# Language Models for Text Classification: Is In-Context Learning Enough?

**Aleksandra Edwards, Jose Camacho-Collados**
Cardiff University
Senghennydd Rd, Cardiff CF24 4AG
{edwardsai, camachocolladosj}@cardiff.ac.uk

**Abstract**

Recent foundational language models have shown state-of-the-art performance in many NLP tasks in zero- and few-shot settings. An advantage of these models over more standard approaches based on fine-tuning is the ability to understand instructions written in natural language (prompts), which helps them generalise better to different tasks and domains without the need for specific training data. This makes them suitable for addressing text classification problems for domains with limited amounts of annotated instances. However, existing research is limited in scale and lacks understanding of how text generation models combined with prompting techniques compare to more established methods for text classification such as fine-tuning masked language models. In this paper, we address this research gap by performing a large-scale evaluation study for 16 text classification datasets covering binary, multiclass, and multilabel problems. In particular, we compare zero- and few-shot approaches of large language models to fine-tuning smaller language models. We also analyse the results by prompt, classification type, domain, and number of labels. In general, the results show how fine-tuning smaller and more efficient language models can still outperform few-shot approaches of larger language models, which have room for improvement when it comes to text classification.

**Keywords:** in-context learning, large language models, text classification

## 1. Introduction

A standard approach for supervised text classification is fine-tuning language models such as BERT using an additional classifier head (Radford et al., 2018; Dong et al., 2019; Devlin et al., 2018; Yin et al., 2019; Viswanathan et al., 2023; Mosbach et al., 2023). However, these approaches require large amounts of data to achieve state-of-the-art results (Edwards et al., 2022) which makes them unsuitable for classification tasks associated with class imbalances and data sparsity (Giridhara et al., 2019; Zhang and Wu, 2015; Türker et al., 2019; Yin et al., 2019). These problems often occur in real world applications where annotation of data can be performed only by scarce domain experts such as medical and legal domains or applications with highly imbalanced classes such as crime data and fraud detection (Giridhara et al., 2019; Zhang and Wu, 2015; Türker et al., 2019).

Recent advances in Natural Language Processing (NLP) lead to the emerge of an alternative approach based on using autoregressive text generation models (Radford et al., 2019) that have zero- and few- shot capabilities and perform unseen tasks through the use of prompting (Schick and Schütze, 2021a; Radford et al., 2019; Le Scao and Rush, 2021; Viswanathan et al., 2023; Plaza-del Arco et al., 2023). The ability of these models to understand natural language instructions let them generalise to different domains and tasks without the need of large training corpora (Plaza-del Arco et al., 2023). There have been even further improvements in the performance of these models in zero-shot settings by fine-tuning them on sets of instructions (task descriptions) (Raffel et al., 2020).

The promising results of these models against various benchmark datasets (Wang et al., 2022b; Liu et al., 2023; Bang et al., 2023) led to increased research into developing methods, mainly based on prompt engineering techniques (Viswanathan et al., 2023; Le Scao and Rush, 2021) for improving their generalisation capabilities. Further, there has been an increased attention into evaluating the suitability of these models for more specialised domains such as the legal, medical, and financial domain (Sarkar et al., 2021; Chalkidis et al., 2020; Yin et al., 2019; Labrak et al., 2023). However, most of the proposed approaches are domain- and task-specific. There is lack of understanding of how these models perform in comparison to more established approaches for text classification. In general, analyses are performed for a small range of model types, domains, and tasks.

Our work is the first attempt to systematically compare how text generation models using zero-shot and one-shot learning compare to more established but data-consuming approaches for classification based on fine-tuning language models. Our goal is to identify how well current large language models (LLMs) can adapt to different text classification tasks and domains given limited in-



*LREC-COLING 2024*, pages 10058–10072
20-25 May, 2024. 

formation, and outline the potential strengths and weaknesses of these models. For these purposes, we evaluate five heterogeneous models of different sizes, including traditional masked language models and more recent autoregressive LLMs. Our analyses span over 16 datasets from 7 domains representing binary, multiclass, and multilabel classification.

Our main contributions are as follows. First, we explore an important but understudied problem of how suitable the newly developed text generation models such as LLaMA, Flan-T5, T5, and ChatGPT are for text classification in few-shot settings compared to lighter models that require training data such as RoBERTa or FastText. In addition to the performance, our analysis helps identify specific strengths and weaknesses of each type of model. Second, in contrast to the majority of existing research focusing on optimisation techniques for prompt creation, we analyse trends in the model's performance that are non-prompt sensitive as well as look at how the amount of specificity provided in the prompt regarding the task and the domain affect the performance of the models. Third, we evaluate generalisation abilities of models for 7 domains, including real-world specialised domains, such as legal, medical, and crime data. We also analyse how the models' behaviour changes for datasets used in the pre-training stage versus when testing on unseen datasets.

## 2. Related Work

We first introduce the different types of methods and models used for text classification along with their strengths and weaknesses (see Section 2.1). Then, we discuss relevant work on comparing prompting and fine-tuning approaches for text classification as well as outline challenges and research gaps within existing work (see Section 2.2).

### 2.1. Text Classification

We distinguish between three main approaches for text classification, linear methods (described in Section 2.1.1), fine-tuning language models (Section 2.1.2) and prompting techniques combined with text generation models (Section 2.1.3).

#### 2.1.1. Linear Methods

FastText (Joulin et al., 2017) is a linear text classification model which provides a strong baseline for many text classification tasks and gives performance comparable to state-of-the-art methods, including language models such as BERT (Zhou, 2020; Edwards et al., 2020). It integrates a linear model with a rank constraint which allows sharing parameters among classes and features. It also integrates word embeddings which are averaged into a text. These features help address many problems associated with other linear models such as out-of-vocabulary words and fine-grained distinctions between classes.

#### 2.1.2. Fine-tuning Methods

Language models like BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019), pre-trained using a masked language modeling (MLM) objective provide a state-of-the-art performance against most standard NLP benchmarks (Wang et al., 2019, 2018). These models can be easily adapted for text classification by using fine-tuning techniques which are based on adding a single classification layer onto the model. However, fine-tuning techniques require large amounts of data to be adapted to targeted tasks and domains which makes them impractical for low resource classification tasks (Strubell et al., 2019; Peng et al., 2021; Lu et al., 2021).

#### 2.1.3. Text Generation Models

Recent advances in NLP have led to the development of bigger models composed of billion of parameters which have shown an improved performance especially in text generation and low resource settings (Zhang et al., 2019; Black et al., 2022; Labrak et al., 2023). These text generation models such as GPT and subsequent releases (Brown et al., 2020; Radford et al., 2018, 2019) as well as LLaMA (Touvron et al., 2023a,b) and T5 (Raffel et al., 2020) can understand natural language instructions (i.e., prompts) and thus can generalise to unseen tasks and domains without the need for large computational and data resources (Brown et al., 2020). Further progress has been made by fine-tuning these models on a set of natural language instructions, consisting of descriptions of the tasks and the expected output (Efrat and Levy, 2020; Mishra et al., 2022). This enables models to generalise even better to tasks, domains, and languages (Ouyang et al., 2022; Wei et al., 2021; Sanh et al., 2022; Efrat and Levy, 2020; Mishra et al., 2022).

The ability of text generation models to make predictions with little or no training makes these models particularly suitable for tackling the problem of data scarcity for text classification (Wang et al., 2020; Gupta et al., 2020). Therefore, much of the approaches in zero- and few-shot learning are focused at optimising the performance of these models mainly through the use of prompting (Gera et al., 2022; Le Scao and Rush, 2021; Deng et al., 2022; Schick and Schütze, 2021a; Radford et al., 2019; Le Scao and Rush, 2021; Viswanathan et al., 2023; Plaza-del Arco et al., 2023).

