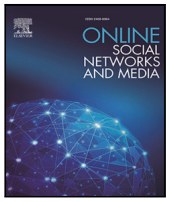
Online Social Networks and Media 33 (2023) 100244

Contents lists available at ScienceDirect

Online Social Networks and Media

journal homepage: www.elsevier.com/locate/osnem



Deep active learning for misinformation detection using geometric deep learning

Giorgio Barnabò [a,*], Federico Siciliano [a], Carlos Castillo [b], Stefano Leonardi [a], Preslav Nakov [c], Giovanni Da San Martino [d], Fabrizio Silvestri [a]

[a] Sapienza University of Rome, Department and Organization, Rome, Italy
[b] ICREA & Pompeu Fabra University,Department and Organization, Barcelone, Spain
[c] Qatar Computing Research Institute, Department and Organization, Doha, Qatar
[d] University of Padova, Department and Organization, Padova, Italy

ARTICLE INFO

Keywords:
Active learning
Fake news
Misinformation
Disinformation
Neural networks
Graph neural

ABSTRACT

Human fact-checkers currently represent a key component of any semi-automatic misinformation detection pipeline. While current state-of-the-art systems are mostly based on geometric deep-learning models, these architectures still need human-labeled data to be trained and updated — due to shifting topic distributions and adversarial attacks. Most research on automatic misinformation detection, however, neither considers time budget constraints on the number of pieces of news that can be manually fact-checked, nor tries to reduce the burden of fact-checking on – mostly pro bono – annotators and journalists. The first contribution of this work is a thorough analysis of active learning (AL) strategies applied to Graph Neural Networks (GNN) for misinformation detection. Then, based on this analysis, we propose Deep Error Sampling (DES) — a new deep active learning architecture that, when coupled with uncertainty sampling, performs equally or better than the most common AL strategies and the only existing active learning procedure specifically targeting fake news detection. Overall, our experimental results on two benchmark datasets show that all AL strategies outperform random sampling, allowing – on average – to achieve a 2% increase in AUC for the same percentage of third-party fact-checked news and to save up to 25% of labeling effort for a desired level of classification performance. As for DES, while it does not always clearly outperform other strategies, it still reduces variance in the performance between rounds, resulting in a more reliable method. To the best of our knowledge, we are the first to comprehensively study active learning in the context of misinformation detection and to show its potential to reduce the burden of third-party fact-checking without compromising classification performance.

1. Introduction

Since the 2016 United States presidential elections, both the general public and the scientific community have become increasingly aware of the threat posed to democracies by the spread of online misinformation. Research on misinformation detection has then experienced significant momentum, with many websites and independent journalists starting to fact-check online news, and releasing new datasets on which automatic detection systems can be trained. Almost at the same time, research on graph neural networks (GNNs) started reaching remarkable results in node and graph classification [1–5]. GNNs are made up of several layers of interconnected nodes, where each node represents a vertex in the graph and each edge represents a connection between two vertices. The nodes in the GNN are able to communicate with one another through these edges, allowing the GNN to process and analyze the graph as a whole, rather than just individual nodes. This makes GNNs well-suited for tasks that require understanding the relationships and dependencies between different elements in the graph.

GNNs have enabled scientists to better model news diffusion patterns in social networks, thus moving away from simple text-based fake news detection pipelines. In a nutshell, state-of-the-art GNN-based misinformation detection methods try to classify graphs that represent URL cascades in social networks. Despite the promising improvement in the performance of GNN-based architectures for fake news detection, in order to train these models, researchers still need high-quality third-party fact-checked news articles that are difficult and expensive to obtain. This problem is further amplified in large social networks and on the web, where the volume of news produced and spread daily makes extensive annotation virtually impossible. Indeed, manual data

* Corresponding author.
E-mail address: giorgio.barnabo@uniroma1.it (G. Barnabò).

https://doi.org/10.1016/j.osnem.2023.100244
Received 1 December 2022; Received in revised form 2 February 2023; Accepted 8 February 2023
Available online 24 February 2023

annotation consists of manually labeling and adding metadata to data, typically for the purpose of training machine learning models, and is a general pain point for most deep learning research due to its high costs — both in terms of human labour and time. In our case, while such scarcity of fake news data makes the problem of efficient annotation particularly urgent, research on misinformation detection under labeling constraints is still very scarce. In previous work, the need to reduce the human effort required to manually label news as fake or authentic has been largely ignored.

Active learning [6,7] is a machine learning approach in which a model is able to interactively query the user (or some other information source) to obtain the desired output, rather than being solely trained on a fixed dataset. In active learning, the model initially starts with a small amount of labeled data and makes predictions on the rest of the data. The model then selects a subset of the data for which it is least confident in its predictions, and asks the user to label this data. The labeled data is then used to update the model, and the process is repeated until the model reaches a satisfactory level of performance. In this work, we then present the first in-depth analysis of active learning (AL) strategies for fake news detection. We also propose Deep Error Sampling (DES) — a new deep-learning method that, when used in conjunction with uncertainty sampling, performs better, on average, than the most common AL strategies, including the only proposed active learning principle specifically targeting fake news detection. All tested active learning strategies were applied to three state-of-the-art GNN-based misinformation classifiers. As for the datasets, we performed experiments on PolitiFact [8] and FbMultiLingMisinfo [9], two high-quality and human-labeled collections of real and fake news. While the former is smaller and only contains news written in English, the latter is more recent, larger, and composed of URLs pointing to news in several languages. Overall, compared to random sampling, the best AL strategies allow to achieve a 2% increase in AUC for the same percentage of third-party fact-checked news and to save up to 25% of labeling effort for a desired level of classification performance.

To sum up, our original contributions are the following:

- Ann in-depth analysis of active learning (AL) strategies in the contest of automatic misinformation detection;
- We showed that, in the context of misinformation detection, active learning represents a viable and convenient strategy to increase the AUC classification metric by up to 5% and to reduce the cost of news labeling up to 25% for a given level of desired performance;
- Deep Error Sampling (DES), a new deep active learning architecture that, when coupled with uncertainty sampling, performs equally or better than the most common AL strategies and the only proposed active learning procedure specifically targeting fake news detection;
- In particularly, while other active learning strategies allow to reach results similar to DES, overall Deep Error Sampling shows lower variance between rounds and can be considered a more robust method.

To the best of our knowledge, no previous deep active learning method has leveraged prediction errors as the main discriminative signal. As shown in the experimental section, its characteristics seem to match well with both uncertainty and diversity sampling, paving the way for new combinations of more robust active learning strategies.

