# Open-Set Text Classification with Limited Labeling Budget

**Anonymous Authors**[1]

## Abstract

Even with tremendous improvements in the performance of NLP models, the practical implementation of such models to different domains, languages or styles is expensive due to the cost associated with gathering and labelling task-specific data. Also, the practical systems need to consider open-set recognition scenarios where a sample from an unknown category may be encountered. We propose methodologies, sample sparsification and amplification, that solve these two problems of learning with small labelled data and open set recognition, respectively. We show the effectiveness of the proposed methods in text classification tasks with multiple open-source text classification datasets.

## 1. Introduction

The usage of deep learning methods has yielded significant improvements in natural language processing (NLP) tasks. These methods are data-hungry and they are nowadays trained on internet-scale databases to achieve good performance on some of the NLP tasks. However, achieving similar performance on other NLP domains, languages or styles becomes difficult due to relevant data unavailability of similar size and quality. The collection and labelling of such data is difficult because of various reasons such as associated cost, unavailability of expert annotators as well as privacy in some domains.

Transfer learning (Weiss et al., 2016), self-supervised learning (Goyal, 2022), few-shot/zero-shot learning (Song et al., 2022), meta-learning (Hospedales et al., 2022) are some of the approaches already proposed to solve the data scarcity problems. These methods reduce the number of samples required for training machine learning models to achieve similar performance. However, in practice, it is difficult to decide not just the size of the data but also which data samples should be chosen. With the rising digitization, a large number of data samples are relatively easily available, especially for NLP tasks. However, the methodology for finding a good quality subset of data, that should be labelled and further used for training machine learning models without affecting the performance, is missing.

Active learning (AL) is another well studied area in which iteratively data samples are evaluated and added to the training set (Ning et al., 2022; Prakhya et al., 2017; Liu & Huang, 2019; Ash et al., 2019; Gissin & Shalev-Shwartz, 2019). In reality, it is impractical to use because the AL strategy depends on warm-starting the model with information about the task. However, all of these approaches suffer with a cold start as initial samples are chosen randomly or methods with high uncertainty (Yuan et al., 2020). Also, calculating the valuation of each sample is very expensive as it involves training models with added samples in each iteration. Therefore, these methods are rarely used in practical settings.

Additionally, the classification methods work in closed-set settings, i. e., they consider only a fixed number of known labels. Such models can incorrectly classify test samples from unknown classes into one of the known classes with high confidence. For example, a classifier trained with a news classification dataset comprising of news with labels politics, science, and business will always wrongly classify sports news in one of those classes. This problem is addressed with Open Set Recognition (OSR) in which the incomplete knowledge of the world is accepted at the training time and the samples, that do not belong to any known category, are classified as unknown category.

In this paper, we propose two sampling techniques to solve the above mentioned problems. First sampling method finds a good quality subset of original samples, termed as *support set*, to accommodate the labelling budget. The second method finds a set of samples from the unknown category, which is termed as *amplified set*. We propose a training methodology with the combination of a support set and an amplified set to solve the problem of OSR and low labeling budget without affecting performance of the model.

Specifically, our contributions are as follow:

- a *sample sparsification* sampling technique to get support set, without the pre-knowledge of labeling set,

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

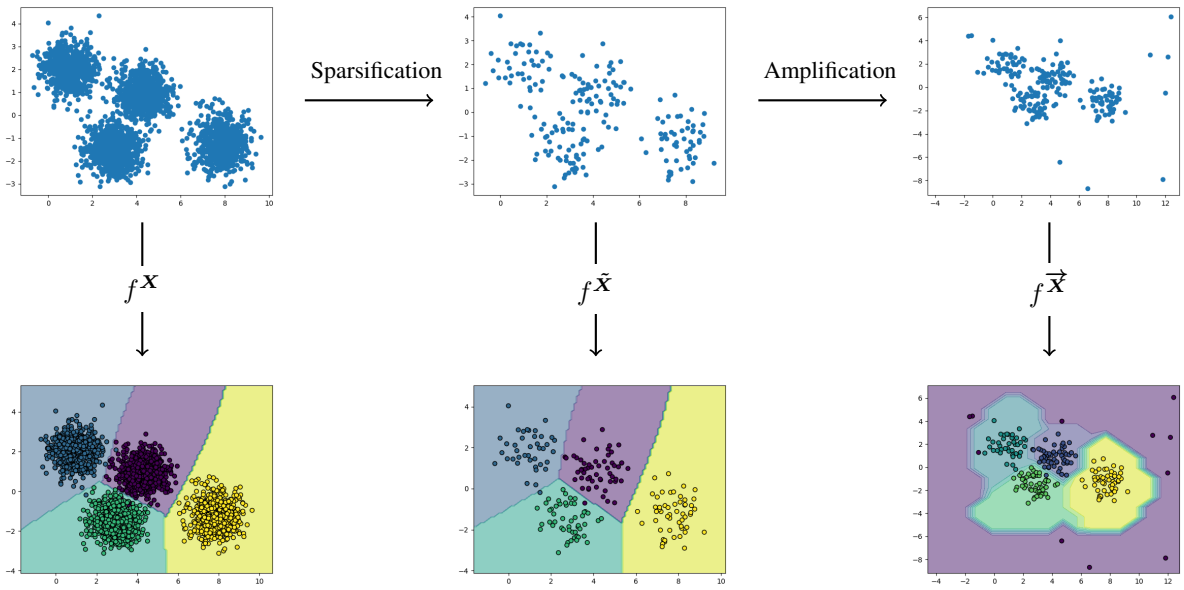


*Figure 1.* The overview of sample sparsification and sample amplification Methodology is described in this figure. The class boundaries are maintained after sample sparsification while with the sample amplification, the class boundaries are bounded to cover the sample spaces for each category. The rest of the sample space is considered an unknown category.

that results in drastic reduction in labeling budget with negligible reduction in accuracy,

- a *sample amplification* sampling technique to get amplified set for training a robust classifier that identifies samples from unknown categories,
- an application of both, sample sparsification and sample amplification procedures, for text classification task with SOTA results and
- the zero-shot multilingual text classification without necessity of task specific target language data.

**Outline:** Section 2 describes the important background work related to the proposed methodologies. The problem formulation is done in Section 3. In Section 4, the sample sparsification and amplification methodologies are presented while their application to the text classification task is illustrated in Section 5. The experimental results to prove the benefits of the proposed methodologies are summarised in Section 6. Finally, the related literature research is mentioned in Section 7 followed by conclusion and future scope in Section 8.

**Notation:** We define vectors with small bold letters, matrices/tensors by capital bold letters. A classifier $f^{(.)}$ indicates a classifier trained with dataset $(.)$.

## 2. Background

This section describes Sentence-BERT, SetFit and ALPS methodologies. The Sentence-BERT is a core part of the proposed methodology which is compared with SetFit and ALPS by performing experimental results.

### 2.1. Sentence-BERT

Sentence-BERT (SBERT)(Reimers & Gurevych, 2019) is a modification of the pre-trained BERT model that is obtained by training a Siamese network on STS-B dataset (Cer et al., 2017) to derive semantically meaningful sentence embeddings that can be compared using cosine-similarity or any other similar metric. It means that this model generates embeddings for similar sentences close to each other and vice versa.