### 2.2. Prompting versus Fine-Tuning

Prompting in zero- and few- shot settings, also known as in-context learning (ICL), is the process of providing natural language instructions that describe a task as an input to a language model, including the expected output (Labrak et al., 2023). In few-shot prompting, the model is presented with some training examples along with the task instructions. In contrast to fine-tuning techniques, prompting does not involve changing the weights of the model which makes the approach less resource consuming. Additionally, previous research has suggested that prompting can lead to comparable or even better performance than standard fine-tuning techniques (Gao et al., 2021; Mosbach et al., 2023). A drawback of this approach is the models' sensitivity to the prompts where slight changes of the instruction can lead to big differences in the performance (Schick and Schütze, 2021b; Le Scao and Rush, 2021; Sun et al., 2023). Thus, much of the work on text generation models is focused on prompt optimisation techniques based on automatic generation for prompts (Wang et al., 2022a; Shin et al., 2020), quantifying the benefits of prompting (Schick and Schütze, 2021b; Le Scao and Rush, 2021), and improving the generalisation abilities of prompts (Zhang et al., 2022; Schönfeld et al., 2019; Song et al., 2021; Wang et al., 2022a; Oniani et al., 2023; Sun et al., 2023)

There has been an increased research into evaluating and improving the performance of text generation models for zero and few shot classification in more specialised domains such as the legal, medical, and financial domains (Ge et al., 2022; Sarkar et al., 2021; Chalkidis et al., 2020). Labrak et al. (2023) evaluate four state-of-the-art instruction-tuned large language models (ChatGPT, Flan-T5 UL2, Tk-Instruct, and Alpaca) on a set of 13 real-world clinical and biomedical natural NLP tasks, including text classification. The results show that instruction-tuned models tend to be outperformed by a specialised model trained for the medical field such as PubMedBERT (Gu et al., 2021). This rises questions into the suitability of text generation models and prompting techniques for more specialised domains which require domain experts for annotation. Another research by Mosbach et al. (2023) conducts a comparison between fine-tuning and prompting techniques for two text classification datasets showing that both approaches have similar performance, although with a large variation in results depending on properties such as model size and number of examples. These works show that adapting these models to tasks, especially text classification for more specialised domains, remains a challenge.

The variance in performance between tasks and models depending on the prompt design makes the generalisation of text generation models a challenging problem. The small scale on which analyses are performed does not give enough knowledge on how well prompting techniques compare to the more established models for classification across different text classification types and more challenging unfamiliar domains. In this paper, we address these challenges by performing a large-scale comparison between different model types across a wider range of classification tasks and domains.

## 3. Experimental Setting

### 3.1. Datasets

For our experiments we selected a suite of datasets representing all three classification types, i.e., binary, multiclass, and multilabel. The datasets span across 7 domains and 13 classification tasks. Specifically, we selected the Twitter datasets from the SemEval 18 on emoji prediction (Barbieri et al., 2018), SemEval 18 on irony Detection (Van Hee et al., 2018), SemEval 19 on hate detection (Basile et al., 2019), SemEval 19 on offense detection (Zampieri et al., 2019), and SemEval 19 on sentiment analysis (Nakov et al., 2019). Further, we include datasets for topic categorisation such as BBC news[1], AG News (Zhang et al., 2015), Reuters (Lewis et al., 2004), and 20 Newsgroups (Lang, 1995) , as well as IMDB reviews dataset for polarity detection (Maas et al., 2011), PCL dataset for patronising language detection (Perez Almendros et al., 2020), and Toxic comments (Hosseini et al., 2017). Additionally, we evaluate models for more specialised domains representing real world applications such as EU legislation documents (Chalkidis et al., 2019) for legal legislation concepts detection, Hallmarks of cancer (Baker et al., 2015) for detecting cancer hallmarks, Ohsumed (Joachims, 1998) for cardiovascular diseases detection, and Safeguarding reports (Edwards et al., 2022) for theme detection. Additionally, we perform prediction for the top classes as well as the sub-classes of the 20 Newsgroups and Safeguarding datasets. In this way, we can analyse how the models performance is affected by the number of classes. The main features and statistics of each dataset are summarized in Table 1. For the EU legislation documents we have performed experiments with the 10 most frequent labels, similarly to Chalkidis et al. (2019). For the Ohsumed dataset, we have selected the top 23 most frequent classes, similar to prior work (Pilehvar et al., 2017).

[1] http://mlg.ucd.ie/datasets/bbc.html

| Dataset | Domain | Task | Type | Class Type | Avg tokens | Labels | # Train | # Dev | # Test |
|---|---|---|---|---|---|---|---|---|---|
| SemEval 18 (Emoji) | Twitter | Emoji Prediction | Sentence | multiclass | 12 | 20 | 45,000 | 5,000 | 50,000 |
| SemEval 18 (Irony) | Twitter | Irony Detection | Sentence | binary | 13 | 2 | 2,862 | 955 | 784 |
| SemEval 19 (Hateval) | Twitter | Hateval | Sentence | binary | 18 | 2 | 9,000 | 1,000 | 2,970 |
| SemEval 19 (OffensEval) | Twitter | OffensEval | Sentence | binary | 19 | 2 | 11,916 | 1,324 | 860 |
| SemEval 17 (Sentiment) | Twitter | Sentiment Analysis | Sentence | multiclass | 20 | 3 | 45,389 | 2,000 | 11,906 |
| BBC news | News | Topic categorisation | Document | multiclass | 220 | 5 | 1602 | 178 | 445 |
| Reuters | News | Topic categorisation | Document | multiclass | 83 | 8 | 6120 | 680 | 2659 |
| AG News | News | Topic categorisation | Document | multiclass | 31 | 4 | 103,346 | 11,482 | 5,928 |
| 20 Newsgroups | News | Topic categorisation | Document | multiclass | 285 | 6 | 9,857 | 1,095 | 7,290 |
| 20 Newsgroups | News | Topic categorisation | Document | multiclass | 285 | 20 | 9,857 | 1,095 | 7,290 |
| IMDB reviews | Reviews | Polarity Detection | Document | binary | 231 | 2 | 25200 | 2,800 | 25,601 |
| Ohsumed | Medical | Cardiovascular diesese det. | Document | multiclass | 104 | 23 | 9,390 | 1,043 | 12733 |
| Toxic Comments | Wikipedia | Toxic prediction | Document | multilabel | 46 | 7 | 143,614 | 15,957 | 63,978 |
| PCL dataset | News | Patronising language det. | Document | multilabel | 37 | 7 | 517 | 57 | 419 |
| EU legislation documents | Legislation | Legal legislation concept det. | Document | multilabel | 27 | 10 | 45,000 | 6,000 | 6,000 |
| Hallmarks of cancer | Medical | Hallmarks of cancer detection | Sentence | multilabel | 22 | 10 | 12,456 | 1,384 | 3624 |
| Safeguarding reports | Safeguarding | Theme detection | Sentence | multilabel | 18 | 5 | 5,719 | 635 | 3496 |
| Safeguarding reports | Safeguarding | Theme detection | Sentence | multilabel | 18 | 10 | 5,719 | 635 | 3496 |

Table 1: Overview of the classification datasets used in our experiments.

### 3.2. Comparison Models

We compare three main types of models: generative language models, masked language models, and linear models, all described below.