2. Related work

In this section, we first review the current state-of-the-art misinformation detection models that leverage geometric deep learning, we then go through the most common active learning strategies, with a focus on deep active learning, and finally, we briefly present recent finding on fake news benchmark datasets to justify our experimental choices.

2.1. Misinformation detection methods

Misinformation detection is not only challenging, but also necessary. As shown in a seminal work by Vosoughi et al. [10], in social networks fake news spreads faster and more extensively than high-quality information. Over the past five years, GNN-based methods have established themselves as the state-of-the-art approach in the fight against fake news. Unlike their predecessors, which were mostly content-based, these methods leverage the diffusion patterns of news in social networks as the main signal. These patterns are not merely features representing the spreading patterns of the news that are appended to content-based features to train traditional machine learning classifiers. Instead, the task is now formulated as a node [11–16] or a graph classification task [1,17,18], using methods such as Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs) [19]. State-of-the-art methods use either node or graph embeddings obtained by training a geometric deep learning architecture on an appropriate graph. The most commonly used architectures include Graph Convolution Networks (GCN) [1,15,20], Bi-Directional Graph Convolution Networks (BiGCN) [2], Graph Attention Networks [11,20–22], and GraphSAGE [17,20]. Depending on the approach, these representations can be further combined with text-based features and/or with non-GNN-based embeddings that capture other aspects of fake news [15].

Convenient APIs offered by Twitter, which can be used for research purposes, have turned this platform into the *de facto* standard for testing and validating misinformation detection methods [1,15,18]. Typically, in graph-based representations used for misinformation detection, nodes correspond to either news articles [11,13,14,16,23,24] or to users [1,12–17,23,25]. In other cases, content creators [11,13,14,23–25] or article authors or sources are included as additional nodes [13,14], and less often, nodes represent topics [11,24] or comments [12]. Regarding edges, news articles can be directly connected to their authors [11,13,14,23,24], topic(s) [11,24], or to users who post/share them [13]. Users, in turn, can be linked through their social graph, e.g., based on following or friendship relationships [1,13,14], reposting activity [16], replies [18], or (posted) content similarity [16]. Moreover, users can be connected to their posts [12,23], to an article through a stance score [13,14], or to nodes representing their posted comments [12], which in turn are usually connected to their corresponding post [12]. As for news-posting URL hostnames/domains, an edge can be added every time two hostnames/domains link to each other [13,14].

Finally, going more in depth into some of the most remarkable contributions, it is worth highlighting the following successful choices. Ren et al. [11] propose a novel hierarchical attention mechanism to perform node representation learning in heterogeneous information networks that effectively tackles fake news detection. They also use an active learning framework to enhance learning performance, especially when facing the paucity of labeled data. Yu et al. [12] aggregate multi-type information in a hierarchical manner and the information can reason over heterogeneous graph for the facticity of the news. Shu et al. [23] propose a tri-relationship embedding framework TriFN, which models publisher–news relations and user–news interactions simultaneously for fake news classification. The system is made up of 5 components, all based on some form of matrix decomposition and factorization. Finally, for each URL, Monti et al. [1] searched for all the related cascades and enriched their Twitter-based characterization (users and tweet data) by drawing edges among users according to Twitter's social network.

2.2. Active learning

Broadly speaking, AL refers to the iterative selection and labeling of samples to train a supervised classification model with the goal of reducing the number of labeled data points required to reach a desired performance. As extensively reviewed in Monarch [6] and Kumar

and Gupta [7], the earliest and still most common AL strategies are variations of uncertainty sampling and diversity sampling. Uncertainty sampling prioritizes the items that the current model is most uncertain about, at the risk of selecting multiple similar, redundant samples. Diversity sampling counteracts this problem by exploiting the fact that data points are usually clustered in feature space, and prioritizes centroids and out-layers. In practice, a combination of uncertainty and diversity sampling generally outperforms random sampling, and can be adapted to work in an online setting [26] and/or with highly unbalanced classes [27,28].

When complex deep learning architectures are deployed, however, standard AL strategies could under-perform due to the known problem of overconfidence of deep learning models. Indeed, the soft-max function is often used in the output layer of a neural network to convert the network's output into a probability distribution. It does this by exponentiating the output of each unit in the output layer, normalizing the resulting values, and then mapping the exponentiated outputs to a probability distribution. It follows that, when the network has learned to make very confident predictions (i.e., the output of a unit is much larger than the output of the other units), the soft-max function will map these outputs to a very high probability. This can happen, for instance, when training data is very unbalanced or when the model is used on out-of-domain samples. For this reason, new AL strategies are specifically designed to work in the deep learning context [29]. This branch of research is sometimes referred to as deep active learning. Recently, some works have also specifically targeted active learning in graphs and graph neural networks. Madhawa and Murata [30] have studied the application of active learning on attributed graphs. They show that algorithms designed for other data types do not perform well on graphs. In Liu et al. [31], after showing that state-of-the-art AL algorithms do not properly work on attributed graphs, a new latent space clustering-based active learning method for node classification (LSCALE) is proposed. Finally, in Madhawa and Murata [30], a novel framework to address the challenge of active learning in large-scale imbalanced graph data (node classification) is presented.

As for active learning in misinformation detection, the scientific literature still lags behind — with very few contributions. Ren et al. [11] use an active learning framework to enhance learning performance of their novel hierarchical attention mechanism. Bhattacharjee et al. [32], on the other end, propose a human–machine collaborative learning system to evaluate the veracity of a news content, with a limited amount of annotated data samples. In this work, we directly compare our Deep Error Sampling (DES) strategy against the active learning component of Ren et al. [11] — named here Deep Unseen Sampling (DUS). As for [32], we decided not to include this method in our analysis for two reasons: 1. the active learning component of the pipeline is very similar to Ren et al. [11], and 2. the whole workflow was optimized for a lexical-based fake news detector.

2.3. Fake news datasets

The robustness of misinformation detection research depends on the quality of the data used to conduct experiments, but we find that fake news benchmark datasets are often small and contain biases that affect the results (few thousand not-randomly-sampled fact-checked URLs). A relatively large dataset coming from the fact-checking website gossipcop.com, and a smaller one sampled from politifact.com – both released as part of FakeNewsNet [8] – constitute two of the most commonly used benchmark datasets [33,34]. While GossipCop still represents the largest fake news detection benchmark dataset, its real discriminative power has been recently put into question [9]. Indeed, GossipCop has proven to be exceptionally easy to classify and thus of limited utility to assess the discriminatory power of misinformation detection methods. For this reason, we decided not to include it in our experiments. Other common sources of annotated URLs or posts include BuzzFeed [35], Twitter [36] and Weibo [19,37].