### 2.2. SetFit

SetFit (Tunstall et al., 2022) is a recently proposed few-shot learning approach of training text-classification framework. It first fine-tunes a pre-trained English SBERT model on a small number of English text pairs, in a contrastive Siamese manner. The resulting model is then used to generate rich text embeddings, which are used to train a classification head.

### 2.3. ALPS

ALPS (Active Learning by Processing Surprisal) uses pre-training loss with the BERT model to find the samples that surprise the model most and claims that these subsets of samples are good quality samples that can be labeled and used for training to get good accuracy.

## 3. Problem Formulation

We consider having training samples $\boldsymbol{X} = \{\boldsymbol{x}_i\}_{i=1}^{N}$. A smaller dataset $\tilde{\boldsymbol{X}} \subset \boldsymbol{X}$ is the support set. The samples from unknown categories, i. e., the amplified set is indicated as $\bar{\boldsymbol{X}}$. The union of $\tilde{\boldsymbol{X}}$ and $\bar{\boldsymbol{X}}$ is represented as $\overrightarrow{\boldsymbol{X}}$. The functions $f^{\boldsymbol{X}}$, $f^{\tilde{\boldsymbol{X}}}$ and $f^{\overrightarrow{\boldsymbol{X}}}$ represent models that have same architecture but trained with $\boldsymbol{X}$, $\tilde{\boldsymbol{X}}$ and $\overrightarrow{\boldsymbol{X}}$, respectively. The goal is to find $\tilde{\boldsymbol{X}}$ of the desired size without a pre-defined label set and $\bar{\boldsymbol{X}}$ such that

- the classifier $f^{\boldsymbol{X}}$ and $f^{\tilde{\boldsymbol{X}}}$ achieves similar accuracy results with the test dataset
- the classifier $f^{\boldsymbol{X}}$ and $f^{\tilde{\boldsymbol{X}}}$ achieves similar accuracy results with test dataset in target languages other than English without explicitly training on these other target languages,
- the classifier $f^{\overrightarrow{\boldsymbol{X}}}$ achieve similar accuracy to that of $f^{\boldsymbol{X}}$ (and also $f^{\tilde{\boldsymbol{X}}}$ indirectly) while robustly classifying samples from unknown classes as out-of-distribution (OOD) samples.

## 4. Proposed Methodology

The overall methodology is described in Fig. 1. The operations, *sample sparsification* and *sample amplification*, in the first row of the figure are the dataset transformation methods. The sample sparsification attempts to find a good support set $\tilde{\boldsymbol{X}}$ much smaller than the original data which still provides similar accuracy. On the other hand, the sample amplification process adds random samples from unknown categories so that the classifier can learn to detect unknown category samples. The second row shows the classification boundaries of $f^{\boldsymbol{X}}$, $f^{\tilde{\boldsymbol{X}}}$ and $f^{\overrightarrow{\boldsymbol{X}}}$. It can be seen from the figure that the classification boundary does not change much after applying the sample sparsification procedure to the input data. On the other hand, the decision boundary of classifier $f^{\overrightarrow{\boldsymbol{X}}}$ bounds the sample space for each class while the rest of the space is considered as an unknown category. The details of these procedures are described in the following subsections.

**Algorithm 1** Sample Sparsification

$\{\boldsymbol{C}_1, \boldsymbol{C}_2, \ldots, \boldsymbol{C}_p\} \leftarrow$**HDBSCAN**$(\boldsymbol{X})$
**for** $i = 1$ **to** $p$ **do**
  $\{\boldsymbol{C}_i^1, \boldsymbol{C}_i^2, \ldots, \boldsymbol{C}_i^m\} \leftarrow$ **Binning** $(\boldsymbol{C}_i)$
  **for** $j = 1$ **to** $m$ **do**
    $\tilde{\boldsymbol{X}}_{\boldsymbol{C}_i}^j \leftarrow$ **RandomSampling**$(\boldsymbol{C}_i^j, \phi_f^j)$
  **end for**
  $\tilde{\boldsymbol{X}}_{\boldsymbol{C}_i} = \{\tilde{\boldsymbol{X}}_{\boldsymbol{C}_i}^1, \tilde{\boldsymbol{X}}_{\boldsymbol{C}_i}^2, \ldots, \tilde{\boldsymbol{X}}_{\boldsymbol{C}_i}^m\}$
**end for**
$\tilde{\boldsymbol{X}} = \{\tilde{\boldsymbol{X}}_{\boldsymbol{C}_1}, \tilde{\boldsymbol{X}}_{\boldsymbol{C}_2}, \ldots, \tilde{\boldsymbol{X}}_{\boldsymbol{C}_p}\}$

### 4.1. Sample Sparsification

The training of machine learning models with a small set of samples is proposed in a framework of few-shot learning, where a fixed number of samples for each pre-defined class are sampled to form the support set. Thus, it requires the knowledge of the label set beforehand. An equal number of samples for each class are included in the support set considering that each class boundary has the same complexity. This will not guarantee that the support set will cover the data distribution/sample space of full data and the good support set can only be found iteratively by sampling different numbers of data samples.

The aim and intuition behind the sample sparsification is to cover the data distribution/sample space of full data as much as possible by sparsely sampling dense regions in the sample space. If the support set covers the data distribution of original data with enough density, as shown in Fig. 1, the accuracy of the model trained with the support set does not degrade with comparison to the model trained with original data. This is valid even if the label set is defined after finding the support set, i. e., the sample sparsification is label-agnostic.

The pseudo-code for the sample sparsification is described in Algorithm 1. In the first step, $p$ clusters of data $\boldsymbol{X}$ are formed using the HDBSCAN clustering method (McInnes et al., 2017). The cluster $\boldsymbol{C}_1$ indicates a set of samples inside the first cluster. The advantage of using HDBSCAN to k-means clustering is that the latter still needs to define how many clusters need to be formed. The number of clusters $p$ formed using the HDBSCAN method are not pre-defined but they depend on the structure/density of the data.

**Binning** We perform further semi-random sampling from the $p$ clusters to ensure that the original data distribution/sample space is covered. We further divide each cluster $i$ into pre-defined $m$ bins $\boldsymbol{C}_i^1, \boldsymbol{C}_i^2, \ldots, \boldsymbol{C}_i^m$ with each bin representing the set of samples at a particular range from the centroid of the cluster. For example, the samples near the cluster centroid are in the first bin while the samples furthest from the cluster centroid are in the final bin. We defined

the sparsity factor set $\phi_f$, which defines the sampling ratio for each bin, e.g. a sparsity factor of 0.1 for the bin will sample randomly 10% of samples uniformly from that bin. We heuristically increased the sparsity factor from the first bin (samples closest to the centroid) to the final bin (samples furthest from the centroid). This is because the denseness of samples decreases as the sample space needs to be covered increases as we move away from the centroid of the cluster. After performing this sparsification procedure for each bin of each cluster, the union of sparsified samples in all bins of all clusters forms the support set $\tilde{\boldsymbol{X}}$.