**Generative Language Models.** We include LLaMA 1 (Touvron et al., 2023a) and 2 (Touvron et al., 2023b) into the analysis as representatives of large auto-regressive generation models, both with 7 billion parameters. As a representative of smaller but instruction-tuned model, we use Flan-T5 (Chung et al., 2022). The model is fine-tuned using the Flan instruction tuning tasks collection (Chung et al., 2022). We use the large Flan-T5 model with 780M parameters. We have also included T5 model (Raffel et al., 2020) into our analysis which we fine-tune, similarly to RoBERTa. In particular, we use T5 base model. We have downloaded the models from Hugging Face (Wolf et al., 2019). As a representative of the GPT family of autoregressive models (Brown et al., 2020), we use OpenAI GPT 3.5-Turbo for our analysis. We added this model for completeness. However, given budget constraints and its closed nature for which few conclusions can be drawn, we only provide results for a sample of all datasets.

**Masked Language Models.** As a representative of masked language model, we use RoBERTa (Liu et al., 2019), pre-trained on English language. It is known to achieve state-of-the-art results for many text classification tasks. We perform experiments with RobERTa base (125 million parameters) and RoBERTa large (354 million parameters) models to allow analysis into the effect of model size over the classification performance. We have downloaded the models from Hugging Face (Wolf et al., 2019).

**Linear Models.** Finally, we use FastText (Joulin et al., 2017) (see Section 2.1.1) as a representative of a linear text classification model. Despite its simplicity the model provides a strong baseline for many text classification tasks and it is known to give comparable results to state-of-the-art methods, including language models such as BERT for some classification problems (Zhou, 2020; Edwards et al., 2020).

### 3.3. Prompting, Training and Evaluation

As mentioned in Section 1, our aim is to estimate how well the text generation models perform for text classification when compared to the more data consuming models such as RoBERTa and FastText. Therefore, we perform experiments for Flan-T5 and LLaMA in zero- and one- shot ICL settings. For zero shot, we provide information about the task to the model through prompting. For one shot, we randomly select a single training instance per label and we provide these examples along with the instruction to the model. To ensure robustness, the random selection of training samples is performed for three iterations and the results are averaged. For generating labels for the test sequences, we use default model settings. We judge the outputs as expected class labels or not by simply checking whether the output of the model matches one of the labels for the given classification task. We experiment with three different prompts which we describe further in Section 3.4.
As for RoBERTa, we fine-tune it for the classification task on the training data of each dataset using a sequence classifier, a learning rate of 2e-5 and 4 epochs. In particular, we made use of RoBERTa's Hugging Face default transformers implementation for classifying sentences (Wolf et al., 2019). As for T5, we fine-tune it using conditional generation, 2 epochs, and learning rate of 5e-5. Finally, we use FastText classifier with 25 epochs and softmax as the loss function.
Finally, we report results based on standard micro and macro averaged F1 (Yang, 1999).

### 3.4. Prompt Design

Our paper does not focus on identifying and describing most efficient prompt engineering prac-

tices (as majority of work described in Section 2) but instead we focus on highlighting prompt-independent trends in the models performance in order to help outline advantages and disadvantages of out-of-the-box approaches for few shot text classification. We selected instructions that led to satisfactory results in previous research or have been used in the training set for the instruction-tuned models Flan-T5 (Sun et al., 2023; Wei et al., 2021). These prompts vary in the detail they provide about the given task and domain. We want to analyse trends across models behaviour that are non-prompt sensitive as well as look at how the amount of specificity provided in the prompt affect the performance of the models. For these purposes, we use the following three prompts: (1) ***generic:*** a prompt which does not give information about the task or domain, used in (Sun et al., 2023); (2) ***task:*** describes the given task, i.e., classification; (3) ***domain:*** a prompt which gives more information about the domain, for instance, it specifies the type of test data, such as an article or tweet. We have created the domain-based prompts following examples provided in Wei et al. (2021). Table 2 presents examples of the prompts per classification type[2].

| | **Binary** | **Multiclass** | **Multilabel** |
|---|---|---|---|
| **generic** | Choose your answer: According to the above paragraph, the question *'Is the text ironic?'*: | Pick one category for the following text. The options are: | Pick one or more from the categories for the following text.The options are: |
| **task** | Classify the input text into one of the following categories: | Classify the input text into one of the following categories: | Classify the input text using one or more from the following categories: |
| **domain** | Is the Tweet classified as irony or non-irony? | Select the topic that the given news is about. The topics are - | Which of the given toxic topics best describe the given comment? Choose one or more from the following topics: |

Table 2: Examples of prompts used for zero- and one-shot learning for Flan-T5 and LLaMA.

## 4. Results and Analysis

The aim of our analysis is (1) identify if and how the use of prompts affect the performance of text generation models (see Section 4.1); (2) compare performance of prompting and fine-tuning techniques in order to identify strengths and weaknesses of the different models – we focus on a comparison between the three types of classification, i.e., binary, multiclass, and multilabel (see Section 4.2); and (3) perform a fine-grained analysis comparing models' performance at the domain and dataset level (see Section 4.3). In addition to this general comparison, we analyse separately the performance of closed-source GPT3.5 and models for the 'IMDB reviews' and 'AG News' datasets as they are used in the fine-tuning of the Flan-T5 model.

### 4.1. Model and Prompt Analysis

A comparison between the two LLaMA models shows an advantage of LLaMA 2 over LLaMA 1 for both zero- and one-shot settings across all prompt types (see Figure 1 and Table 3). The two models have similar performance in the zero-shot setting in terms of F1 score. However, the number of wrong labels for LLaMA 1 is much larger with 0.470 wrong labels compared to the 0.100 wrong labels from LLaMA2. Results in Figure 1 also show a clear advantage of Flan-T5 over the other models for all three prompts in terms of micro- and macro- F1 for both zero- and one- shot settings. The Flan-T5 model also leads to smaller number of wrong labels in zero-shot prompting. This suggests that smaller but instruction-tuned models can be more beneficial in zero- and few- shot classification in comparison to larger text generation models. Specifically, Flan-T5 has on average 0.110 improvement in micro- and macro-F1 for both zero- and one- shot settings over LLaMA 2.

Further analysis into the prompts reveal that prompt choice does not lead to significant changes in the models behaviour where the deviation for the three prompts across all models is relatively small. For instance, for LLaMA 1 and LLaMA 2 is less than 0.02 difference in micro-F1 for both zero- and one- shot settings while for Flan-T5 it gradually decreases from 0.07 in zero-shot to 0.01 for one-shot. This suggests that smaller models such as Flan-T5 are more sensitive to the prompt in zero settings versus few shot learning. The benefits from one-shot prompting are evident across all three models where the F1 measure tends to increase and the number of wrong labels decreases. Flan-T5 improves its performance on a higher rate compared to to the other two models with around 0.047 increase in the micro-F1 score versus 0.027 increase for LLaMA 2. This illustrates the strong abilities of these models to learn tasks with minimal amount of training data.