These datasets for benchmarking fake news detection have reliable labels, but tend to include news in a single language, and to be created following unknown selection criteria — see, e.g., a recent in-depth review of these datasets [38]. Moreover, they are usually quite easy to classify. Larger datasets, such as NELA, can be created by sampling news from notoriously reliable and unreliable sources using distant supervision [39,40]. However, they are also noisy and biased since news articles are labeled as true or false according to their source, and are not individually fact-checked. Recently, a new multilingual benchmark dataset for misinformation detection was published [9]. This dataset comes from the recently published Facebook Privacy-Protected Full URLs Data Set [41], which comprises all 36 million URLs publicly shared on Facebook at least 100 times between January 2017 and July 2019, and includes fact-checking labels for 7334 of these URLs.

3. Problem statement

We consider a collection U of unlabeled news items (news articles/URLs) that we want to categorize as real news or fake with the highest possible accuracy. Since human labeling is both expensive and time-consuming, we assume that we are allowed to annotate only b news pieces. In other words, only the subset $B \subset U$, with $|B| = b$, will be sent to annotators. The quantity b represents a budget of possible annotations. We can also define b as a fraction of the size of U. Furthermore, we assume that each annotation has a unit cost. Using this labeled news, we train an automatic misinformation detection system, which we will leverage, in turn, to annotate the remaining unlabeled news $U \setminus B$. The budget b cannot be too low because it would not allow training a good classifier, but it cannot be too large because, in most practical cases, it would be unfeasible to send each news item for human review. Given the budget b, the question is: how can we efficiently and effectively select the B items to be fact-checked by professional journalists? This is precisely the question that active learning (AL) tries to answer in order to maximize the performance of the final model. The first step of any AL procedure is creating and annotating a validation set to guide the subsequent optimization steps. Following the literature [6], a fraction p_{test} of the initial dataset is selected uniformly at random to be used as the test set, and another percentage p_{val} of the remaining data is selected uniformly at random to form the validation set. Of course, the validation set must be human-labeled as well, and the $p_{val}(1 - p_{test})U$ samples will be subtracted from our labeling budget B. The AL strategy we use consists of a series of M iterations. At every iteration, new samples are identified, labeled, and added to the training set. Specifically, at each iteration, first we select k new URLs to annotate and add to the training set. Then, we train the classifier on all the URLs labeled so far. The validation set is used to assess the model performance and perform early stopping if its accuracy exceeds a pre-defined threshold. Iterations are executed until the annotation budget is exhausted. Most AL strategies require a somewhat reliable model to choose which samples to annotate — such a model is used by AL to find instances that bring more discriminative power to the current model. Since at the very beginning of the AL procedure, the training set is empty, and thus a classification model cannot be reliably obtained, for the first M_{rnd} iterations, we randomly select the k URLs instead of relying on the chosen AL technique.

4. Active learning strategies

In this section, we first present standard and well-established AL strategies that only use the input and output of a classifier to select the next batch of samples to annotate. Then we introduce two deep-learning-based AL methods where the AL strategy itself is a deep neural network. As explained in more detail just below, Deep Unseen Sampling (DUS) is based on a recently proposed active learning procedure for misinformation detection (Ren et al. [11]), while Deep Error Sampling (DES) represents a new active learning strategy that we personally designed to overcome some limitations of current neural approaches to active learning.

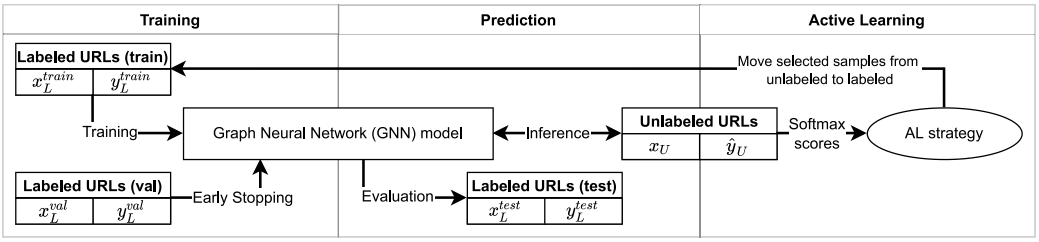


Fig. 1. Pipeline for "shallow" Active Learning strategies. First the GNN model is trained on the training set of labeled URLs (x_L^{train}, y_L^{train}), using the validation set of labeled URLs (x_L^{val}, y_L^{val}) to stop the training. The model is then used to predict the label ($\hat{y}_U$) for the unlabeled set of URLs (x_U). This set is finally passed to the Active Learning Strategy to select the set of samples to be removed from it and added to the training set. While in a real case scenario there would not be any test set, since in our experiments we have the labels for all URLs, at every iteration we use x_L^{test}, y_L^{test} to measure the quality of the AL strategies — in a sort of ex-post analysis.

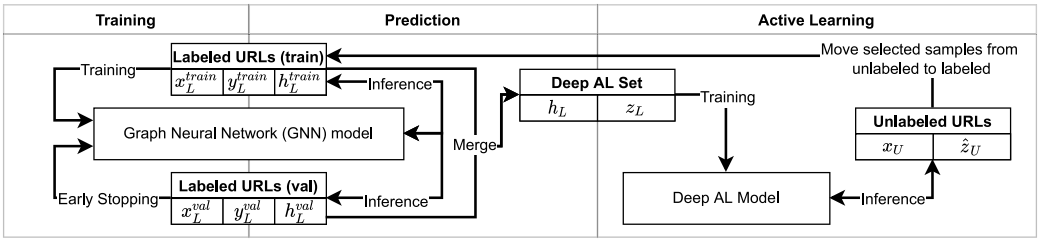


Fig. 2. Pipeline for Deep Active Learning strategies. First the GNN model is trained on the training set of labeled URLs (x_L^{train}, y_L^{train}), using the validation set of labeled URLs (x_L^{val}, y_L^{val}) to stop the training. The labeled test set (x_L^{test}, y_L^{test}) is then used to evaluate the model's performance (omitted for graphical reasons). From the trained model, embeddings are extracted for the training h_L^{train} and test h_L^{test} set. These embeddings are used, together with a label z_L to train the Deep AL model. For our new AL technique called Deep Error Sampling, $z_L = 1$ for misclassified samples, while $z_L = 0$ for correctly classified samples. At last, this model predicts the labels $\hat{z}_U$ for the unlabeled set of URLs (x_U), which are used to select the set of samples to be removed from it and added to the training set.

4.1. Classical active learning strategies

These Active Learning methods use the input and output of the classifier – or even the classifier itself – to decide which URLs to select. The overall structure of these types of methods is shown in Fig. 1.

