that entropy can serve as a good proxy to identify shots that are representative of a targeted phenomenon. Table A in Appendix C presents a summary.

**Dataset's entropy can help to select LLM size** Entropy can also be used as a proxy to select the model's size. We ran a correlation analysis using Spearman correlation between the best models (per size) and the entropy scores of each dataset. Although all correlations are not statistically significant, we observe different behaviors according to the models' size. Similarly to the label order, smaller models (`3B` and `7/8B` models) obtain better scores on datasets with lower entropy scores ($\rho$ = -0.372 for 3B, and $\rho$ = -0.421 for `7/8B`), while the opposite holds for larger models ($\rho$ = 0.378 for `70/72B`). Although these findings have limitations in generalizability, the trend indicates that larger models should be used with challenging datasets (according to the annotators' entropy), whereas smaller models can achieve strong, if not optimal, results on simpler datasets.

**Models fail to follow instructions rather than refusing to answer** Following Wang et al. (2024), acknowledging the refusal rate of LLMs is an integral part of the evaluation of these technologies. In our evaluations, we took into account both the refusal rate (i.e., a model refusing to complete the task because of safeguard railways) and their failure to adhere to the answer format. The overall picture that emerges is that the refusal rate is almost zero in the large majority of cases.[3] However, failure to follow the instructions is much higher, with peaks of 84% for `Qwen2.5-72B` in zero-shot. Small models, i.e., `3Bs`, are more likely to fail to follow the instructions. We also observe that some datasets (namely MD and SBIC) trigger more failures than others. The full overview is available in Table B, Appendix D.

### 4.3 ICL vs. Fine-tuning

Table 5 summarizes the final set of experiments, comparing LLM performance against HateBERT with both frozen layers and full fine-tuning on each dataset.

The advantage of using LLMs is clear in zero-shot when compared to HateBERT with frozen layers. Although HateBERT has been further-pretrained with data where toxic language is highly present and covers different language phenomena, it consistently underperforms. On the other hand, fine-tuned versions of HateBERT proves to be highly

[3] Refusals have been identified using regular expressions. For metrics calculations, for all cases of failure we have always assigned the negative class.

| Dataset | Setting | Macro-F1 | Δ LLM |
|---|---|---|---|
| **EDOS** | frozen | 0.431 (0.000) | [-0.241;-0.064] |
| | fine-tuned | 0.831 (0.744) | [+0.079;+0.158] |
| **Brexit** | frozen | 0.417 (0.000) | [-0.182;-0.179] |
| | fine-tuned | 0.659 (0.363) | [-0.121;-0.153] |
| **GAB** | frozen | 0.475 (0.000) | [-0.191;-0.171] |
| | fine-tuned | 0.631 (0.305) | [-0.050;-0.050] |
| **MD** | frozen | 0.400 (0.000) | [-0.347;-0.235] |
| | fine-tuned | 0.766 (0.678) | [+0.005;+0.034] |
| **SBIC** | frozen | 0.564 (0.704) | [-0.217;-0.193] |
| | fine-tuned | 0.846 (0.866) | [+0.043;+0.045] |

Table 5: Results for HateBERT (frozen layers and fine-tuned). Deltas with LLMs are reported as intervals, with the first score referring to the best `Llama3*` model and the second to the best `Qwen2.5`. HabetBERT with frozen layer is compared to zero-shot LLMs.

competitive against LLMs, if not better as shown by the positive deltas (in favor of HateBERT) for EDOS, MD, and SBIC. Notably, even the results for Brexit and GAB, although lower, are very close.

These findings challenge the prevailing trend in NLP to use generative models for all tasks, regardless of the available data. It seems quite clear that LLMs have a distinct advantage over encoder models in zero-shot scenarios (where no training data is available). Conversely, if training data is accessible, fine-tuning encoder models offer a more affordable, faster, and environmentally friendly choice relative to ICL with LLMs. Rather than fully rejecting LLMs,, these findings point to possible ways to investigate techniques using the best features of both encoder models and LLMs to maximize performance.