### 4.2. Prompting versus Fine-tuning

Results in Figure 2 show the same trends for prompting methods where Flan-T5 outperforms LLaMA 1 and LLaMA 2 for all text classification types in terms of micro- and macro-F1. All three models improve their performance for one-shot prompting regarding the number of wrong labels. In one shot setting, Flan-T5 and LLaMA 2 tend to have close to 0 wrong labels with LLaMA 2 returning slightly lower number of irrelevant results,

[2]A list of all prompts is given in the Appendix.

| Model | Prompt | zero shot | | | one shot | | | all | |
|---|---|---|---|---|---|---|---|---|---|
| | | micro F1 | macro F1 | missing labs | micro F1 | macro F1 | missing labs | micro F1 | macro F1 |
| Flan-T5 | generic | .510 | .459 | .076 | .446 | .401 | .020 | – | — |
| | task | .368 | .373 | .055 | .462 | .415 | .012 | – | — |
| | domain | .369 | .302 | .092 | .480 | .432 | .072 | – | — |
| | AVG | .416 | .378 | .074 | .463 | .416 | .035 | — | — |
| LLaMA 1 | generic | .309 | .213 | .484 | .274 | .274 | .043 | – | — |
| | task | .319 | .230 | .471 | .339 | .303 | .414 | – | — |
| | domain | .284 | .235 | .463 | .318 | .270 | .066 | – | — |
| | AVG | .304 | .279 | .469 | .311 | .267 | .038 | – | — |
| LLaMA 2 | generic | .332 | .282 | .086 | .305 | .253 | .436 | – | — |
| | task | .286 | .238 | .061 | .333 | .282 | .679 | – | — |
| | domain | .309 | .269 | .153 | .360 | .322 | .007 | – | — |
| | AVG | .309 | .263 | .100 | .336 | .288 | .006 | – | — |
| T5 | — | – | — | — | .134 | .109 | .851 | .702 | .625 |
| RoBERTa (base) | — | – | — | — | .273 | .207 | — | .707 | .625 |
| RoBERTa (large) | — | – | — | — | .338 | .278 | — | .727 | .657 |
| fastText | — | — | — | — | .254 | .164 | — | .505 | .419 |

Table 3: Prompt Analysis where Micro-F1 and Macro-F1 results averaged across all datasets, comparing the performance of Flan-T5, LLaMA 1, and LLaMA 2 models for all three types of prompts, i.e., *'generic'*, *'task'*, and *'domain'* as well as the average ('AVG') between them. 'Missing labs' shows the fraction of results returned by the three models that are different from the classification labels. Results are displayed for zero-shot ('zero') and one-shot setting ('one').

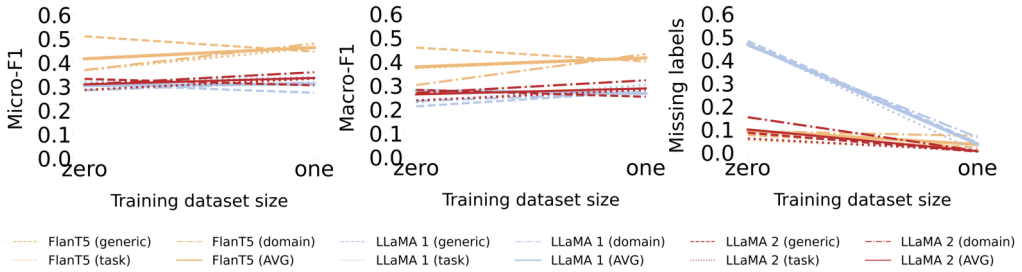


Figure 1: Micro-F1 (left) and Macro-F1 (middle) results averaged across all datasets, comparing the performance of Flan-T5, LLaMA 1, and LLaMA 2 models for all three types of prompts, i.e., *'generic'*, *'task'*, and *'domain'* as well as the average ('AVG') between them. 'Missing label' (right) shows the fraction of results returned by the three models that are different from the classification labels. Results are displayed for zero-shot ('zero') and one-shot setting ('one').

while Flan-T5 has a better F1 score (see Figures 2 and 3[3]). The advantage of LLaMA 2 over LLaMA 1 is clearly shown for all classification tasks, especially binary and multilabel where LLaMA 2 has a smaller number of irrelevant results and higher F1 score (see Figure 2).

Regarding fine-tuning approaches, results in Figure 2 show a clear dominance of RoBERTa-large in one-shot setting for all classification types. When fine-tuning is performed using the entire dataset, T5 outperforms the rest of the models for binary classification with micro-F1 = 0.672 versus RoBERTa-large with micro-F1 = 0.607. However, for multiclass and multilabeling tasks, the performance of T5 decreases and the model is outperformed by both RoBERTa-base and RoBERTa-large. For instance, for multiclass problems RoBERTa-large achieves micro-F1 of 0.726 versus micro-F1 for T5 = 0.700. For multilabeling problems the performance gap between the models increases and RoBERTa-large has a micro-F1 = 0.788 versus T5 with micro-F1 = 0.718. These results suggest that fine-tuned masked language models are more suitable for complex classification tasks such as multiclass and multilabeling problems when the number of labels is higher versus fine-tuning text-to-text models such as T5.

A comparison between prompting and fine-tuning techniques for low resource settings suggests a better performance of prompting for binary and multiclass problems (see Figure 2) where Flan-T5 and LLaMA 2 outperform fine-tuning models by a significant margin. For instance, Flan-T5 has micro-F1 = 0.553 versus micro-F1 for RoBERTa-large with micro-F1 = 0.485 for binary classification in one shot settings. The advantage of prompting in one shot settings becomes even more evident for multiclass problems where Flan-T5 achieves micro-F1 = 0.489 versus RoBERTa-large with micro-F1 = 0.162. However, for multilabeling problems, fine-tuning approaches outper-

[3]Macro-F1 results are available in the Appendix.

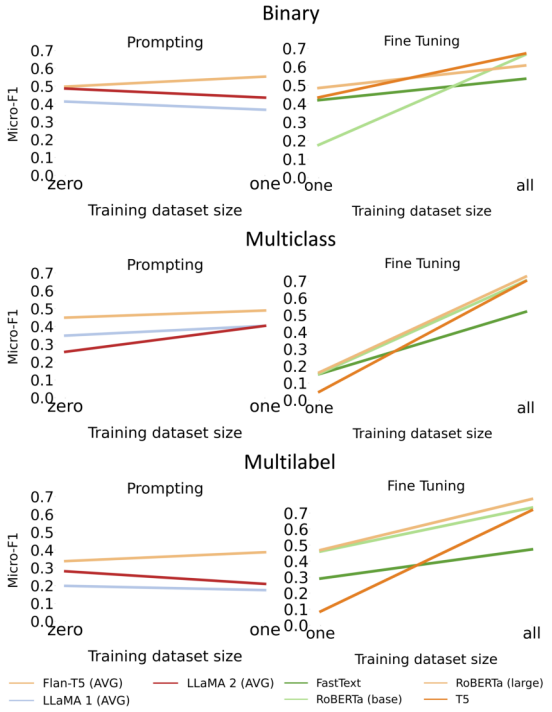


Figure 2: Comparison between prompting (left) and fine-tuning (right) approaches per text classification type where 'AVG' refers to averaged results across all prompt types per model. In 'Prompting', 'zero' and 'one' refer to zero- and one- shot prompt-based learning techniques, in 'Fine Tuning', 'one' refers to fine-tuning the models with one training instance per label and 'all' refers to fine-tuning using the entire dataset.

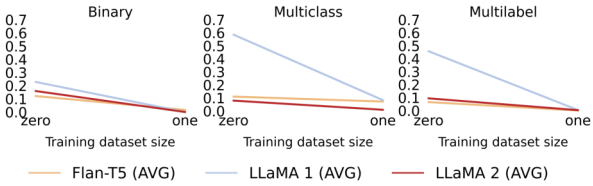


Figure 3: Wrong labels for prompting approaches per binary (left), multiclass (middle), and multilabel (right) classification where 'zero' refers to zero-shot learning and 'one' refers to one-shot learning.

form prompting methods with a difference in micro-F1 of 0.082 between the best fine-tuned model, RoBERTa-large, and the best prompting model, i.e. Flan-T5. It is worth noting that during one-shot training, all models have been provided the same training examples. However, further analyses are needed to identify most efficient ways for representing multi-labeling problems as part of prompting techniques.