4.1.1. Random sampling

Random sampling is the most intuitive baseline for the task at hand and represents the de-facto standard in the training of deep learning architectures. At each step, k samples are selected at random from the pool of unlabeled samples. Given that samples are independently picked, this method logically corresponds to selecting and labeling all the B URLs at once.

4.1.2. Uncertainty sampling

In uncertainty sampling, we use the most recently trained model to infer the labels of unlabeled samples. We assume that the last layer of a neural network-based classifier outputs soft-max scores for every class, and we use them to measure how confident the model is about its predictions. According to this principle, we will sample the k for which the model is most uncertain about and we will fact-check them in order to subsequently add them to the next-iteration training dataset. A known disadvantage of this methodology – when applied to deep learning models – is that usually deep learning architectures are overconfident of their predictions [29]. That is, they tend to predict soft-max scores very close to 0% or 100%.

4.1.3. Diversity sampling

Diversity sampling aims at avoiding the selection of very similar samples. The idea is that the model will not receive much help if it is trained with samples that are similar among each other. It is indeed much possible that – for a cluster of very similar samples – the model only needs a few of them to classify the whole cluster correctly. It is then important that the k samples represent different concepts, so that the model can generalize as much as possible. In practice, diversity sampling first clusters samples according to an algorithm like K-Means and then selects only a few examples from each cluster — for instance the centroid, a certain number of outliers and a certain number of random samples, such that the total is always equal to k. In our work, we used diversity sampling as an additional step for filtering the samples selected with the other AL strategies. After identifying $3k$ samples with one of active learning method, we applied K-Means on the sample features to form k clusters and then we selected the most uncertain URL according to the AL strategy metric. Each sample was represented through its activation scores of the second to last layer of the classification model.

4.2. Deep active learning strategies

Deep active learning refers to AL strategies that are specifically designed to work well with deep learning models. In this context, we will use deep active learning to group those pipelines where the AL strategy is itself a deep neural network. The Pipeline for this type of methods is shown in Fig. 2. In order to train a deep neural network able to identify worth-annotating URLs, we first need to define a suitable training set and a learning objective. Our idea is to use the second-to-last layer activation scores h_L of the fake news classifier for both the training and validation sets as input to this Deep Neural Network. Concerning labels z_L, we experimented with two different DeepAL models. Deep Unseen Sampling (DUS) mimics what was done in the only paper on active learning for misinformation detection (Ren et al. [11]). While the original contribution embeds active learning as an additional feature of a more complex adversarial model for learning node classes on heterogeneous graphs — we decided to test the core idea behind their AL procedure, that is to use internal activation scores of the misinformation classifier to predict whether a sample was already labeled and part of the training set. Deep Error Sampling (DES), instead, is our proposed DeepAL strategy, where we try to predict whether a sample will be correctly classified, thus getting around the problem of soft-max overconfidence. For both methods, the network used is a fully-connected deep neural network. The specific parameters can be found

in our anonymized Github repository.[1] Let us see the two techniques in more details.

4.2.1. Deep unseen sampling

Ren et al. [11] start from the assumption that it is good for the classifier to receive new samples other than those it has already seen. They therefore set the labels for the already labeled samples as 0, because the model has already seen them during training, and as 1 for the samples belonging to the validation set, because the model has not in fact seen them during its training. Since the training set of the classifier grows in time, at every iteration the number of samples taken from the validation set is equal to the current size of the training set of the misinformation classifier. Finally, each of the samples used to train the DeepAL architecture are represented through the second-to-last activation scores of the current misinformation detection model, i.e. that trained with the URLs labeled so far. At this point, using these labels as output and the embedding samples as input, we trained a feed-forward neural network to predict whether a URL has been already seen by the fake news classifier or not. In the end, the unlabeled data is given as input to the trained DeepAL architecture and the k samples with the higher prediction, i.e. those which the model predicts are more likely to be unseen by the classifier, are added to the training set.

4.2.2. Our method: Deep error sampling

This is the new method we propose in this paper. Always assuming that we want to train a Deep model that can select the best samples to send to fact-checking, and always constructing the network input from the samples' embeddings, we have chosen the labels differently this time. Our conjecture here is that we might try to predict in advance whether the classifier will mis-classify new samples. We pass the samples that we already have labeled, either training or validation, to the classifier and label 0 those that are classified correctly, and label 1 those that are classified incorrectly. On this set, we train our neural network, and then get the prediction on the unlabeled data. The k samples that the network thinks are most likely to have label 1, will be the ones where our fact-checking classifier is most likely to get it wrong, and it is our belief that they will be most useful for further training.

4.3. Mixed strategies

As in many other areas, often the best result is obtained by aggregating different methodologies. Also here, as the various AL techniques are capable of capturing different information about the samples, it may be useful to combine their outputs. Specifically, we used a simple rank aggregation technique to merge the top-k samples received as output from 2 AL techniques.

5. Fake news detection classifiers

We experimented with three state-of-the-art GNN-based approaches for misinformation detection that work on news diffusion graphs.

- GCN [3] A simple GCN that uses an efficient layer-wise propagation rule based on a first-order approximation of spectral convolutions on graphs. It can learn hidden layer representations that encode both local graph structure and features of nodes.
- GAT [4] The use of multi-head graph attention makes this model computationally highly efficient, thus allowing it to deal with neighborhoods of various sizes without depending on knowing the entire graph structure upfront.

Table 1
Statistics about FbMultiLingMisinfo and PolitiFact.

Dataset	FbMultiLingMisinfo	Politifact
Fake news	4,034	157
True news	3,300	157
Total news	7,334	314
Twitter posts	3,219,383	22,340
Twitter users	1,240,592	14,873

- GraphSAGE [5] This model exploits inductive node embedding by making use of node features in order to generalize to unseen nodes.

Implementation-wise, we re-implemented in PyTorch Lightning the code, written in PyTorch, distributed by Dou et al. [20].[2] Concerning the hyper-parameters, we used the values from the original papers as they performed well on both our datasets, as shown in Barnabò et al. [9]. The whole code of our project can be found on a GitHub repository.[3]

6. Datasets

We tested our pipeline on FbMultiLingMisinfo and Politifact, two publicly-available misinformation detection benchmarks. FbMultiLingMisinfo is a recently published multilingual collection of fact-checked news, extracted from the Facebook Privacy-Protected Full URLs Data Set [41], and including diffusion cascades on Twitter for each news article [9].

This dataset includes any URL publicly shared on Facebook at least 100 times between January 2017 and July 2019.