### 4.2. Sample Amplification

Even after finding a good support set and deciding on a good labeling schema, we cannot ensure that the real data distribution is covered as it is generally unknown and therefore many out-of-distribution (OOD) samples (that include samples from unknown categories) will be classified wrongly. Therefore, we propose to amplify the support set by adding OOD samples to aid the machine learning classifier in finding a decision boundary that can distinguish the OOD samples.

The basic intuition of finding OOD samples is finding large enough regions in sample space not covered by the original data followed by taking random samples from such spaces. The procedure to find these OOD samples is described in pseudo-code in Algorithm 2. The sample space is divided into Voronoi regions (Wikipedia, 2023) of original data. The Voronoi region $\boldsymbol{V}_i$ for a sample $i$ consists of all points in the sample space closer to that sample than any other sample. We sample $n_s$ samples $\bar{\boldsymbol{X}}_{\boldsymbol{i}}$ from Voronoi regions $\boldsymbol{V}_i$ if the area of that Voronoi region is greater than a pre-defined threshold $a_t$. The collection of these samples from all valid Voronoi regions forms the OOD samples $\bar{\boldsymbol{X}}$ which is used as an amplified set. The visualization of Voronoi regions for original data from Fig. 1 are shown in Fig. 2.

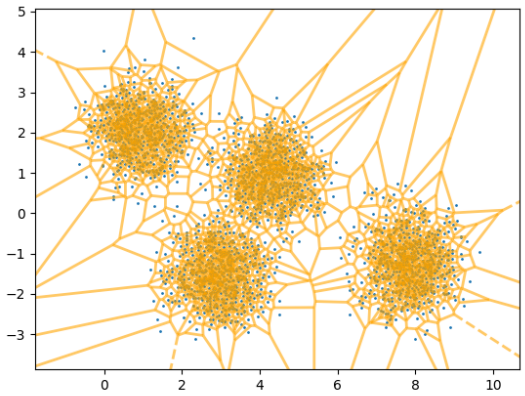


*Figure 2.* Visualization of Voronoi Regions of Raw Data

**Algorithm 2** Sample Amplification

**Input:** data $\boldsymbol{X}$
$\{\boldsymbol{V}_0, \boldsymbol{V}_1, \ldots, \boldsymbol{V}_{k-1}\} \leftarrow$ **Voronoi** $(\boldsymbol{X})$
**for** $i = 1$ **to** $k$ **do**
  **if** $Area(\boldsymbol{V}_i) > a_t$ **then**
    $\bar{\boldsymbol{X}}_{\boldsymbol{V}_i} \leftarrow$ **RandomSampling**$(\boldsymbol{V}_i, n_s)$
  **else**
    $\bar{\boldsymbol{X}}_{\boldsymbol{V}_i} = \emptyset$
  **end if**
**end for**
$\bar{\boldsymbol{X}} = \{\bar{\boldsymbol{X}}_{\boldsymbol{V}_1}, \bar{\boldsymbol{X}}_{\boldsymbol{V}_2}, \ldots, \bar{\boldsymbol{X}}_{\boldsymbol{V}_k}\}$

## 5. Application to Text Classification

This section describes the application of methods described in Section 4 to the text classification task. To apply the sample sparsification and amplification methods to the text, we need to transform the text into a representation such that the text belonging to the same class or semantically similar to each other lies close to each other and vice versa. We propose a pre-processing step to convert the text into sentence embedding with the help of the SBERT model $g$ described in Sec. 2.1.

As the similar text lies close to each other in the SBERT model's output embedding space, it is perfectly suitable for the application of the sample sparsification and amplification procedure. These sentence-embeddings of input texts calculated using the SBERT model are considered as the sample set $\boldsymbol{X}$ with which the support set $\tilde{\boldsymbol{X}}$ and amplified set $\bar{\boldsymbol{X}}$ can be obtained as described in section 4.1 and 4.2, respectively.

The procedure for text classification with SetFit is different in few aspects. It samples a fixed number of samples for each class randomly like in any other few-shot learning method and creates the sample set $\tilde{\boldsymbol{s}}$ from a dataset of texts $\boldsymbol{s}$. It is to be noted here that the method requires both datasets $\boldsymbol{s}$ and its corresponding labels $\boldsymbol{y}$. The pre-trained SBERT model is fine-tuned on the sample set $\tilde{\boldsymbol{s}}$ in a contrastive Siamese manner, where positive and negative pairs are created by in-class and out-class selection. The SBERT model then trains on these pairs. The fine-tuned SBERT model is used to find the embeddings for the support set $\tilde{\boldsymbol{X}}$ after which they are fed to classification head.

We also propose a combination of SetFit with sample sparsification. The only difference is that the sample set $\tilde{\boldsymbol{s}}$ is found by the sample sparsification method and the rest of the procedure is the same as in the SetFit method. The key difference here is that the support set may include varying

*Table 1.* Classification accuracies with Sample Sparsification and Sample Amplification on Diverse 2D Synthetic Datasets with Different Classifiers. A cell in the table shows the classifier(row) accuracies on the dataset (column) with original data, after sample sparsification and then after sample amplification, respectively.

| CLASSIFIER/DATASET | MOONS (267 SAMPLES) | CONCENTRIC CIRCLES (253 SAMPLES) | LINEARLY SEPARABLE (217 SAMPLES) | RANDOM BLOBS (215 SAMPLES) |
|---|---|---|---|---|
| NEAREST NEIGHBOUR | 0.97, 0.97, 0.94 | 0.63, 0.64, 0.62 | 0.86, 0.84, 0.80 | 1.00, 1.00, 0.98 |
| RBF SVM | 0.98, 0.95, 0.95 | 0.69, 0.67, 0.61 | 0.87, 0.81, 0.82 | 1.00, 1.00, 0.97 |
| DECISION TREE | 0.96, 0.93, 0.85 | 0.66, 0.64, 0.56 | 0.86, 0.86, 0.82 | 0.99, 0.99, 0.95 |
| RANDOM FOREST | 0.96, 0.94, 0.87 | 0.67, 0.66, 0.58 | 0.88, 0.87, 0.80 | 1.00, 0.99, 0.97 |
| NEURAL NET | 0.96, 0.90, 0.89 | 0.68, 0.59, 0.61 | 0.87, 0.87, 0.81 | 1.00, 1.00, 0.96 |
| NAIVE BAYES | 0.88, 0.88, 0.86 | 0.68, 0.60, 0.65 | 0.87, 0.86, 0.85 | 1.00, 1.00, 0.99 |
| QDA | 0.88, 0.88, 0.86 | 0.68, 0.58, 0.64 | 0.87, 0.87, 0.85 | 1.00, 1.00, 0.99 |

numbers of samples for different classes unlike in pure SetFit or few-shot learning cases where the same number of samples for all classes are included in the support set.

## 6. Results

This section describes the different experiments and their corresponding results to show the benefits of the proposed methods in this paper.