Despite the better overall performance of prompting techniques in zero- and one- shot settings, these approaches lead to unsatisfactory performance when compared to fine-tuned masked language models on a larger training set. Further, the difference in the performance between the two techniques grows larger for more complex text classification tasks such as multiclass and multi-labeling problems. For instance, for binary classification, the difference in performance in terms of micro-F1 between best performing prompting and fine-tuning technique is 0.119 while for multiclass the difference in performance is 0.240. This shows that large autoregressive text generation models coupled with few shot learning techniques still have room for improvement when it comes to text classification. Fine-tuned masked language models, despite being smaller, lead to better performance for text classification versus LLMs in ICL settings.

### 4.3. Trends across datasets and models

Results presented in Table 4 confirm findings from Section 4.2 showing a clear dominance of Flan-T5 over LLaMA for zero- and one-shot prompting for the majority of datasets. Exceptions are the 'irony', 'sentiment', and 'PCL' datasets where LLaMA performs better for either zero or one shot setting, or both. For some datasets such as 'hate', prompting models give better performance in zero- shot than one-shot setting. However, models still improve performance for these datasets in terms of number of wrong labels. Further, the choice of one shot training instances can influence the performance of models in few-shot learning. For the purposes of this analysis we have selected the one shot examples randomly. Analysing the impact of the training examples in few-shot learning can be a future research direction which we leave for future work.

In contrast to the prompting approaches, results for the fine-tuned models do not show a clear dominance of either RoBERTa or T5. T5 shows a better performance for the majority of the binary classification tasks (those associated with Twitter datasets) as well as the datasets 'AG news', '20 News' (top 6 classes)', and the 'legal' domain. The two models attain a similar macro-F1 for the emoji prediction and safeguarding reports datasets.

**Impact of the number of labels.** Analysis into the effect of the number of classification labels in the performance shows an interesting trend with the fine-tuned models (RoBERTa and T5) performing slightly better for classification tasks with 6 to 9 labels than classification with less labels (see Figure 4). For RoBERTa this trend occurs for both micro-F1 and macro-F1 while for T5 it appears only for micro-F1. This can be attributed to the nature of the binary classification tasks ('irony', 'offense', 'hate') which express human emotions

| Dataset | Model | zero shot | | | one shot | | | all | |
|---|---|---|---|---|---|---|---|---|---|
| | | micro F1 | macro F1 | wrong labs | micro F1 | macro F1 | wrong labs | micro F1 | macro F1 |
| irony | RoBERTa | – | – | – | .459 (±.005) | .459 (±.005) | – | .508 | .508 |
| | T5 | — | — | — | .455(±.021) | .455(±.021) | .589 | **.688** | **.688** |
| | FlanT5 | .428 | .428 | .049 | **.491(±.034)** | **.491(±.034)** | .009 | – | – |
| | LLaMA | **.499** | **.499** | .214 | .443 (±.003) | .443 (±.003) | .000 | – | – |
| | GPT 3.5* | .727 | .727 | .000 | – | – | – | – | – |
| offense | RoBERTa | – | – | – | .550 (±.143) | .550 (±.143) | – | .705 | .705 |
| | T5 | – | – | – | .462 (±.001) | .462 (±.001) | .864 | **.709** | **.709** |
| | FlanT5 | **.429** | **.429** | .269 | **.558(±.019)** | **.558(±.019)** | .003 | – | – |
| | LLaMA | .419 | .419 | .227 | .347 (±.026) | .347 (±.026) | .001 | – | – |
| | GPT 3.5* | .635 | .635 | .000 | – | – | – | – | – |
| hate | RoBERTa | – | – | – | .445 (±.118) | .445 (±.118) | – | .607 | .607 |
| | T5 | — | — | – | .386 (±.312) | .386 (±.312) | .732 | **.619** | **.619** |
| | FlanT5 | **.634** | **.634** | .004 | **.611 (±.006)** | **.611 (±.006)** | .005 | – | - |
| | LLaMA | .539 | .539 | .004 | .514 (±.111) | .514 (±.111) | .000 | – | – |
| emoji | RoBERTa | — | — | – | .047 (±.009) | .005 (±.001) | – | **.366** | **.317** |
| | T5 | — | — | — | .000 (±.000) | .000 (±.000) | .100 | .259 | **.317** |
| | FlanT5 | **.059** | **.042** | .036 | **.114(±.021)** | **.082(±.007)** | .006 | – | – |
| | LLaMA | .060 | .041 | .091 | .033(±.036) | .020(±.017) | .001 | – | – |
| sentiment | RoBERTa | — | — | – | .449 (±.108) | .271 (±.008) | – | **.714** | **.714** |
| | T5 | — | – | – | .312 (±.121) | .272 (±.078) | .563 | .708 | .709 |
| | FlanT5 | **.459** | **.402** | .109 | .417 (±.004) | .381 (±.007) | .000 | – | – |
| | LLaMA | .369 | .334 | .027 | **.482 (±.037)** | **.402 (±.143)** | .000 | – | – |
| BBC | RoBERTa | — | — | – | .217 (±.027) | .112 (±.029) | – | **.989** | **.989** |
| | T5 | — | — | – | .001 (±.001) | .001 (±.001) | .999 | .977 | .977 |
| | FlanT5 | **.922** | **.867** | .096 | **.939(±.008)** | **.936(±.009)** | .038 | – | – |
| | LLaMA | .498 | .439 | .021 | .849 (±.098) | .843 (±.081) | .004 | – | – |
| | GPT 3.5* | .912 | .913 | .000 | – | – | – | – | – |
| Reuters | RoBERTa | — | — | – | .154 (±.111) | .054 (±.021) | – | **.939** | **.869** |
| | T5 | — | — | – | .010 (±.034) | .010 (±.067) | .990 | .929 | .833 |
| | FlanT5 | **.321** | **.334** | .334 | .467 (±.023) | **.504 (±.032)** | .017 | – | – |
| | LLaMA | .212 | .168 | .006 | **.528 (±.076)** | .304 (±.145) | .006 | – | – |
| | GPT 3.5* | .852 | .718 | .000 | – | – | – | – | – |
| 20 News(all) | RoBERTa | — | — | – | .190 (±.028) | .101 (±.019) | – | .859 | .853 |
| | T5 | — | — | – | .000 (±.000) | .000 (±.000) | .999 | **.861** | **.854** |
| | FlanT5 | **.564** | **.520** | **.001** | **.684 (±.008)** | **.654 (±.007)** | .057 | – | – |
| | LLaMA | .324 | .272 | .094 | .368 (±.034) | .300(±.079) | .001 | – | – |
| 20 News(subcl) | RoBERTa | — | — | – | .055 (±.013) | .015 (±.003) | – | **.741** | **.728** |
| | T5 | — | — | – | .000 (±.000) | .000 (±.000) | .990 | .717 | .693 |
| | FlanT5 | **.510** | **.507** | .000 | **.512 (±.013)** | **.501 (±.013)** | .011 | – | – |
| | LLaMA | .185 | .194 | .167 | .376 (±.015) | .342 (±.014) | .020 | – | – |
| Ohsumed | RoBERTa | — | — | – | .025 (±.019) | .002 (±.004) | – | **.476** | **.415** |
| | T5 | — | — | – | .002 (±.001) | .002 (±.001) | .958 | .452 | .362 |
| | FlanT5 | **.306** | **.283** | .194 | **.288 (±.003)** | **.241 (±.001)** | .375 | – | – |
| | LLaMA | .151 | .099 | .154 | .180 (±.110) | .162 (±.110) | .036 | – | – |
| Toxic | RoBERTa | — | — | – | .671 (±.005) | **.550** (±.003) | – | .899 | **.782** |
| | T5 | — | — | – | .020 (±.001) | .010 (±.011) | .989 | **.913** | .661 |
| | FlanT5 | **.629** | **.380** | .140 | **.710 (±.066)** | .262 (±.014) | .003 | – | – |
| | LLaMA | .331 | .142 | .211 | .005 (±.079) | .002 (±.077) | .004 | – | – |
| Legal | RoBERTa | — | — | – | .429 (±.030) | .285 (±.030) | – | .965 | .601 |
| | T5 | — | — | – | **.500** (±.037) | .125 (±.042) | .970 | **.982** | **.612** |
| | FlanT5 | **.251** | **.233** | .000 | .351 (±.047) | **.352 (±.028)** | .000 | – | – |
| | LLaMA | .224 | .167 | .069 | .269 (±.091) | .232 (±.175) | .005 | – | – |
| Cancer | RoBERTa | — | — | – | .309 (±.003) | .290 (±.002) | – | **.524** | **.414** |
| | T5 | — | — | – | .000 (±.000) | .000 (±.000) | .000 | .344 | .157 |
| | FlanT5 | **.296** | **.286** | .246 | **.361 (±.027)** | **.319 (±.017)** | .000 | – | – |
| | LLaMA | .249 | .178 | .141 | .168 (±.131) | .104 (±.098) | .004 | — | – |
| PCL | RoBERTa | — | – | – | **.555(±.004)** | **.518 (±.006)** | – | **.719** | **.592** |
| | T5 | — | — | – | .001 (±.001) | .001 (±.001) | .999 | .654 | .525 |
| | FlanT5 | .224 | .124 | .000 | .224 (±.008) | .141 (±.112) | .001 | – | – |
| | LLaMA | **.392** | **.303** | .050 | .287 (±.095) | .159 (±.114) | .000 | – | – |
| | GPT 3.5* | .207 | .117 | .000 | – | – | – | – | – |
| Safeguard(all) | RoBERTa | — | — | – | **.601 (±.011)** | **.589 (±.011)** | – | **.905** | **.895** |
| | T5 | — | — | – | .000 (±.000) | .000 (±.000) | .000 | .756 | .725 |
| | FlanT5 | **.347** | **.326** | .000 | .392 (±.007) | .360 (±.003) | .000 | – | – |
| | LLaMA | .291 | .233 | .041 | .286 (±.007) | .197 (±.003) | .001 | – | – |
| | GPT 3.5* | .369 | .340 | .000 | – | – | – | – | – |
| Safeguard(subcl) | RoBERTa | — | — | – | .247 (±.105) | .201 (±.102) | – | **.718** | .515 |
| | T5 | — | — | – | .010 (±.002) | .020 (±.002) | .969 | .657 | **.516** |
| | FlanT5 | **.275** | **.253** | .000 | **.281 (±.012)** | **.265 (±.009)** | .000 | – | – |
| | LLaMA | .195 | .176 | .051 | .237 (±.049) | .231 (±.089) | .006 | – | – |
| | GPT 3.5* | .359 | .366 | .012 | – | – | – | – | – |