It is particularly relevant because, to the best of our knowledge, 1. it is the only multilingual dataset for misinformation detection; 2. it is the second-largest benchmark dataset for misinformation detection fact-checked at the level of individual news articles (URLs); 3. all included URLs are highly impactful (shared at least 100 times on Facebook); 4. it was shown to be more complex than PolitiFact and GossipCop, the two most used benchmark datasets for misinformation detection [9].

We also experimented with PolitiFact, a widely used benchmark for fake news detection collected from a fact-checking website that focuses on political reporting [8]. Statistics about both datasets are shown in Table 1.

The difference in the characteristics of the two datasets (one smaller and in only in English, the other multilingual) makes it possible to obtain information on the performance of the AL strategies proposed by us in two different scenarios.

6.1. Modeling the diffusion cascades of URLs shared on Twitter

The models we experimented with take as input a graph representing each URL diffusion cascades. As in Dou et al. [20], given the sequence of tweets and retweets mentioning a URL, we built a graph as follows: a central node represents the news and there is an additional node for each tweet. All direct tweets are connected to the central node, while re-tweets are connected to the tweet they are re-tweeting. Finally, similarly to Dou et al. [20], we obtained the node features by encoding the user description with the paraphrase-multilingual-mpnet-base-v2 model from the Hugging Face multilingual sentence embedding model trained as in Reimers and Gurevych [42].

For the central node representing the URL, we used the news title embedding. Our choice of a multilingual model is due to the multilingual nature of the FbMultiLingMisinfo dataset. For the PolitiFact dataset, we used the diffusion graphs shared in [20], but we replaced the given node features with the multilingual sentence embeddings.

[1] https://anonymous.4open.science/r/Active-Learning-for-Misinformation-Detection-10CChttps://anonymous.4open.science/r/Active-Learning-for-Misinformation-Detection-10CC

[2] http://github.com/safe-graph/GNN-FakeNews

[3] https://github.com/GiorgioBarnabo/Active-Learning-for-Misinformation-Detection

Table 2
Results on FbMultiLingMisinfo. For each AL strategy, we show the AUC at key iterations. Under the number of iterations – in round brackets – we placed the percentage of the dataset that has been selected and used as training. In addition to that, we must also factor in the 10% validation set that is part of the final fact-checking budget used. With an asterisk we have marked our novel method. DUS = Deep Unseen Sampling, DES = Deep Error Sampling. Results are averaged over 5 runs and reported with their standard deviations.

Results on the FbMultiLingMisinfo dataset					
AL strategy	Iterations				
metric: AUC	20 (3%)	40 (5,5%)	60 (8%)	80 (11%)	100 (13%)
GAT					
Random	0.71 ± 0.9	0.76 ± 0.10	0.82 ± 0.7	0.84 ± 0.04	0.85 ± 0.05
Uncertainty	0.73 ± 0.06	0.78 ± 0.08	0.82 ± 0.05	0.85 ± 0.06	**0.87 ± 0.06**
Uncertainty + Diversity	**0.74 ± 0.03**	**0.80 ± 0.04**	**0.84 ± 0.06**	0.85 ± 0.04	**0.87 ± 0.03**
DUS	0.72 ± 0.11	0.77 ± 0.09	0.81 ± 0.10	0.84 ± 0.08	0.85 ± 0.09
DUS + Diversity	0.71 ± 0.09	0.76 ± 0.09	0.80 ± 0.08	0.84 ± 0.09	0.85 ± 0.07
DES*	0.73 ± 0.05	0.78 ± 0.06	0.82 ± 0.06	0.85 ± 0.03	**0.87 ± 0.02**
DES* + Diversity	0.73 ± 0.04	**0.80 ± 0.05**	0.83 ± 0.04	0.85 ± 0.06	0.86 ± 0.04
DES* + Uncertainty	**0.74 ± 0.02**	**0.80 ± 0.02**	**0.84 ± 0.03**	**0.86 ± 0.04**	**0.87 ± 0.02**
GraphSAGE					
Random	0.74 ± 0.08	0.82 ± 0.07	0.85 ± 0.10	0.86 ± 0.09	0.87 ± 0.07
Uncertainty	0.75 ± 0.11	**0.84 ± 0.07**	0.86 ± 0.08	**0.88 ± 0.08**	**0.89 ± 0.09**
Uncertainty + Diversity	**0.78 ± 0.07**	0.83 ± 0.07	**0.87 ± 0.07**	**0.88 ± 0.05**	**0.89 ± 0.06**
DUS	0.75 ± 0.08	0.81 ± 0.09	0.84 ± 0.09	0.86 ± 0.07	0.86 ± 0.06
DUS + Diversity	0.76 ± 0.08	0.81 ± 0.05	0.85 ± 0.05	0.87 ± 0.04	0.87 ± 0.07
DES*	0.77 ± 0.05	**0.84 ± 0.07**	0.86 ± 0.04	**0.88 ± 0.03**	**0.89 ± 0.04**
DES* + Diversity	0.76 ± 0.05	**0.84 ± 0.05**	**0.87 ± 0.04**	**0.88 ± 0.05**	**0.89 ± 0.07**
DES* + Uncertainty	0.77 ± 0.06	**0.84 ± 0.05**	**0.87 ± 0.03**	**0.88 ± 0.04**	**0.89 ± 0.03**
GCN					
Random	0.74 ± 0.08	0.79 ± 0.06	0.82 ± 0.09	0.83 ± 0.10	0.85 ± 0.06
Uncertainty	0.76 ± 0.07	0.80 ± 0.10	0.83 ± 0.08	0.84 ± 0.09	0.85 ± 0.07
Uncertainty + Diversity	0.75 ± 0.09	**0.81 ± 0.11**	0.83 ± 0.09	0.84 ± 0.08	0.86 ± 0.09
DUS	**0.77 ± 0.06**	**0.81 ± 0.05**	0.82 ± 0.07	0.84 ± 0.08	0.85 ± 0.07
DUS + Diversity	0.76 ± 0.07	0.80 ± 0.06	0.82 ± 0.05	0.84 ± 0.07	0.85 ± 0.06
DES*	**0.77 ± 0.07**	**0.81 ± 0.06**	0.82 ± 0.05	0.85 ± 0.06	**0.87 ± 0.04**
DES* + Diversity	**0.77 ± 0.04**	**0.81 ± 0.05**	**0.84 ± 0.03**	**0.86 ± 0.02**	**0.87 ± 0.02**
DES* + Uncertainty	0.75 ± 0.05	0.80 ± 0.04	0.83 ± 0.04	0.85 ± 0.03	**0.87 ± 0.01**

7. Experiments & results

7.1. Experimental setting

We tested all the different AL strategies on GraphSAGE, GAT and GCN — three different state-of-the-art GNN-based misinformation classifiers [9]. We also tested all possible mixed strategies by combining two sampling strategies as explained in Section 4.3. The sampling strategies we show in the following results are only those that performed best. The experiment setting was as follows. For both Politifact and FbMultiLingMisinfo we set aside a random 10% of the URLs to use as validation sets. Validation sets are needed to perform early stopping and regularize the training throughout the active learning cycle. Since we assume the validation sets to be labeled as well, they must be subtracted to the total fact-checking budget. For FbMultiLingMisinfo, we set the number of AL iterations to 100, and select 10 URLs per iteration. For Politifact, given its reduced size, we opted for 20 iterations and 5 URLs per iteration. Regardless of the AL method, for FbMultiLingMisinfo the first $M_{rnd} = 5$ iterations always use random sampling, while for Politifact $M_{rnd} = 2$. All experiments were repeated 5 times and results were averaged. In addition, we applied a 3-step moving average on all the sequences of results to make the trends clearer.