We first describe some common settings in all the experiments. We used the distiluse-base-multilingual-cased-v1 SBERT model for calculating the sentence embeddings of the text and bert-base-uncased with the ALPS method. For all the experiments, we started with 1% samples from the first bin of the cluster and the percentage of samples is increased by the *increment factor* for other bins in the cluster. For example, with an increment factor of 2, the sparsity factor for bins would be $1, 3, 5, \ldots$ from the first bin to the last bin of each cluster. The number of bins for all the clusters in all experiments is set to 10. The classifier ($f^{\boldsymbol{X}}, f^{\boldsymbol{X}'}, f^{\overline{\boldsymbol{X}}}$) for all text classification experiments is scikit-learn RBF SVM model with the regularisation parameter 100 and kernel coefficient of 2. The choice of these parameters is selected by performing the ablation study described in Appendix A.1 and A.2. For a fair comparison, we use the same number of random samples with the SetFit and ALPS methods as sample sparsification found in each experiment.

### 6.1. Results with 2D Data

We tested the effectiveness of sample sparsification and amplification methodology for different classifiers using different 2D synthetic datasets. We used moons, concentric circles, blobs and linearly separable datasets with 4000 samples generated from sklearn functions. The results of sample sparsification and amplification on these datasets for scikit-learn classifiers K-nearest neighbor, SVM with RBF kernel, decision tree, random forest, MLP, Gaussian naive Bayes and quadratic discriminant analysis are summarised in Table 1. The increment factor for these experiments was 2. The results show that the sample sparsification finds a support set that is roughly 10% of the original data and the classifier accuracy reduces negligibly. Also, the sample amplification does not affect the accuracy too much making it suitable also to classify OOD samples. This experiment shows that RBF SVM comparatively provides best results with these toy datasets, therefore we use this classifier for further experiments with real-world text datasets.

### 6.2. Results with Text Classification Dataset

We performed experiments with the datasets AGNews (Del Corso et al., 2005), PubMed-RCT (Dernoncourt & Lee, 2017), Emotions (Saravia et al., 2018), IMDB Review (Maas et al., 2011), Amazon Counterfactual (O'Neill et al., 2021) and cyberbullying classification (Wang et al., 2020) to understand whether the sample sparsification and amplification works with real-world NLP dataset and compare it with SetFit and ALPS methodology. The results of the experiments with different increment factors are shown in Table 2. The results prove the benefit of the sample sparsification method to achieve similar accuracy with drastically reduced labeled samples and it generates better results than ALPS. Although, SetFit achieves slightly better accuracy with the same number of samples, it needed pre-knowledge of label set. The combination of SetFit with sample sparsification, i. e., providing samples found with sample sparsification to SetFit methodology achieves better accuracy than pure SetFit methodology. The application of sample amplification also does not degrade the accuracy showing its applicability to real-world open-set recognition scenarios. It is to be noted that the sentence-embeddings of the support set were first reduced to two-dimensional vectors using TSNE method to simplify the Voronoi region formation. If the Voronoi regions are formed in the original dimension of the sentence embeddings, it is expected to reduce the accuracy loss indicated in the results with sample amplification.

*Table 2.* Classification accuracies with Sample Sparsification,Sample Amplification, SetFit and SetFit with Sample Sparsification for AGNews Dataset. The total number of original samples are 120000. SetFit results are with roughly same number samples in the support set corresponding to the number of samples in the support set found with sample sparsification.

| DATASET (# SAMPLES) | INCREMENT FACTOR (# SAMPLES) | SAMPLE SPARSIFY | SAMPLE AMPLIFY | SETFIT | ALPS | SETFIT + SAMPLE SPARSIFY |
|---|---|---|---|---|---|---|
| AGNEWS (200000) | 1 (6036) | $0.83 \pm 0.03$ | $0.81 \pm 0.04$ | $0.84 \pm 0.03$ | $0.79 \pm 0.02$ | $\mathbf{0.86 \pm 0.02}$ |
| | 5 (29952) | $0.85 \pm 0.03$ | $0.82 \pm 0.03$ | $0.85 \pm 0.02$ | $0.80 \pm 0.02$ | $\mathbf{0.88 \pm 0.01}$ |
| | 10 (60236) | $0.86 \pm 0.01$ | $0.83 \pm 0.02$ | $0.87 \pm 0.02$ | $0.82 \pm 0.02$ | $\mathbf{0.90 \pm 0.01}$ |
| PUBMED-RCT (200000) | 1 (68) | $0.81 \pm 0.03$ | $0.79 \pm 0.01$ | $0.82 \pm 0.02$ | $0.79 \pm 0.01$ | $\mathbf{0.83 \pm 0.01}$ |
| | 5 (28550) | $0.82 \pm 0.02$ | $0.80 \pm 0.01$ | $0.84 \pm 0.02$ | $0.80 \pm 0.01$ | $\mathbf{0.86 \pm 0.01}$ |
| | 10 (55722) | $0.84 \pm 0.02$ | $0.81 \pm 0.01$ | $0.85 \pm 0.01$ | $0.82 \pm 0.01$ | $\mathbf{0.89 \pm 0.01}$ |
| EMOTIONS (16000) | 1 (725) | $0.60 \pm 0.01$ | $0.59 \pm 0.01$ | $0.59 \pm 0.02$ | $0.54 \pm 0.02$ | $\mathbf{0.63 \pm 0.01}$ |
| | 5 (2930) | $0.63 \pm 0.02$ | $0.60 \pm 0.01$ | $0.61 \pm 0.02$ | $0.58 \pm 0.02$ | $\mathbf{0.66 \pm 0.01}$ |
| | 10 (5684) | $0.70 \pm 0.02$ | $0.65 \pm 0.01$ | $0.69 \pm 0.01$ | $0.65 \pm 0.02$ | $\mathbf{0.73 \pm 0.01}$ |
| IMDB REVIEW (50000) | 1 (1473) | $0.76 \pm 0.02$ | $0.73 \pm 0.03$ | $0.78 \pm 0.01$ | $0.71 \pm 0.02$ | $\mathbf{0.78 \pm 0.01}$ |
| | 5 (6143) | $0.78 \pm 0.01$ | $0.76 \pm 0.02$ | $0.79 \pm 0.01$ | $0.72 \pm 0.02$ | $\mathbf{0.81 \pm 0.01}$ |
| | 10 (11983) | $0.79 \pm 0.02$ | $0.77 \pm 0.01$ | $0.80 \pm 0.01$ | $0.74 \pm 0.02$ | $\mathbf{0.82 \pm 0.01}$ |
| AMAZON COUNTERFACTUAL (15218) | 1 (836) | $0.82 \pm 0.02$ | $0.79 \pm 0.02$ | $0.83 \pm 0.02$ | $0.78 \pm 0.02$ | $\mathbf{0.85 \pm 0.01}$ |
| | 5 (3495) | $0.84 \pm 0.02$ | $0.80 \pm 0.02$ | $0.84 \pm 0.02$ | $0.79 \pm 0.01$ | $\mathbf{0.86 \pm 0.01}$ |
| | 10 (6821) | $0.8448 \pm 0.02$ | $0.81 \pm 0.02$ | $0.85 \pm 0.01$ | $0.80 \pm 0.02$ | $\mathbf{0.88 \pm 0.01}$ |
| CYBERBULLYING CLASSIFICATION (28615) | 1 (1649) | $0.81 \pm 0.03$ | $0.79 \pm 0.02$ | $0.82 \pm 0.02$ | $0.78 \pm 0.02$ | $\mathbf{0.84 \pm 0.02}$ |
| | 5 (6711) | $0.83 \pm 0.02$ | $0.79 \pm 0.01$ | $0.84 \pm 0.02$ | $0.79 \pm 0.02$ | $\mathbf{0.86 \pm 0.01}$ |
| | 10 (13051) | $0.84 \pm 0.02$ | $0.82 \pm 0.01$ | $0.87 \pm 0.01$ | $0.82 \pm 0.02$ | $\mathbf{0.87 \pm 0.01}$ |