Table 4: Micro-F1 and Macro-F1 results per dataset for RoBERTa (large), fine-tuned T5, Flan-T5, LLaMA 2, and GPT 3.5-Turbo. The ratio of wrongly-formatted outputs is included in the wrong labels (labs) column.The results for Flan-T5 and LLaMA 2 are based on averaged results across all prompts.

| Dataset | Model | zero shot | | | one shot | | | all | |
|---|---|---|---|---|---|---|---|---|---|
| | | micro F1 | macro F1 | wrong labs | micro F1 | macro F1 | wrong labs | micro F1 | macro F1 |
| IMDB | RoBERTa | — | — | – | .436 (±.311) | .436 (±0.311) | – | **.955** | **.955** |
| | T5 | — | — | — | .751 (±.065) | .751 (±.065) | .711 | .952 | .952 |
| | FlanT5 | **.948** | **.948** | .097 | **.900 (±.007)** | **.900 (±.007)** | .017 | – | – |
| | LLaMA | .628 | .628 | .219 | .803 (±.012) | .803 (±.012) | .005 | – | – |
| AG News | RoBERTa | — | — | – | .280 (±.022) | .111 (±.024) | – | .906 | .884 |
| | T5 | — | — | – | .010 (±.003) | .010 (±.003) | .990 | **.907** | **.886** |
| | FlanT5 | **.819** | **.789** | .000 | **.813 (±.008)** | **.782(±.009)** | .000 | – | – |
| | LLaMA | .479 | .463 | .011 | .787 (±.006) | .753 (±.005) | .003 | – | – |

Table 5: Micro- and Macro-F1 results for 'AG News' and 'IMDB' datasets for RoBERTa-large, fine-tuned T5 model, Flan-T5, LLaMA 2. The ratio of wrongly-formatted outputs is included in the wrong labels (labs) column. The results for Flan-T5 and LLaMA 2 are based on averaged results across all prompts.

and represent the Twitter domain. This suggests that the models find it more challenging to categorise such texts versus more categorical-based datasets such as news and articles which are part of the datasets with 6 to 9 labels. In contrast, the performance of both prompting approaches decreases as the number of labels for the classification task increases.

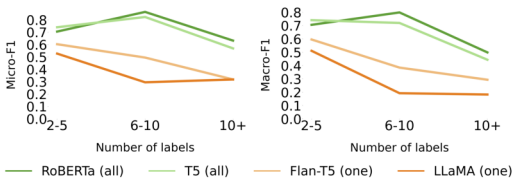


Figure 4: Averaged Micro-F1 and Macro-F1 results based on number of classification labels: 'RoBERTa (all)' and 'T5 (all)' refer to models fine-tuned on the entire training set, 'Flan-T5 (one)' and 'LLaMA (one)' refer to one-shot prompting.

**Datasets used for pre-training.** As mentioned earlier in the section, we analyse the performance of models for the 'IMDB reviews' and 'AG News' datasets separately as they are used in the fine-tuning of the Flan-T5 model. For these datasets (see Table 5) Flan-T5 performance significantly improves achieving micro- and macro-F1 results comparable to fine-tuning models on the entire dataset. For instance, for the IMDB dataset, the difference in macro-F1 between Flan-T5 and RoBERTa is 0.007 while for the AG news the difference in macro-F1 is 0.027. In contrast, the performance gap for the rest of the datasets between Flan-T5 and the best performing fine-tuning model is on average around 0.250 in micro-F1. This shows the significant impact that data contamination may have in the final results. However, a careful data contamination analysis becomes harder on large models for which training data is not available, and especially for closed models.

**GPT Analysis.** Table 4 presents zero-shot prompting results for the GPT 3.5-Turbo model for the following datasets: 'irony', 'offense', 'bbc', 'reuters', 'pcl', and 'safeguard'. We have used the class-based prompt for prompting with GPT 3.5 because it has shown to lead to the higher overall performance for Flan-T5 and LLaMA. Results show a clear advantage of the GPT-based model over Flan-T5 and LLaMA achieving on average 0.350 higher micro- and macro-F1 across the majority of the datasets, except for the 'PCL' dataset. Additionally, results achieved with zero-shot learning with GPT 3.5-Turbo outperform fine-tuned models on the entire dataset for the 'irony' dataset. However, for the rest of the datasets the model is still outperformed by fine-tuning approaches confirming the lack of generalisation abilities of few-shot learning techniques and text generation models for text classification.