## 5 Related Work

Since the advent of LLMs, two main paradigms of ICL (zero-shot and few-shot) have enabled model prompting without the need for large amounts of labeled data (Liu et al., 2023). These methods use less training data, making LLMs efficient and scalable, especially in subjective tasks where the presence of labeled data can be very limited, such as for hate speech detection. Several papers have mainly explored zero-shot ICL for this task (Chiu et al., 2021; Liu et al., 2023; Goldzycher and Schneider, 2022). However, few studies have focused on measuring the impact of evaluation choices, like the prompt phrasing or the impact of the selection of the shots in the few-shot setup. For instance, Plaza-del-Arco et al. (2023) provides a benchmark for zero-shot hate speech detection and show that both the prompt and the model have a significant impact on achieving more accurate predictions in this task. García-Díaz et al. (2023) evaluate zero-shot and few-shot approaches on English and Spanish datasets. For the few-shot approach, they randomly selected five shots of each label. They find that few-shot does not outperform zero-shot in most cases, but they do not provide an exploration of the selection of the shots. Hee et al. (2024) explore the transferability of hate speech detection between modalities (language and vision) using few-shot. They show that vision-language hate speech detection benefits from few-shot learning with text-based hate speech examples. (Maronikolakis et al., 2024) introduce HATELEXICON, a lexicon of slurs and targets of hate speech for Brazil, Germany, India and Kenya. They show that selecting shots based on their lexicon leads to models performing better than models trained on shots sampled randomly.

Other works have investigated the impact of the order of shot selection (Lu et al., 2022) and the quality of the shots devising different solutions to identify the optimal shots and mitigate the sensitivity of models to prompts (Zhang et al., 2022; Gonen et al., 2023; Yang et al., 2023). For few-shot ICL, the impact of shot selection across various toxic language phenomena – such as hate speech, sexism, and offensive language – remains an open question, which we have addressed by building on the entropy-based sampling approach proposed by Plaza-del-Arco et al. (2024b).

## 6 Conclusions

This paper presents a comprehensive investigation of the functionalities of two collections of LLMs, `Llama-3*` and `Qwen2.5`, with different ICL settings. In particular, we have tested on six LLMs - ranging from `3B` up to `70/72B` - the impact of shot selection and label ordering by benchmarking them on five English datasets targeting different toxic language phenomena such as sexism (EDOS), hate speech (Brexit and GAB), and offensive language (MD and SBIC). Unlike previous studies investigating the solutions for identifying optimal shots, we have adopted simple solutions leveraging annotators' disagreement. We have used MACE to calculate the entropy scores of each data item and then use it to identify shots with varying levels of complexity, inspired by the Data Cartography method (Swayamdipta et al., 2020). Although entropy is not a good proxy to predict models' performance, it could offer strategic insights into model

selection based on dataset complexity.

Our results indicate that shot selection plays a prominent role in boosting LLM performance - as indicated by previous work - also for toxic language phenomena. We have identified that ambiguous shots, i.e., those with a median value of entropy, are those that, in general, allows our selected LLMs to obtain the best results in the few-shot setting. Contrary to previous findings (Lu et al., 2022), we have identified that performant shot ordering seems dependent on the LLM's size. Furthermore, the results highlight that when ample training data is available, fine-tuned models offer a more efficient resource-effective alternative to ICL.

Future work could explore the portability of few-shot ICL to out-of-domain distributions, both for the same toxic language phenomenon and for different types of toxicity. This would help assess the reliability of few-shot ICL in scenarios where labeled data is scarce or unavailable.

## Limitations

Our work presents some limitations suggesting directions for future work. One key limitation is that entropy scores are computed using disaggregated annotations, which are not always available across NLP tasks.

We have used two collections of models. Although we vary models' sizes and the models are representative of current trends, the absence of evaluations on other architectures (e.g., encoder-decoder models) restricts the applicability of the conclusions regarding ICL and shot selection.

Finally, the sensitivity of ICL performance to prompt formulation is an aspect that must be taken into account, as it could limit the complete reproducibility or the application of our findings to other settings.

## Acknowledgments

During part of this study, Flor Miriam Plaza-del-Arco was supported by the European Research Council (ERC) through the European Union's Horizon 2020 research and innovation program (grant agreement No. 949944, INTEGRATOR), as part of the MilaNLP group and the Data and Marketing Insights Unit of the Bocconi Institute for Data Science and Analysis (BIDSA).

We thank the Center for Information Technology of the University of Groningen for their support and for providing access to the Hábrók high performance computing cluster.

This work used the Dutch national e-infrastructure with the support of NWO Small Compute applications grant no. EINF-12946.

## References

Sohail Akhtar, Valerio Basile, and Viviana Patti. 2021. Whose opinions matter? perspective-aware models to identify opinions of hate speech victims in abusive language detection. *arXiv preprint arXiv:2106.15896*.