### 6.3. Testing Sample Amplification

There are no suitable datasets with out-of-domain datasets to the best of our knowledge. Therefore, we artificially created out-of-domain dataset by defining samples from one of the class in datasets AGNews, PUBMED-RCT and Cyberbullying classification as out-of-domain samples. The text classification model is trained with samples of rest of the classes and amplified samples found using the methodology described in section 2. The test data for evaluating the trained model contains samples from all classes including the out-of-domain class as well. The results in Table 5 show that the sample amplification procedure is effective for classsifying out-of-domain samples as the accuracy of the model is improved.

*Table 5.* Sample Amplification Results with One Pre-Defined Class Samples Held-Out

| DATASET | WITHOUT SAMPLE AMPLIFICATION | WITH SAMPLE AMPLIFICATION |
|---|---|---|
| AGNEWS | $0.73 \pm 0.03$ | $0.92 \pm 0.04$ |
| PUBMED-RCT | $0.81 \pm 0.03$ | $0.89 \pm 0.01$ |
| CYBERBULLYING CLASSIFICATION | $0.84 \pm 0.04$ | $0.93 \pm 0.02$ |

### 6.4. Zero-Shot Multilingual Transfer with MLSum

The results from Table 3 shows that pure sample sparsification achieves better zero-shot multilingual results than SetFit as the base multilingual transformer model was frozen unlike in SetFit where it was initially finetuned. The models were trained with AGNews dataset, which is an English news dataset, and the test inference results were generated using the samples from MLSUM dataset (Scialom et al., 2020) from categories that are semantically similar to that of in AGNews dataset.

### 6.5. Sample Sparsification Results with Different Labeling Set

We use DBPedia dataset (Auer et al., 2007) to show that the sample sparsification methodology does not require pre-knowledge of label set. This dataset provides taxonomic, hierarchical categories for 342,782 wikipedia articles. There are 3 levels, with 9, 70 and 219 classes respectively.

Even if the labelling set is changed, the same support set obtained by the sample sparsification achieves comparable accuracy to the accuracy achieved with original data. The results of these experiments with different increment factors

*Table 3.* Zero-Shot Multilingual Transfer with Sample Sparsification and SetFit On MLSUM dataset of Selected Topics. The Original Topic column is the topic names from MLSUM dataset while the Target AGNews Class column indicates the class name in AGNews dataset to which the original topic should be mapped.

| ORIGINAL TOPIC | TARGET AGNEWS CLASS | LANGUAGE | SETFIT | SAMPLE SPARSIFICATION |
|---|---|---|---|---|
| FOOTBALL | SPORTS | FRENCH | 0.88± 0.04 | 0.94 ± 0.005 |
| WIRTSHAFT/GELD | BUSINESS | GERMAN | 0.51 ± 0.06 | 0.77 ± 0.12 |
| DEPORTES BALENCESTO | SPANISH | SPORTS | 1± 0 | 1 ± 0 |
| ECONOMICS | BUSINESS | RUSSIAN | 0.46± 0.11 | 0.79 ± 0.09 |
| JUDO | SPORT | FRENCH | 0.75± 0 | 1 ± 0 |
| SPORT | SPORT | RUSSIAN, GERMAN | 0.8425 ± 0.06 | 0.91 ± 0.02 |
| SCIENCE | SCIENCE | RUSSIAN | 0.5688 ± 0.02 | 0.5921 ± 0.03 |

*Table 4.* Results with Same Support Obtained by Set Sample Sparsification for Labelling Set of Different Sizes

| INCREMENT FACTOR (# SAMPLES) | LABEL SET SIZE | | |
|---|---|---|---|
| | 9 | 70 | 219 |
| NO SPARSITY/ FULL DATA (240942) | 0.9867 ± 0.002 | 0.9636 ± 0.003 | 0.9327 ± 0.006 |
| 5 (61966) | 0.9836 ± 0.001 | 0.9441 ± 0.001 | 0.9094 ± 0.001 |
| 2 (2641) | 0.9429 ± 0.003 | 0.9184 ± 0.002 | 0.8846 ± 0.006 |

and different labelling set are shown in Table 4.

## 7. Literature Survey

This section presents different related works that are already proposed.

### 7.1. Meta Learning Approaches

The use of Siamese neural networks have been widely used with few shot learning setup (Müller et al., 2022; Koch, 2015). However, such methods require pre-defined label set.

In a meta learning category of learning to compare methods, the input is embedded in vector space and then distance/similarity/relation between two inputs in this space are used for inference. The matching networks (Vinyals et al., 2016) map a small labeled support set and an unlabelled example to its label, and obviate the need for fine-tuning to adapt to new class types. Prototypical networks (Snell et al., 2017) learns a metric space in which the model can perform well by computing distance between query and prototype representations of each class and classify the query to the nearest prototype's class. Sung et al. (2018) propose a two-branch relation networks, which learns to compare query against few-shot labelled sample support data. All of these method however focus only on computer vision tasks.

There are numerous learning to compare methods for text classification task (Yu et al., 2018; Geng et al., 2019a; Wu et al., 2019; Gao et al., 2019; Geng et al., 2020; Zhao et al., 2022) based on the above mentioned methods have been recently proposed. These methods use class prototypes for inferencing. However, the sample sparsification method proposes better class representation with multiple samples that preserves the class boundaries.

SentCluster (Bansal et al., 2021) samples from the clusters of samples similar to sample sparsification, but it does not have a systematic strategy for sampling from clusters. Similarly, the cluster IDs are used as pseudo labels in (Hsu et al., 2019) with fixed number of random samples from each cluster. Although, these methods do not require pre-defined label set, they still need to define the number of clusters to be formed with K-means clustering which is not required with the sample sparsification method as it uses HDBSCAN clustering.

The meta learning approach in (Murty et al., 2021) considers knowledge of labels and apply K-means clustering separately on each label group and construct tasks by choosing a cluster from each label group. An Induction Networks for few-shot text classification is presented in (Geng et al., 2019b) to deal with sample-wise diversity in the few-shot learning task, by explicitly modelling the ability to induce class-level representations from small support sets.

(Zhao et al., 2022) propose a memory imitation meta-learning (MemIML) method that enhances the model's reliance on support sets for task adaptation. Specifically, they introduce a task-specific memory module to store support set information and construct an imitation module to force query sets to imitate the behaviours of some representative support set samples stored in the memory.