7.2. Key findings

First and foremost, our analysis shows that active learning is a more efficient method for training GNN-based misinformation detection models. Indeed, as shown in Tables 4 and 5 — results from experiments on both FbMultiLingMisinfo and PolitiFact indicate that all tested active learning strategies, except for Deep Unseen Sampling, outperform random sampling, allowing to reach a certain level of classification performance (AUC) with much less labeled data. For FbMultiLingMisinfo specifically, Deep Error Sampling + Uncertainty Sampling yields the best results on GAT and GraphSAGE, while for GCN Deep Error Sampling + Diversity Sampling works better. In all three cases, for lower value of AUC, Uncertainty Sampling and Deep Error Sampling seem to outperform other methods. This is due to the fact that the active learning process is at the very beginning and the Deep Error Sampling architecture needs more data to be trained. Overall, the decrease in number of iterations required to reach a desired level of AUC is significant, with up to 50% less annotated URLs. As for PolitiFact, results reported in Table 3 suggest similar trends, but it is harder to draw definitive conclusions given the small size of this benchmark dataset and the larger overlap among different active learning procedures. For both benchmark datasets, however, experiments highlight how active learning strategies could make the process of training GNN-based misinformation detection methods not only faster, but also lighter for annotators. Findings on FbMultiLingMisinfo are further confirmed when looking at Figs. 3–5 — which show F1 Macro trends as the number of annotation rounds increases. For instance, the dotted line on Fig. 3 shows that Uncertainty Sampling + Diversity Sampling, Deep Error Sampling + Diversity Sampling, and Deep Error Sampling + Uncertainty Sampling reach an F1 Macro of 0.72 in just over 40 iterations, while the same result takes almost 100 iterations with Random Sampling or Deep Unseen Sampling + Diversity Sampling. On average, when we use GAT to detect fake news in FbMultiLingMisinfo, choosing Uncertainty Sampling + Diversity Sampling, Deep Error Sampling + Diversity Sampling, or Deep Error Sampling + Uncertainty Sampling reduces the number of iterations needed to reach a desired level of F1 Macro by 25 to 40. A similar pattern can be seen in Figs. 4 and 5, although the average gap is narrower for GCN.

If we now change the point of observation and look at the performance of different methods given the number of iterations, differences

Table 3
Results on PolitiFact. For each AL strategy, we show the AUC at key iterations. Under the number of iterations – in round brackets – we placed the percentage of the dataset that has been selected and used as training. In addition to that, we must also factor in the 10% validation set that is part of the final fact-checking budget used. With an asterisk we have marked our novel method. DUS = Deep Unseen Sampling, DES = Deep Error Sampling. Results are averaged over 5 runs and reported with their standard deviations.

Results on the Politifact dataset					
AL strategy	Iterations				
metric: AUC	8 (12%)	11 (17%)	14 (22%)	17 (27%)	20 (31%)
GAT					
Random	0.83 ± 0.12	0.85 ± 0.07	0.89 ± 0.09	0.88 ± 0.08	0.89 ± 0.07
Uncertainty	0.86 ± 0.09	0.86 ± 0.07	0.88 ± 0.09	**0.91 ± 0.07**	0.91 ± 0.08
Uncertainty + Diversity	**0.87 ± 0.08**	0.84 ± 0.10	**0.90 ± 0.09**	**0.91 ± 0.07**	0.91 ± 0.08
DUS	0.79 ± 0.11	0.86 ± 0.08	0.88 ± 0.010	0.89 ± 0.09	0.90 ± 0.09
DUS + Diversity	0.76 ± 0.10	0.86 ± 0.09	0.88 ± 0.11	**0.91 ± 0.08**	0.91 ± 0.07
DES*	0.83 ± 0.06	**0.89 ± 0.05**	**0.90 ± 0.07**	**0.91 ± 0.03**	**0.92 ± 0.04**
DES* + Diversity	0.86 ± 0.09	0.86 ± 0.06	**0.90 ± 0.08**	**0.91 ± 0.5**	0.91 ± 0.07
DES* + Uncertainty	0.84 ± 0.05	0.86 ± 0.04	0.88 ± 0.07	0.90 ± 0.05	0.91 ± 0.03
GraphSAGE					
Random	0.85 ± 0.8	0.85 ± 0.10	0.90 ± 0.09	0.90 ± 0.09	0.90 ± 0.11
Uncertainty	0.84 ± 0.08	0.89 ± 0.11	0.89 ± 0.10	0.90 ± 0.11	0.91 ± 0.09
Uncertainty + Diversity	0.82 ± 0.09	0.88 ± 0.08	0.90 ± 0.07	0.91 ± 0.10	**0.92 ± 0.08**
DUS	0.80 ± 0.11	0.86 ± 0.09	0.88 ± 0.07	0.90 ± 0.09	0.90 ± 0.08
DUS + Diversity	0.78 ± 0.7	0.87 ± 0.10	0.88 ± 0.09	0.91 ± 0.08	0.91 ± 0.08
DES*	**0.88 ± 0.4**	**0.89 ± 0.05**	**0.89 ± 0.06**	0.91 ± 0.07	0.91 ± 0.06
DES* + Diversity	0.85 ± 0.08	**0.89 ± 0.05**	**0.90 ± 0.04**	**0.92 ± 0.04**	0.91 ± 0.07
DES* + Uncertainty	0.87 ± 0.06	**0.89 ± 0.03**	**0.90 ± 0.05**	0.91 ± 0.04	**0.92 ± 0.04**
GCN					
Random	0.87 ± 0.09	**0.90 ± 0.09**	0.91 ± 0.10	0.89 ± 0.08	0.89 ± 0.09
Uncertainty	**0.90 ± 0.08**	0.86 ± 0.10	0.91 ± 0.7	**0.93 ± 0.09**	0.93 ± 0.08
Uncertainty + Diversity	0.85 ± 0.09	0.87 ± 0.07	0.91 ± 0.9	0.92 ± 0.07	0.92 ± 0.08
DUS	0.87 ± 0.10	**0.90 ± 0.07**	0.91 ± 0.11	**0.93 ± 0.09**	0.93 ± 0.08
DUS + Diversity	0.86 ± 0.07	**0.90 ± 0.07**	0.90 ± 0.08	**0.93 ± 0.09**	0.93 ± 0.10
DES*	0.87 ± 0.10	0.87 ± 0.09	**0.92 ± 0.07**	**0.93 ± 0.07**	0.93 ± 0.07
DES* + Diversity	0.88 ± 0.08	0.87 ± 0.05	0.89 ± 0.06	**0.93 ± 0.05**	**0.94 ± 0.06**
DES* + Uncertainty	0.88 ± 0.06	0.89 ± 0.05	0.91 ± 0.05	**0.93 ± 0.04**	0.92 ± 0.04