Ron Artstein and Massimo Poesio. 2008. Survey article: Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596.

Federico Cabitza, Andrea Campagner, and Valerio Basile. 2023. Toward a perspectivist turn in ground truthing for predictive computing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 6860–6868.

Tommaso Caselli, Valerio Basile, Jelena Mitrović, and Michael Granitzer. 2021. HateBERT: Retraining BERT for abusive language detection in English. In *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, pages 17–25, Online. Association for Computational Linguistics.

Ke-Li Chiu, Annie Collins, and Rohan Alexander. 2021. Detecting hate speech with gpt-3. *arXiv preprint arXiv:2103.12407*.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *Proceedings of the international AAAI conference on web and social media*, volume 11, pages 512–515.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. 2024. A survey on in-context learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1107–1128, Miami, Florida, USA. Association for Computational Linguistics.

Allyson Ettinger, Sudha Rao, Hal Daumé III, and Emily M. Bender. 2017. Towards linguistically generalizable NLP systems: A workshop and shared task. In *Proceedings of the First Workshop on Building Linguistically Generalizable NLP Systems*, pages 1–10, Copenhagen, Denmark. Association for Computational Linguistics.

José Antonio García-Díaz, Ronghao Pan, and Rafael Valencia-García. 2023. Leveraging zero and few-shot learning for enhanced model generality in hate speech detection in spanish and english. *Mathematics*, 11(24):5004.

Janis Goldzycher and Gerold Schneider. 2022. Hypothesis engineering for zero-shot hate speech detection. *arXiv preprint arXiv:2210.00910*.

Hila Gonen, Srini Iyer, Terra Blevins, Noah Smith, and Luke Zettlemoyer. 2023. Demystifying prompts in language models via perplexity estimation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10136–10148, Singapore. Association for Computational Linguistics.

Ming Shan Hee, Aditi Kumaresan, and Roy Ka-Wei Lee. 2024. Bridging modalities: Enhancing cross-modality hate speech detection with few-shot in-context learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7785–7799, Miami, Florida, USA. Association for Computational Linguistics.

Dirk Hovy, Taylor Berg-Kirkpatrick, Ashish Vaswani, and Eduard Hovy. 2013. Learning whom to trust with MACE. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1120–1130, Atlanta, Georgia. Association for Computational Linguistics.

Brendan Kennedy, Mohammad Atari, Aida Mostafazadeh Davani, Leigh Yeh, Ali Omrani, Yehsong Kim, Kris Coombs, Shreya Havaldar, Gwenyth Portillo-Wightman, Elaine Gonzalez, Joe Hoover, Aida Azatian, Alyzeh Hussain, Austin Lara, Gabriel Cardenas, Adam Omary, Christina Park, Xin Wang, Clarisa Wijaya, Yong Zhang, Beth Meyerowitz, and Morteza Dehghani. 2022. Introducing the gab hate corpus: defining and applying hate-based rhetoric to social media posts at scale. *Lang. Resour. Eval.*, 56(1):79–108.

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

Elisa Leonardelli, Gavin Abercrombie, Dina Almanea, Valerio Basile, Tommaso Fornaciari, Barbara Plank, Verena Rieser, Alexandra Uma, and Massimo Poesio. 2023. SemEval-2023 task 11: Learning with disagreements (LeWiDi). In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 2304–2318, Toronto, Canada. Association for Computational Linguistics.

Elisa Leonardelli, Stefano Menini, Alessio Palmero Aprosio, Marco Guerini, and Sara Tonelli. 2021. Agreeing to disagree: Annotating offensive language datasets with annotators' disagreement. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10528–10539, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM computing surveys*, 55(9):1–35.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2022. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8086–8098, Dublin, Ireland. Association for Computational Linguistics.

Antonis Maronikolakis, Abdullatif Köksal, and Hinrich Schuetze. 2024. Sociocultural knowledge is needed for selection of shots in hate speech detection tasks. In *Proceedings of the Fourth Workshop on Language Technology for Equality, Diversity, Inclusion*, pages 1–13, St. Julian's, Malta. Association for Computational Linguistics.

Louise Matsakis. 2018. Pittsburgh synagogue shooting suspect's gab posts are part of a pattern. WIRED.