## 5. Conclusions

This paper presents a large-scale study on how prompt-based LLMs in zero- and one- shot settings compare to smaller but fine-tuned language models for text classification. The evaluation spans across 16 datasets covering binary, multi-class, and multilabel problems. In particular, we compared three different types of models, i.e., linear models such as FastText, masked language models (RoBERTa), and text generation models tested in ICL settings (T5, Flan-T5, and LLaMA, as well as GPT 3.5-Turbo). Analyses on prompting techniques showed a clear advantage of the Flan-T5 model over LLaMA 1 and LLaMA 2 regardless of the prompt used for both zero- and one-shot settings. This shows that smaller but instruction-tuned models have better generalisation abilities for text classification than larger text generation models. Further, our analysis showed that results from zero- and few-shot learning LLMs are considerably lower in comparison to smaller models fine-tuned on the entire training set. This highlights the need for training data, even in the age of LLMs, and that fine-tuning smaller and more efficient language models can still outperform in-context learning methods of larger text generation models.

## 6. Acknowledgements

Aleksandra Edwards and Jose Camacho-Collados are supported by a UKRI Future Leaders Fellowship.
The safeguarding documents used for performing analysis in the paper have been collected in collaboration with the Wales Safeguarding Repository (WSR) project, funded by the National Independent Safeguarding Board (NISB), the Crime and Security Research Institute at Cardiff University (CSRI), and the School of Social Sciences at Cardiff University (SOCSI). We would like to thank the WSR team for their support.

## 7. Limitations

The main limitation of this research is the lack of experiments on fine-tuning Flan-T5 and LLaMA models as well as the lack of further analysis with larger text generation models such as LLaMA with 13 and 17 billion parameters. Moreover, the paper presents a study for zero- and one-shot prompting. As future work, we plan to extend analysis to understand how the number of training instances affect the performance of in-context learning approaches. Further, considering the sensitivity of in-context learning approaches to the given instructions, it would be beneficial to perform further analysis on a larger more diverse set of prompts. Finally, the paper presents results for a single high resource language (English). Experiments for other languages (especially low-resource) could show a different tendency.

## 8. Bibliographical References

Simon Baker, Ilona Silins, Yufan Guo, Imran Ali, Johan Högberg, Ulla Stenius, and Anna Korhonen. 2015. Automatic semantic classification of scientific literature according to the hallmarks of cancer. *Bioinformatics*, 32(3):432–440.

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, et al. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. *arXiv preprint arXiv:2302.04023*.

Francesco Barbieri, Jose Camacho-Collados, Francesco Ronzano, Luis Espinosa Anke, Miguel Ballesteros, Valerio Basile, Viviana Patti, and Horacio Saggion. 2018. Semeval 2018 task 2: Multilingual emoji prediction. In *Proceedings of the 12th international workshop on semantic evaluation*, pages 24–33.

Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. 2019. Semeval-2019 task 5: Multilingual detection of hate speech against immigrants and women in twitter. In *Proceedings of the 13th international workshop on semantic evaluation*, pages 54–63.

Sidney Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, et al. 2022. Gpt-neox-20b: An open-source autoregressive language model. In *Proceedings of BigScience Episode# 5–Workshop on Challenges & Perspectives in Creating Large Language Models*, pages 95–136.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Ilias Chalkidis, Emmanouil Fergadiotis, Prodromos Malakasiotis, and Ion Androutsopoulos. 2019. Large-scale multi-label text classification on eu legislation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6314–6322.

Ilias Chalkidis, Manos Fergadiotis, Sotiris Kotitsas, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. An empirical study on large-scale multi-label text classification including few and zero-shot labels. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7503–7515.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Mingkai Deng, Jianyu Wang, Cheng-Ping Hsieh, Yihan Wang, Han Guo, Tianmin Shu, Meng Song, Eric Xing, and Zhiting Hu. 2022. Rlprompt: Optimizing discrete text prompts with reinforcement learning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3369–3391.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. *Advances in neural information processing systems*, 32.

Aleksandra Edwards, Jose Camacho-Collados, Hélène De Ribaupierre, and Alun Preece. 2020. Go simple and pre-train on domain-specific corpora: On the role of training data for text classification. In *Proceedings of the 28th international conference on computational linguistics*, pages 5522–5529.

Aleksandra Edwards, Asahi Ushio, Jose Camacho-Collados, Helene Ribaupierre, and Alun Preece. 2022. Guiding generative language models for data augmentation in few-shot text classification. In *Proceedings of the Fourth Workshop on Data Science with Human-in-the-Loop (Language Advances)*, pages 51–63.

Avia Efrat and Omer Levy. 2020. The turking test: Can language models understand instructions? *arXiv preprint arXiv:2010.11982*.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3816–3830.

Yao Ge, Yuting Guo, Yuan-Chi Yang, Mohammed Ali Al-Garadi, and Abeed Sarker. 2022. Few-shot learning for medical text: A systematic review. *arXiv preprint arXiv:2204.14081*.

Ariel Gera, Alon Halfon, Eyal Shnarch, Yotam Perlitz, Liat Ein-Dor, and Noam Slonim. 2022. Zero-shot text classification with self-training. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1107–1119, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Praveen Kumar Badimala Giridhara, Chinmaya Mishra, Reddy Kumar Modam Venkataramana, Syed Saqib Bukhari, and Andreas Dengel. 2019. A study of various text augmentation techniques for relation classification in free text. *ICPRAM*, 3:5.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, 3(1):1–23.

Aakriti Gupta, Kapil Thadani, and Neil O'Hare. 2020. Effective few-shot classification with transfer learning. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1061–1066.

Hossein Hosseini, Sreeram Kannan, Baosen Zhang, and Radha Poovendran. 2017. Deceiving google's perspective api built for detecting toxic comments. arxiv 2017. *arXiv preprint arXiv:1702.08138*.

Thorsten Joachims. 1998. Text categorization with support vector machines: Learning with many relevant features. In *European conference on machine learning*, pages 137–142. Springer.

Armand Joulin, Édouard Grave, Piotr Bojanowski, and Tomáš Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431.

Yanis Labrak, Mickael Rouvier, and Richard Dufour. 2023. A zero-shot and few-shot study of instruction-finetuned large language models applied to clinical and biomedical tasks. *arXiv preprint arXiv:2307.12114*.

Ken Lang. 1995. Newsweeder: Learning to filter netnews. In *Proceedings of the 12th International Conference on Machine Learning*, pages 331–339, Tahoe City, California.

Teven Le Scao and Alexander M Rush. 2021. How many data points is a prompt worth? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2627–2636.

David D Lewis, Yiming Yang, Tony Russell-Rose, and Fan Li. 2004. Rcv1: A new benchmark collection for text categorization research. *Journal of machine learning research*, 5(Apr):361–397.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. Gpteval: Nlg evaluation using gpt-4 with better human alignment. *arXiv preprint arXiv:2303.16634*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin

Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Jinghui Lu, Maeve Henchion, Ivan Bacher, and Brian Mac Namee. 2021. A sentence-level hierarchical bert model for document classification with limited labelled data. In *Discovery Science: 24th International Conference, DS 2021, Halifax, NS, Canada, October 11–13, 2021, Proceedings 24*, pages 231–241. Springer.

Andrew L Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies-volume 1*, pages 142–150. Association for Computational Linguistics.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2022. Cross-task generalization via natural language crowdsourcing instructions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3470–3487.

Marius Mosbach, Tiago Pimentel, Shauli Ravfogel, Dietrich Klakow, and Yanai Elazar. 2023. Few-shot fine-tuning vs. in-context learning: A fair comparison and evaluation. *arXiv preprint arXiv:2305.16938*.

Preslav Nakov, Alan Ritter, Sara Rosenthal, Fabrizio Sebastiani, and Veselin Stoyanov. 2019. Semeval-2016 task 4: Sentiment analysis in twitter. *arXiv preprint arXiv:1912.01973*.