Table 4
Results on FbMultiLingMisinfo. For each AL strategy, we show how many iterations are needed to reach a desired level of expected/average AUC.

FbMultiLingMisinfo. Numbers of iterations required to reach a desired level of AUC									
AL strategy	Expected average AUC								
metric: #iterations (lower is better)	0.73	0.75	0.77	0.79	0.81	0.83	0.85	0.87	0.89
GAT									
Random	33	37	46	51	57	69	100	–	–
Uncertainty	20	27	36	44	54	66	80	100	–
Uncertainty + Diversity	**17**	25	34	38	44	57	78	96	–
DUS	24	35	40	53	59	76	94	–	–
DUS + Diversity	30	37	44	57	65	77	97	–	–
DES*	19	26	35	43	47	67	79	95	–
DES* + Diversity	18	26	32	37	44	58	79	–	–
DES* + Uncertainty	18	**22**	**30**	**36**	**43**	**54**	**72**	**90**	–
GraphSAGE									
Random	18	19	33	35	37	51	60	97	–
Uncertainty	17	20	30	33	37	39	53	69	93
Uncertainty + Diversity	**9**	**14**	**18**	35	38	40	52	59	90
DUS	16	20	36	38	40	57	68	–	–
DUS + Diversity	14	18	35	38	39	54	59	80	–
DES*	13	15	20	29	34	37	50	64	92
DES* + Diversity	15	17	23	26	33	37	46	57	90
DES* + Uncertainty	11	15	**18**	**24**	**32**	**36**	**44**	**54**	**88**
GCN									
Random	18	32	37	40	56	77	96	–	–
Uncertainty	15	18	23	36	49	59	97	–	–
Uncertainty + Diversity	16	20	26	32	38	58	92	–	–
DUS	**11**	**15**	20	33	40	83	93	–	–
DUS + Diversity	14	16	24	35	50	71	93	–	–
DES*	13	16	**18**	34	37	62	80	96	–
DES* + Diversity	13	**15**	19	**30**	**35**	**53**	**72**	**88**	–
DES* + Uncertainty	14	20	27	37	47	59	79	91	–

Table 5
Results on PolitiFact. For each AL strategy, we show how many iterations are needed to reach a desired level of expected/average AUC.

PolitiFact. Numbers of iterations required to reach a desired level of AUC						
AL strategy	Expected average AUC					
metric: #iterations (lower is better)	0.83	0.86	0.88	0.90	0.92	0.94
GAT						
Random	8	13	17	–	–	–
Uncertainty	6	8	11	16	–	–
Uncertainty + Diversity	**4**	**7**	10	**11**	–	–
DUS	9	10	11	20	–	–
DUS + Diversity	10	12	14	16	–	–
DES*	8	9	10	**11**	**20**	–
DES* + Diversity	6	8	**9**	**11**	–	–
DES* + Uncertainty	5	9	11	17	–	–
GraphSAGE						
Random	6	**7**	10	11	–	–
Uncertainty	7	8	10	17	–	–
Uncertainty + Diversity	8	9	10	13	20	–
DUS	9	10	12	17	–	–
DUS + Diversity	9	11	13	16	–	–
DES*	5	**7**	**8**	16	–	–
DES* + Diversity	7	9	10	13	**17**	–
DES* + Uncertainty	**4**	**7**	9	**11**	20	–
GCN						
Random	5	7	10	13	–	–
Uncertainty	**3**	7	9	12	15	–
Uncertainty + Diversity	5	8	11	14	17	–
DUS	4	**5**	9	13	15	–
DUS + Diversity	6	8	10	11	14	–
DES*	5	7	9	13	14	–
DES* + Diversity	4	6	**8**	**10**	**13**	18
DES* + Uncertainty	4	**5**	**8**	12	14	–

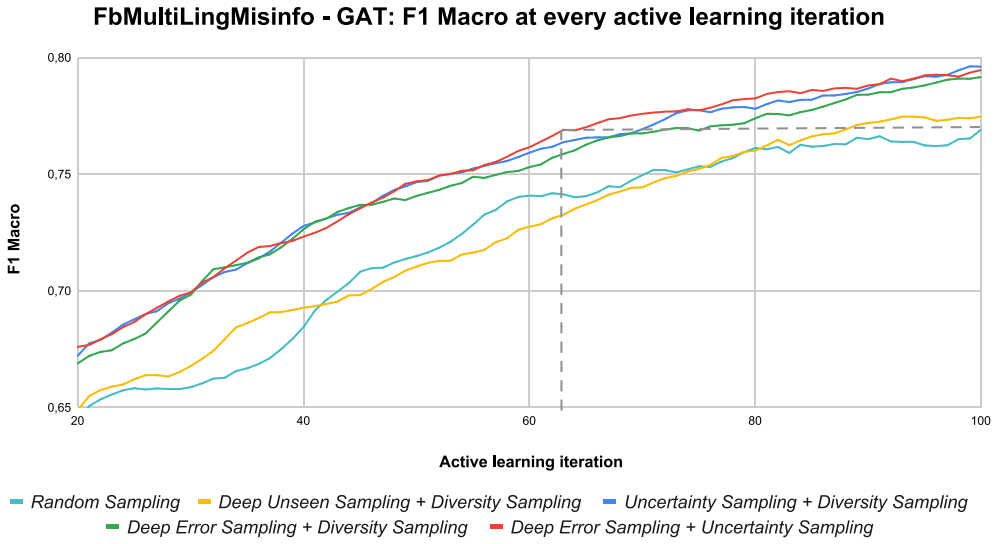


Fig. 3. F1 Macro at each iteration for 5 AL strategies using GAT on FbMultiLingMisinfo. Deep Unseen + Diversity, Deep Error Uncertainty and Uncertainty + Diversity all perform similarly and better than both Random and Deep Unseen + Diversity.

might seem less remarkable. Tables 2 and 3 still show that all tested AL strategies except Deep Unseen Sampling outperform random sampling — sometimes to a significant extent. On average, however, the AUC is only 2% higher with little difference among Uncertainty Sampling + Diversity Sampling, Deep Error Sampling + Diversity Sampling, and Deep Error Sampling + Uncertainty Sampling. While 2% might seem low, it is worth mentioning that AUC is a demanding metric, and that – in a large news ecosystem like the web or a social network – even small increases might lead to substantial improvements in the information quality inside the system. Let us now review the results more in depth, and for the two datasets separately.