Flor Miriam Plaza-del-Arco, Debora Nozza, Marco Guerini, Jeffrey Sorensen, and Marcos Zampieri. 2024a. Countering Hateful and Offensive Speech Online - Open Challenges. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Tutorial Abstracts*, pages 11–16, Miami, Florida, USA. Association for Computational Linguistics.

Flor Miriam Plaza-del-Arco, Debora Nozza, and Dirk Hovy. 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)*, pages 60–68, Toronto, Canada. Association for Computational Linguistics.

Flor Miriam Plaza-del-Arco, Debora Nozza, and Dirk Hovy. 2024b. Wisdom of instruction-tuned language model crowds. exploring model label variation. In *Proceedings of the 3rd Workshop on Perspectivist Approaches to NLP (NLPerspectives) @ LREC-COLING 2024*, pages 19–30, Torino, Italia. ELRA and ICCL.

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*, 55:477–523.

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, 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 *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5477–5490, Online. Association for Computational Linguistics.

Zhenmei Shi, Junyi Wei, Zhuoyan Xu, and Yingyu Liang. 2024. Why larger language models do in-context learning differently? In *International Conference on Machine Learning*, pages 44991–45013. PMLR.

Swabha Swayamdipta, Roy Schwartz, Nicholas Lourie, Yizhong Wang, Hannaneh Hajishirzi, Noah A. Smith, and Yejin Choi. 2020. Dataset cartography: Mapping and diagnosing datasets with training dynamics. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9275–9293, Online. Association for Computational Linguistics.

Bertie Vidgen and Leon Derczynski. 2020. Directions in abusive language training data, a systematic review: Garbage in, garbage out. *Plos one*, 15(12):e0243300.

Xinpeng Wang, Bolei Ma, Chengzhi Hu, Leon Weber-Genzel, Paul Röttger, Frauke Kreuter, Dirk Hovy, and Barbara Plank. 2024. "my answer is C": First-token probabilities do not match text answers in instruction-tuned language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 7407–7416, Bangkok, Thailand. Association for Computational Linguistics.

Xinyi Wang, Wanrong Zhu, Michael Saxon, Mark Steyvers, and William Yang Wang. 2023. Large language models are latent variable models: Explaining and finding good demonstrations for in-context learning. *Advances in Neural Information Processing Systems*, 36:15614–15638.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Zhe Yang, Damai Dai, Peiyi Wang, and Zhifang Sui. 2023. Not all demonstration examples are equally beneficial: Reweighting demonstration examples for in-context learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13209–13221, Singapore. Association for Computational Linguistics.

Yiming Zhang, Shi Feng, and Chenhao Tan. 2022. Active example selection for in-context learning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9134–9148, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

# A Entropy Distribution Datasets

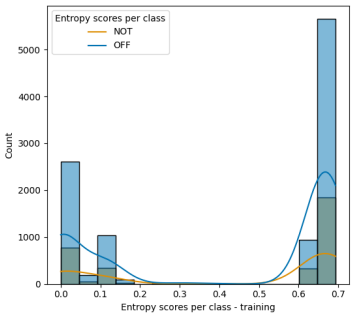


Figure A: EDOS (Kirk et al., 2023).

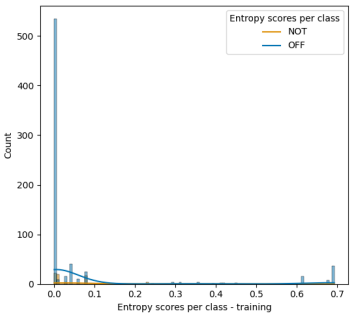


Figure B: Brexit (Akhtar et al., 2021).

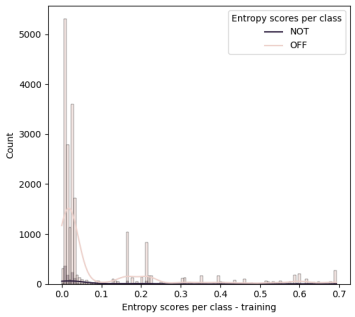


Figure C: GAB (Kennedy et al., 2022).

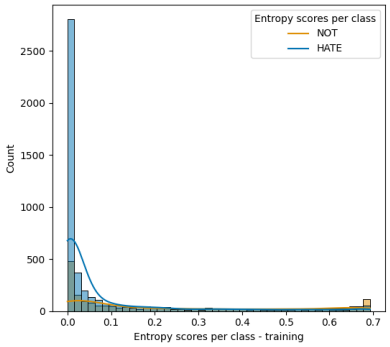


Figure D: MD (Leonardelli et al., 2021).