### 7.2. Open Set Recognition in NLP

The survey of Open Set Recognition works are summarised in (Yang et al., 2021; Mahdavi & Carvalho, 2021). Statistical approaches such as threshold-based decision technique are being widely employed in text document openset classifications (Doan & Kalita, 2017; Fei & Liu, 2016) The approach cbsSVM (Fei & Liu, 2016) is popularly considered as the first open multiclass text classifier. This model is based on the CBS (Center-Based Similarity) space learning method, whereby a center for each class in the original problem is computed first. Then the data is transformed into a vector of their similarities to the class centroids to limit positive labelled area from an infinite space to a finite space. A decision threshold is then applied on posterior probabilities which are estimated from the SVM scores for each classifier.

(Dengxiong & Kong, 2023) introduce a side information learning algorithm for generalized open set recognition. It propose a hyperbolic side information learning framework to identify the unseen samples and an ancestor search algorithm to search the most similar ancestor from the taxonomy of selected known classes. An OOD resistant Prototypical Network to tackle the zero-shot OOD detection and few-shot ID classification task is proposed in (Tan et al., 2019).

Although, there are many approaches solving individual problem of learning with small data and open set recognition, to the best of our knowledge, this is the first work that proposes methodology to solve both problems in a single framework.

## 8. Conclusion and Future Scope

This work presents sample sparsification and sample amplification methodologies that propose to solve problems: i) finding a small high-quality dataset for labelling that can be used for training without affecting the performance, ii) open set recognition and iii) zero-shot multilingual transfer. The experimental results show that the methodology works better when used in combination with the recently proposed few-shot learning methodology named SetFit for the text classification task. Although, the focus in this work was for text classification, the proposed approaches can be applied to other tasks/modalities by using corresponding suitable embedding models.

There are multiple extensions for this work. We plan to use active learning as a next step to find samples from original samples which can be added to support set to reduce the accuracy loss further.

There are several limitations to the work that we describe in this section and hope that they provide researchers with problem statements to work on. Additionally, it is possible to extend this methodology to other modalities (e. g. images, videos, etc.) if suitable embedding models can be trained.

We used the HDBSCAN clustering method so that the proposed methods are not sensitive to skewed data distribution over unlabelled data but additional mitigation strategies can be proposed to address this issue even more concretely.

## Ethics Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

## References

Almeida, T. A., Hidalgo, J. M. G., and Yamakami, A. Contributions to the study of sms spam filtering: new collection and results. In *ACM Symposium on Document Engineering*, 2011. URL https://api.semanticscholar.org/CorpusID:13871930.

Ash, J. T., Zhang, C., Krishnamurthy, A., Langford, J., and Agarwal, A. Deep batch active learning by diverse, uncertain gradient lower bounds. *ArXiv*, abs/1906.03671, 2019. URL https://api.semanticscholar.org/CorpusID:182953134.

Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z. Dbpedia: A nucleus for a web of open data. In *Proceedings of the 6th International The Semantic Web and 2nd Asian Conference on Asian Semantic Web Conference*, ISWC'07/ASWC'07, pp. 722–735, Berlin, Heidelberg, 2007. Springer-Verlag. ISBN 3540762973.

Bansal, T., Gunasekaran, K. P., Wang, T., Munkhdalai, T., and McCallum, A. Diverse distributions of self-supervised tasks for meta-learning in nlp. In *Conference on Empirical Methods in Natural Language Processing*, 2021.

Cer, D. M., Diab, M. T., Agirre, E., Lopez-Gazpio, I., and Specia, L. Semeval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *International Workshop on Semantic Evaluation*, 2017.

Del Corso, G. M., Gullí, A., and Romani, F. Ranking a stream of news. In *Proceedings of the 14th International Conference on World Wide Web*, WWW '05, pp. 97–106, New York, NY, USA, 2005. Association for Computing Machinery. ISBN 1595930469. doi: 10.1145/1060745.1060764. URL https://doi.org/10.1145/1060745.1060764.

Dengxiong, X. and Kong, Y. Ancestor search: Generalized open set recognition via hyperbolic side information

learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 4003–4012, January 2023.

Dernoncourt, F. and Lee, J. Y. Pubmed 200k RCT: a dataset for sequential sentence classification in medical abstracts. *CoRR*, abs/1710.06071, 2017. URL http://arxiv.org/abs/1710.06071.

Doan, T. and Kalita, J. Overcoming the challenge for text classification in the open world. In *2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC)*, pp. 1–7, 2017. doi: 10.1109/CCWC.2017.7868366.

Fei, G. and Liu, B. Breaking the closed world assumption in text classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 506–514, San Diego, California, June 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1061. URL https://aclanthology.org/N16-1061.

Gao, T., Han, X., Zhu, H., Liu, Z., Li, P., Sun, M., and Zhou, J. FewRel 2.0: Towards more challenging few-shot relation classification. In *Proceedings of Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 6250–6255. Association for Computational Linguistics, 2019. doi: 10.18653/v1/D19-1649. URL https://aclanthology.org/D19-1649.

Geng, R., Li, B., Li, Y., Zhu, X., Jian, P., and Sun, J. Induction networks for few-shot text classification. In *Conference on Empirical Methods in Natural Language Processing*, 2019a.

Geng, R., Li, B., Li, Y., Zhu, X., Jian, P., and Sun, J. Induction networks for few-shot text classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3904–3913, Hong Kong, China, November 2019b. Association for Computational Linguistics. doi: 10.18653/v1/D19-1403. URL https://aclanthology.org/D19-1403.

Geng, R., Li, B., Li, Y., Sun, J., and Zhu, X. Dynamic memory induction networks for few-shot text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 1087–1094, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.102. URL https://aclanthology.org/2020.acl-main.102.

Gissin, D. and Shalev-Shwartz, S. Discriminative active learning. *ArXiv*, abs/1907.06347, 2019. URL https://api.semanticscholar.org/CorpusID:86571678.

Goyal, N. A survey on self supervised learning approaches for improving multimodal representation learning. *ArXiv*, abs/2210.11024, 2022.

Hospedales, T., Antoniou, A., Micaelli, P., and Storkey, A. Meta-learning in neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(09):5149–5169, sep 2022. ISSN 1939-3539. doi: 10.1109/TPAMI.2021.3079209.

Hsu, K., Levine, S., and Finn, C. Unsupervised learning via meta-learning. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=r1My6sR9tX.

Koch, G. R. Siamese neural networks for one-shot image recognition. 2015.

Liu, Z.-Y. and Huang, S.-J. Active sampling for open-set classification without initial annotation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33 (01):4416–4423, Jul. 2019. doi: 10.1609/aaai.v33i01.33014416. URL https://ojs.aaai.org/index.php/AAAI/article/view/4353.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., and Potts, C. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P11-1015.

Mahdavi, A. and Carvalho, M. A survey on open set recognition. In *2021 IEEE Fourth International Conference on Artificial Intelligence and Knowledge Engineering (AIKE)*, pp. 37–44, 2021. doi: 10.1109/AIKE52691.2021.00013.

McInnes, L., Healy, J., and Astels, S. hdbscan: Hierarchical density based clustering. *Journal of Open Source Software*, 2(11):205, 2017. doi: 10.21105/joss.00205. URL https://doi.org/10.21105/joss.00205.

Müller, T., Pérez-Torró, G., and Franco-Salvador, M. Few-shot learning with siamese networks and label tuning. In *Annual Meeting of the Association for Computational Linguistics*, 2022.