David Oniani, Jordan Hilsman, Hang Dong, Fengyi Gao, Shiven Verma, and Yanshan Wang. 2023. Large language models vote: Prompting for rare disease identification. *arXiv preprint arXiv:2308.12890*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Bo Peng, Emmanuele Chersoni, Yu-Yin Hsu, and Chu-Ren Huang. 2021. Is domain adaptation worth your investment? comparing bert and finbert on financial tasks. In *Proceedings of the Third Workshop on Economics and Natural Language Processing*, pages 37–44.

Carla Perez Almendros, Espinosa Anke, Luis, and Steven Schockaert. 2020. Don't patronize me! an annotated dataset with patronizing and condescending language towards vulnerable communities. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5891–5902, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Mohammad Taher Pilehvar, Jose Camacho-Collados, Roberto Navigli, and Nigel Collier. 2017. Towards a seamless integration of word senses into downstream nlp applications. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1857–1869.

Flor Miriam Plaza-del Arco, Debora Nozza, and Dirk Hovy. 2023. Leveraging label variation in large language models for zero-shot text classification. *arXiv preprint arXiv:2307.12973*.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. Improving language understanding by generative pre-training. *OpenAI*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. 2022. Multitask prompted training enables zero-shot task generalization. In *ICLR 2022-Tenth International Conference on Learning Representations*.

Rajdeep Sarkar, Atul Kr. Ojha, Jay Megaro, John Mariano, Vall Herard, and John P. McCrae. 2021. Few-shot and zero-shot approaches to legal text classification: A case study in the financial sector. In *Proceedings of the Natural Legal Language Processing Workshop 2021*, pages 102–106, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Timo Schick and Hinrich Schütze. 2021a. Exploiting cloze-questions for few-shot text classifica-

tion and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269.

Timo Schick and Hinrich Schütze. 2021b. It's not just size that matters: Small language models are also few-shot learners. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2339–2352.

Edgar Schönfeld, Sayna Ebrahimi, Samarth Sinha, Trevor Darrell, and Zeynep Akata. 2019. Generalized zero-and few-shot learning via aligned variational autoencoders. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8239–8247. IEEE.

Taylor Shin, Yasaman Razeghi, Robert L Logan IV, Eric Wallace, and Sameer Singh. 2020. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. *arXiv preprint arXiv:2010.15980*.

Congzheng Song, Shanghang Zhang, Najmeh Sadoughi, Pengtao Xie, and Eric Xing. 2021. Generalized zero-shot text classification for icd coding. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 4018–4024.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in nlp. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3645–3650.

Jiuding Sun, Chantal Shaib, and Byron C Wallace. 2023. Evaluating the zero-shot robustness of instruction-tuned language models. *arXiv preprint arXiv:2306.11270*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023a. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Rima Türker, Lei Zhang, Maria Koutraki, and Harald Sack. 2019. Knowledge-based short text categorization using entity and category embedding. In *The Semantic Web: 16th International Conference, ESWC 2019, Portorož, Slovenia, June 2–6, 2019, Proceedings 16*, pages 346–362. Springer.

Cynthia Van Hee, Els Lefever, and Véronique Hoste. 2018. Semeval-2018 task 3: Irony detection in english tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 39–50.

Vijay Viswanathan, Chenyang Zhao, Amanda Bertsch, Tongshuang Wu, and Graham Neubig. 2023. Prompt2model: Generating deployable models from natural language instructions. *arXiv preprint arXiv:2308.12261*.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*, 32.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations*.

Han Wang, Canwen Xu, and Julian McAuley. 2022a. Automatic multi-label prompting: Simple and interpretable few-shot classification. *arXiv preprint arXiv:2204.06305*.

Yaqing Wang, Quanming Yao, James T Kwok, and Lionel M Ni. 2020. Generalizing from a few examples: A survey on few-shot learning. *ACM computing surveys (csur)*, 53(3):1–34.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, et al. 2022b. Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5085–5109.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan

Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Yiming Yang. 1999. An evaluation of statistical approaches to text categorization. *Information retrieval*, 1(1-2):69–90.

Wenpeng Yin, Jamaal Hay, and Dan Roth. 2019. Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach. In *2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019*, pages 3914–3923. Association for Computational Linguistics.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. Semeval-2019 task 6: Identifying and categorizing offensive language in social media (offenseval). In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 75–86.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, pages 649–657.

Xinwei Zhang and Bin Wu. 2015. Short text classification based on feature extension using the n-gram model. In *2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, pages 710–716. IEEE.

Yiwen Zhang, Caixia Yuan, Xiaojie Wang, Ziwei Bai, and Yongbin Liu. 2022. Learn to adapt for generalized zero-shot text classification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 517–527, Dublin, Ireland. Association for Computational Linguistics.

Yifan Zhou. 2020. A review of text classification based on deep learning. In *Association for Computing Machinery*, ICGDA '20, page 132–136, New York, NY, USA.

# A. Appendix

In Section A.1 we present a comparison between prompting and fine-tuning techniques based on Macro-F1. In Section A.2, we present the prompts we used for performing analysis with zero- and one- shot in-context learning with Flan-T5 and LLaMA 1 and LLaMA 2.

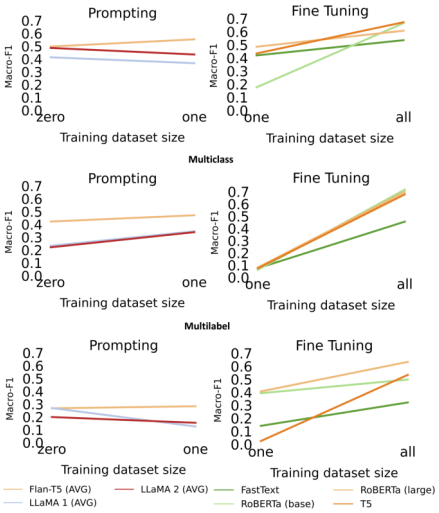


Figure 5: Comparison between prompting (left) and fine-tuning (right) approaches per text classification type where 'AVG' refers to averaged results across all prompt types per model. In 'Prompting', 'zero' and 'one' refer to zero- and one- shot prompt-based learning techniques, in 'Fine Tuning', 'one' refers to fine-tuning the models with one training instance per label and 'all' refers to fine-tuning using the entire dataset.

## A.1. Prompting versus Fine-tuning: Macro Results

Figure 5 shows the Macro-F1 results comparing prompting and fine-tuning techniques. Results show similar trends to those observed based on Micro-F1, presented in Section 4.2.

## A.2. Prompts

In Table 6 we have listed all 'domain' prompts we used per dataset. The 'task' and the 'generic' prompts are the same for all datasets and are presented in Table 2 in Section 3.4.

| Dataset | Domain Prompt |
|---|---|
| irony | Is the Tweet classified as irony or non-irony? |
| offense | Is the Tweet classified as offensive or non-offensive? |
| hate | Is the Tweet classified as hate or non-hate? |
| emoji | Which of the given emojis best describe the given Tweet?The emojis are: |
| sentiment | Is the Tweet positive, negative, or neutral? |
| BBC | Classify the news into one of the following topics: |
| Reuters | Classify the news into one of the following topics: |
| 20 News | Classify the newsgroup into one of the following topics: |
| Ohsumed | Select the medical conditions that this article is about. The options are: |
| Toxic | Which of the given toxic topics best describe the given comment? Choose one or more from the following topics: |
| Legal | Which of the given legal topics best describe the given legislation document?Choose one or more from the following topics: |
| Cancer | Which hallmarks of cancer are present in the text? Choose one or more from the following options |
| PCL | Which of the given topics best describe the patronising comment. Choose one or more from the following topics: |
| Safeguard | Which of the given themes best describe the sentence? Choose one or more from the following themes: |
| IMDB | Is the movie review positive or negative? |
| AG News | Select the topic that the given article is about.The topics are: |

Table 6: A list of all domain-based prompts used per dataset.