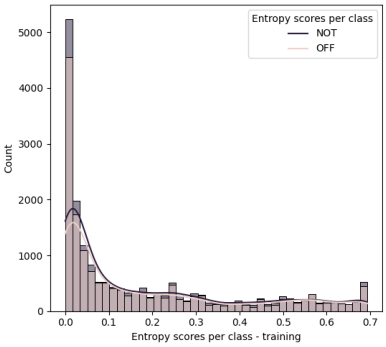


Figure E: SBIC (Sap et al., 2020).

## B Prompt Templates

**SYSTEM:** You are an AI expert in text classification and content moderation.

You will read a text related to [DATASET]. Does the text contain [TOXIC PHENOMENON]?

Answer only with 0 for NOT and 1 for YES.

Do not write an introduction or summary. Classify always the text.

Now classify the following text: `{text}`

Respond only with the valid JSON format below:
`{'text': '{text}', 'label': }`

Figure F: Zero-shot prompt.

**SYSTEM:** You are an AI expert in text classification and content moderation.

You will read a text related to [DATASET]. Does the text contain [TOXIC PHENOMENON]?

Answer only with 0 for NOT and 1 for YES.

Do not write an introduction or summary. Classify always the text.
Here are twenty examples:
`{'text': text_example_1, 'label': 0},`
...
`{'text': text_example_10, 'label': 0},`
...
`{'text': text_example_11, 'label': 1},`
...
`{'text': text_example_20, 'label': 1},`

Now classify the following text: `{text}`

Respond only with the valid JSON format below:
`{'text': '{text}', 'label': }`

Figure G: Few-shot prompt ordered.

**SYSTEM:** You are an AI expert in text classification and content moderation.

You will read a text related to [DATASET]. Does the text contain [TOXIC PHENOMENON]?

Answer only with 0 for NOT and 1 for YES.

Do not write an introduction or summary. Classify always the text.
Here are twenty examples:
`{'text': text_example_1, 'label': 1},`
...
`{'text': text_example_10, 'label': 0},`
...
`{'text': text_example_11, 'label': 1},`
...
`{'text': text_example_20, 'label': 1},`

Now classify the following text: `{text}`

Respond only with the valid JSON format below:
`{'text': '{text}', 'label': }`

Figure H: Few-shot prompt shuffled.

## C Average Results Across Models and ICL Settings

| Model | 0-shot | amb-ord. | amb-shuff. | diff-ord. | diff-shuff. | random-ord. | random-shuff. |
|---|---|---|---|---|---|---|---|
| Llama3.2-3B | 0.560 | 0.629 | 0.636 | 0.626 | 0.657 | 0.622 | **0.667** |
| Qwen2.5-3B | 0.504 | 0.637 | 0.647 | 0.614 | **0.655** | 0.615 | 0.626 |
| Llama3-8B | 0.569 | 0.703 | 0.661 | **0.709** | 0.686 | **0.709** | 0.660 |
| Qwen2.5-7B | 0.627 | 0.715 | 0.709 | **0.716** | 0.697 | 0.698 | 0.700 |
| Llama3-70B | 0.693 | 0.717 | **0.721** | 0.702 | 0.701 | 0.700 | 0.707 |
| Qwen2.5-72B | 0.481 | **0.699** | 0.680 | 0.634 | 0.690 | 0.695 | 0.688 |
| *Mean* | 0.572 | **0.683** | 0.676 | 0.667 | 0.681 | 0.673 | 0.675 |

Table A: Average of macro-F1 scores across all ICL settings and datasets. Best ICL setting per model is highlighted in **bold**.