Murty, S., Hashimoto, T. B., and Manning, C. DReCa: A general task augmentation strategy for few-shot natural language inference. In *Proceedings of the*

*2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1113–1125, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.88. URL https://aclanthology.org/2021.naacl-main.88.

Ning, K.-P., Zhao, X., Li, Y., and Huang, S.-J. Active learning for open-set annotation. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 41–49, 2022. URL https://api.semanticscholar.org/CorpusID:246015682.

O'Neill, J., Rozenshtein, P., Kiryo, R., Kubota, M., and Bollegala, D. I wish I would have loved this one, but I didn't - A multilingual dataset for counterfactual detection in product reviews. *CoRR*, abs/2104.06893, 2021. URL https://arxiv.org/abs/2104.06893.

Prakhya, S., Venkataram, V., and Kalita, J. Open set text classification using CNNs. In Bandyopadhyay, S. (ed.), *Proceedings of the 14th International Conference on Natural Language Processing (ICON-2017)*, pp. 466–475, Kolkata, India, December 2017. NLP Association of India. URL https://aclanthology.org/W17-7557.

Reimers, N. and Gurevych, I. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. URL https://arxiv.org/abs/1908.10084.

Saravia, E., Liu, H.-C. T., Huang, Y.-H., Wu, J., and Chen, Y.-S. CARER: Contextualized affect representations for emotion recognition. In Riloff, E., Chiang, D., Hockenmaier, J., and Tsujii, J. (eds.), *Proceedings of Empirical Methods in Natural Language Processing*, pp. 3687–3697. Association for Computational Linguistics, 2018. doi: 10.18653/v1/D18-1404. URL https://aclanthology.org/D18-1404.

Scialom, T., Dray, P.-A., Lamprier, S., Piwowarski, B., and Staiano, J. MLSUM: The multilingual summarization corpus. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 8051–8067, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.647. URL https://aclanthology.org/2020.emnlp-main.647.

Snell, J., Swersky, K., and Zemel, R. S. Prototypical networks for few-shot learning. *ArXiv*, abs/1703.05175, 2017.

Song, Y., Wang, T.-Y., Mondal, S. K., and Sahoo, J. P. A comprehensive survey of few-shot learning: Evolution, applications, challenges, and opportunities. *ArXiv*, abs/2205.06743, 2022.

Tan, M., Yu, Y., Wang, H., Wang, D., Potdar, S., Chang, S., and Yu, M. Out-of-domain detection for low-resource text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3566–3572. Association for Computational Linguistics, 2019. doi: 10.18653/v1/D19-1364. URL https://aclanthology.org/D19-1364.

Tunstall, L., Reimers, N., Jo, U. E. S., Bates, L., Korat, D., Wasserblat, M., and Pereg, O. Efficient few-shot learning without prompts. *ArXiv*, abs/2209.11055, 2022.

Vinyals, O., Blundell, C., Lillicrap, T., kavukcuoglu, k., and Wierstra, D. Matching networks for one shot learning. In Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/file/90e1357833654983612fb05e3ec9148c-Paper.pdf.

Wang, J., Fu, K., and Lu, C.-T. Sosnet: A graph convolutional network approach to fine-grained cyberbullying detection. In *2020 IEEE International Conference on Big Data (Big Data)*, pp. 1699–1708, 2020. doi: 10.1109/BigData50022.2020.9378065.

Weiss, K. R., Khoshgoftaar, T. M., and Wang, D. A survey of transfer learning. *Journal of Big Data*, 3:1–40, 2016.

Wikipedia. Voronoi diagram — Wikipedia, the free encyclopedia. http://en.wikipedia.org/w/index.php?title=Voronoi%20diagram&oldid=1133579526, 2023. [Online; accessed 26-January-2023].

Wu, J., Xiong, W., and Wang, W. Y. Learning to learn and predict: A meta-learning approach for multi-label classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 4354–4364. Association for Computational Linguistics, 2019. doi: 10.18653/v1/D19-1444. URL https://aclanthology.org/D19-1444.

Yang, J., Zhou, K., Li, Y., and Liu, Z. Generalized out-of-distribution detection: A survey. *ArXiv*, abs/2110.11334, 2021.

Yu, M., Guo, X., Yi, J., Chang, S., Potdar, S., Cheng, Y., Tesauro, G., Wang, H., and Zhou, B. Diverse few-shot text classification with multiple metrics. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 1206–1215, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1109. URL https://aclanthology.org/N18-1109.

Yuan, M., Lin, H.-T., and Boyd-Graber, J. Cold-start active learning through self-supervised language modeling. In Webber, B., Cohn, T., He, Y., and Liu, Y. (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 7935–7948, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.637. URL https://aclanthology.org/2020.emnlp-main.637.

Zhao, Y., Tian, Z., Yao, H., Zheng, Y., Lee, D., Song, Y., Sun, J., and Zhang, N. Improving meta-learning for low-resource text classification and generation via memory imitation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pp. 583–595. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.acl-long.44. URL https://aclanthology.org/2022.acl-long.44.

# A. Appendix

We used SMS-Phishing dataset (Almeida et al., 2011) for ablation study of different hyperparameters selected for generating results in the paper.

## A.1. Ablation Study for Number of Bins

From Table 6, we can see as the number of bins increases, the accuracy also increases. However, as it also increases the computational cost and the increase from bins 10 to 15 is marginal, we chose 10 bins as hyperparameters in our experiments.

*Table 6.* Ablation study for sample sparsification hyperparameters for number of bins with SMS-Phishing dataset

| INCREMENT FACTOR | # BINS | | |
|---|---|---|---|
| | 5 | 10 | 15 |
| 1 | 0.8757 | 0.9098 | 0.9121 |
| 5 | 0.9091 | 0.9412 | 0.9485 |
| 10 | 0.9317 | 0.9502 | 0.9598 |

## A.2. Ablation Study for SVM hyperparameters

From Table 7, we can observe that the accuracy does not change drastically for different SVM kernels and regularization parameter value. Nonetheless, we chose RBF kernel with regularization parameter value of 100 in our experiments as it provides best accuracy.

*Table 7.* Ablation study for SVM hyperparameters (Kernel and Regularization parameter) with SMS-Phishing dataset

| KERNEL | REGULARIZATION PARAMETER | | |
|---|---|---|---|
| | 10 | 100 | 500 |
| LINEAR | 0.9506 | 0.9552 | 0.9531 |
| POLYNOMIAL | 0.9522 | 0.9480 | 0.9485 |
| RBF | 0.9501 | **0.9552** | 0.9506 |

# B. TSNE Graphs

This section includes a 2D visualisation of the sentence-embeddings obtained from the SBERT model for some datasets used in this paper with the TSNE method. The graphs show that the samples belonging to the same class are close to each other. The graph also shows there are multiple clusters for some of the classes. This shows the benefit of the sample sparsification method as it adds samples from these clusters in the support set which cannot be guaranteed in few-shot learning scenarios.

The TSNE plots for DBPedia dataset samples with different labeling sets also prove the benefit of sample sparsification methodology. The sample sparsification methodology will cover the sample space therefore any of the labeling set can be chosen even after finding the support set.