## D Missing Answers and Refusal Rates

| Dataset | ICL Setting | Model | | | | | |
|---|---|---|---|---|---|---|---|
| | | Llama3.2-3B | Qwen2.5-3B | Llama3-8B | Qwen2.5-7B | Llama3-70B | Qwen2.5-72B |
| **EDOS** | 0-shot | 3 (0) | 0 (0) | 152 (151) | 0 (0) | 1 (0) | 3,375 (0) |
| | amb-ordered | 10 (0) | 42 (0) | 1 (0) | 0 (0) | 1 (0) | 38 (0) |
| | amb-shuffled | 11 (0) | 4 (0) | 2 (0) | 0 (0) | 1 (0) | 32 (0) |
| | diff-ordered | 18 (0) | 35 (0) | 1 (0) | 0 (0) | 1 (0) | 20 (0) |
| | diff-shuffled | 9 (0) | 105 (0) | 1 (0) | 0 (0) | 1 (0) | 20 (0) |
| | random-ordered | 37 (0) | 0 (0) | 1 (0) | 0 (0) | 1 (0) | 25 (0) |
| | random-shuffled | 9 (0) | 8 (0) | 1 (0) | 0 (0) | 1 (0) | 33 (0) |
| **Brexit** | 0-shot | 0 (0) | 0 (0) | 1 (1) | 0 (0) | 0 (0) | 98 (0) |
| | amb-ordered | 0 (0) | 50 (0) | 0 (0) | 0 (0) | 0 (0) | 1 (0) |
| | amb-shuffled | 0 (0) | 17 (0) | 0 (0) | 0 (0) | 0 (0) | 1 (0) |
| | diff-ordered | 1 (0) | 11 (0) | 0 (0) | 0 (0) | 0 (0) | 1 (0) |
| | diff-shuffled | 0 (0) | 0 (0) | 0 (0) | 0 (0) | 0 (0) | 1 (0) |
| | random-ordered | 1 (0) | 37 (0) | 0 (0) | 0 (0) | 0 (0) | 1 (0) |
| | random-shuffled | 0 (0) | 18 (0) | 0 (0) | 0 (0) | 0 (0) | 0 (0) |
| **GAB** | 0-shot | 11 (2) | 1 (0) | 38 (36) | 1 (0) | 1 (0) | 3,806 (0) |
| | amb-ordered | 5 (0) | 3 (0) | 1 (0) | 1 (0) | 1 (0) | 20 (0) |
| | amb-shuffled | 3 (0) | 2 (0) | 1 (0) | 1 (0) | 1 (0) | 33 (0) |
| | diff-ordered | 8 (0) | 3 (0) | 1 (0) | 1 (0) | 1 (0) | 1,638 (0) |
| | diff-shuffled | 5 (0) | 2 (0) | 1 (0) | 1 (0) | 1 (0) | 1 (0) |
| | random-ordered | 15 (0) | 5 (1) | 1 (0) | 1 (0) | 1 (0) | 78 (0) |
| | random-shuffled | 7 (0) | 3 (0) | 1 (0) | 1 (0) | 1 (0) | 16 (0) |
| **MD** | 0-shot | 68 (2) | 0 (0) | 26 (26) | 0 (0) | 0 (0) | 2,594 (0) |
| | amb-ordered | 23 (0) | 960 (0) | 0 (0) | 0 (0) | 0 (0) | 4 (0) |
| | amb-shuffled | 15 (0) | 1,400 (0) | 0 (0) | 0 (0) | 0 (0) | 6 (0) |
| | diff-ordered | 26 (0) | 1,532 (0) | 0 (0) | 0 (0) | 0 (0) | 3 (0) |
| | diff-shuffled | 22 (0) | 1,163 (0) | 0 (0) | 1 (0) | 0 (0) | 6 (0) |
| | random-ordered | 24 (0) | 1,550 (0) | 0 (0) | 1 (0) | 0 (0) | 3 (0) |
| | random-shuffled | 12 (0) | 1,375 (0) | 0 (0) | 0 (0) | 0 (0) | 4 (0) |
| **SBIC** | 0-shot | 3 (0) | 3 (0) | 249 (247) | 2 (0) | 2 (0) | 4,069 (0) |
| | amb-ordered | 7 (0) | 177 (0) | 2 (0) | 2 (0) | 2 (0) | 19 (0) |
| | amb-shuffled | 5 (0) | 30 (0) | 2 (0) | 3 (0) | 2 (0) | 12 (0) |
| | diff-ordered | 9 (0) | 30 (0) | 2 (0) | 3 (0) | 2 (0) | 53 (0) |
| | diff-shuffled | 4 (0) | 2 (0) | 2 (0) | 3 (0) | 2 (0) | 38 (0) |
| | random-ordered | 6 (0) | 15 (0) | 2 (0) | 2 (0) | 2 (0) | 13 (0) |
| | random-shuffled | 7 (0) | 6 (0) | 2 (0) | 3 (0) | 2 (0) | 21 (0) |

Table B: Overview of models' failure to provide an answer (absolute numbers). In brackets we report the number of refused answers.