*Figure 3.* The TSNE plot of AGNews Embeddings

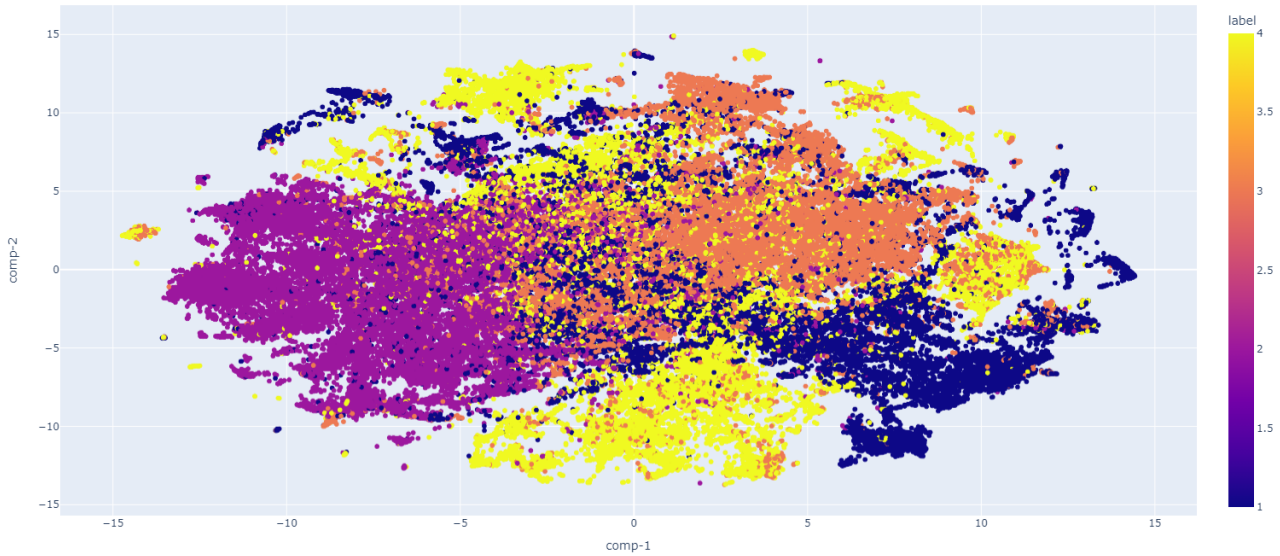


*Figure 4.* The TSNE plot of DBpedia Embeddings with 9 Classes

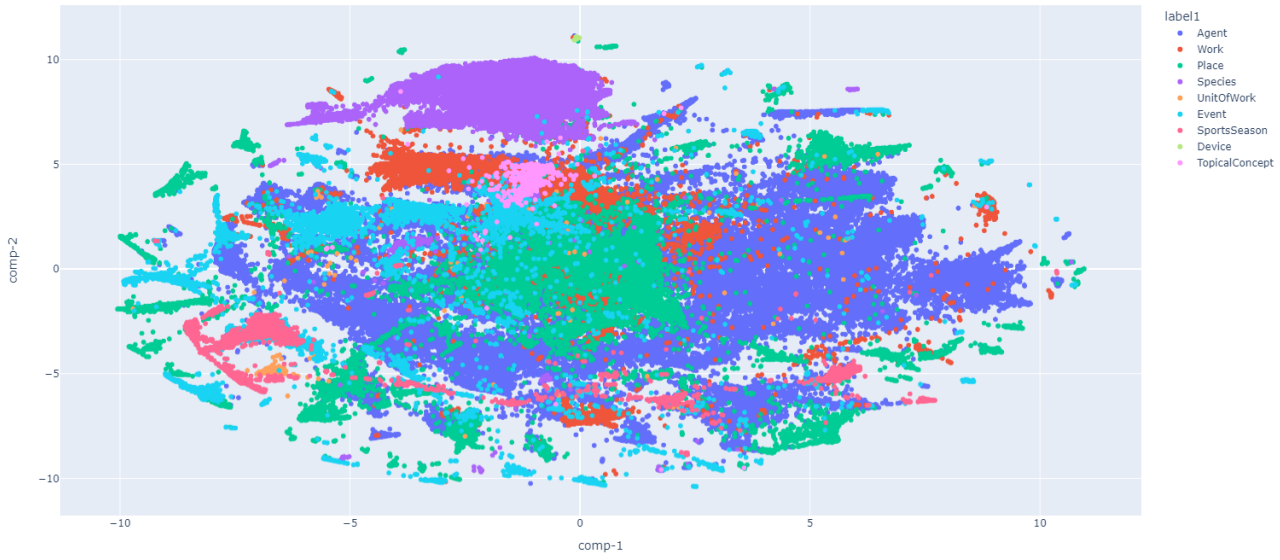

*Figure 5.* The TSNE plot of DBpedia Embeddings with 70 Classes

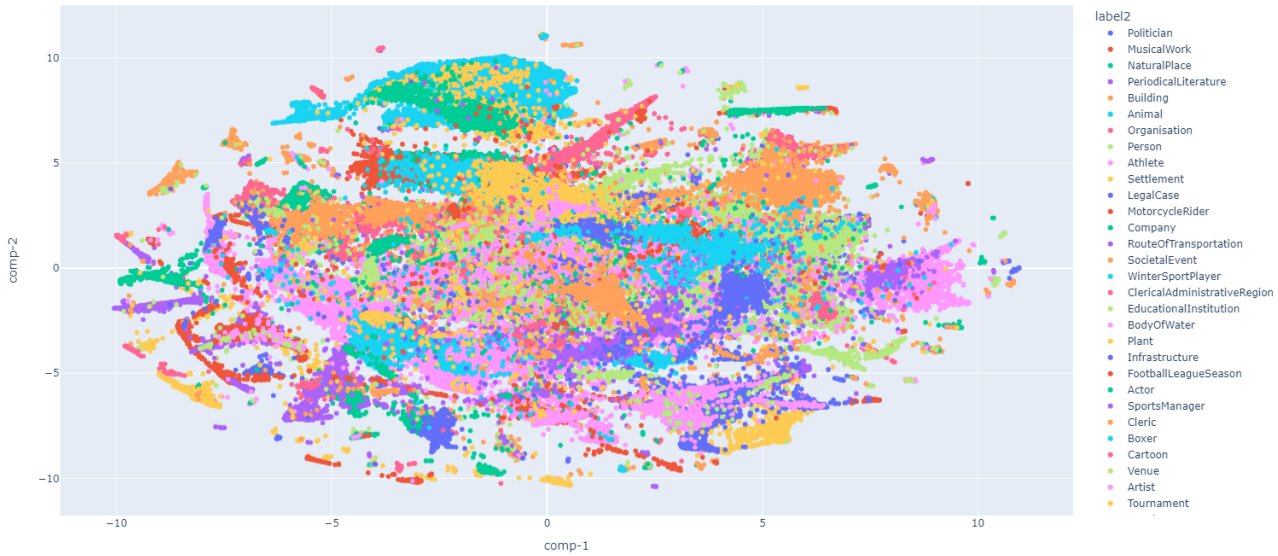


*Figure 6.* The TSNE plot of DBpedia Embeddings with 219 Classes