On FbMultiLingMisinfo, for GAT, Uncertainty + Diversity and Deep Error Sampling + Uncertainty performed equally good or better than any other sampling strategy, while for GraphSAGE also Deep Error Sampling + Diversity reached top performance. In general, regardless of the GNN used, random and Deep Unseen Sampling were always the two methods delivering the worst results – as well exemplified in Figs. 3–5. Finally, when using GCN, Deep Error Sampling + Diversity showed the best performance overall for AUC; its performance in terms of F1 Macro are more clearly highlighted in Fig. 4 – with Deep Error Sampling + Uncertainty and Uncertainty + Diversity as second and third best performing methods respectively. On Politifact, results are more nuanced. Especially when using GCN as the base fake news classifier, no

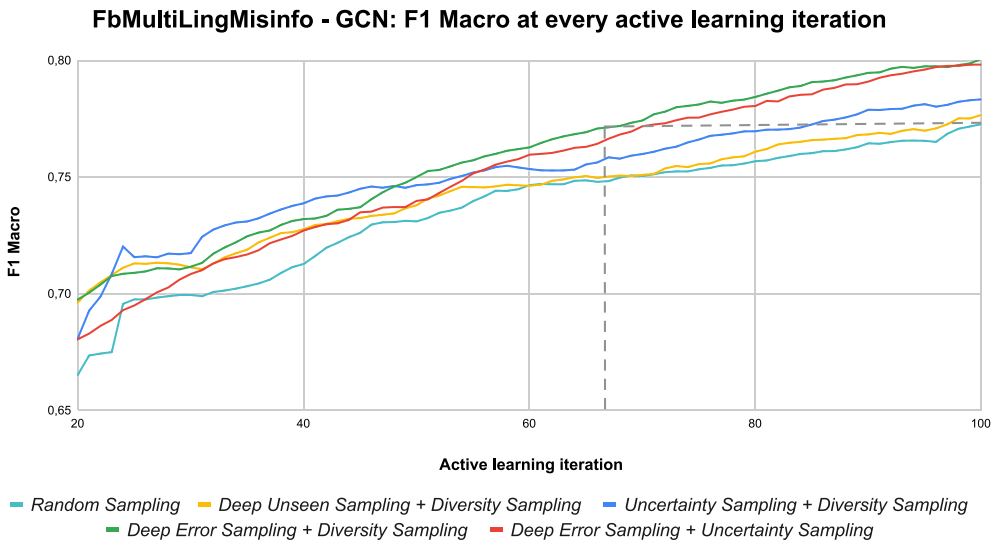


Fig. 4. F1 Macro at each iteration for 5 AL strategies using GCN on FbMultiLingMisinfo. Deep Error + Diversity and Deep Error + Uncertainty outperform all the other methods.

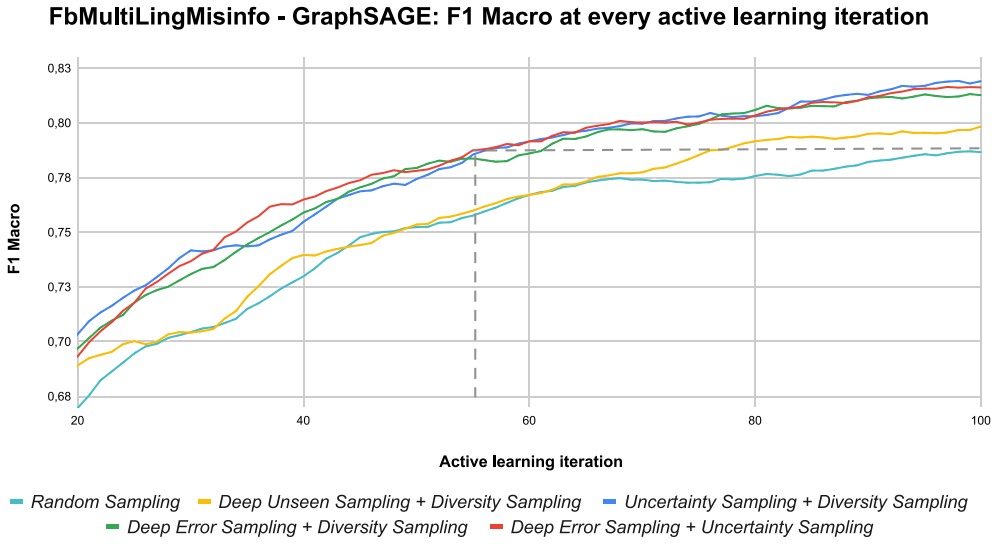


Fig. 5. F1 Macro at each iteration for 5 AL strategies using GraphSAGE on FbMultiLingMisinfo. Deep Unseen + Diversity, Deep Error + Uncertainty and Uncertainty + Diversity all perform similarly and better than both Random and Deep Unseen + Diversity.

AL method clearly outperforms all the others. When using graphSAGE, Deep Error Sampling and Deep Error Sampling + Uncertainty start emerging as the top performing methods — in the first and second half of the process respectively. Finally it is worth noting that, when using GAT and starting from the 11th iteration, the Deep Error Sampling always reaches the best performance in terms of AUC. The most likely reason why no active learning strategy seems to prevail is that Politifact is too small and homogeneous to really make AL necessary. Overall, while in many cases other active learning strategies perform as well as our proposed Deep Error Sampling, for both FbMultiLingMisinfo and PolitiFact DES produces more stable outcomes — as measure by the lower variance in the results. In addition, when Deep Error Sampling is coupled with either Diversity or Uncertainty Sampling — result variance between rounds seems to further decrease. Our method thus adds to those already available with its own uniqueness and opens the way for new combinations of more robust active learning strategies.

To conclude, the most remarkable result of our enquiry on active learning for misinformation detection is what we show in Figs. 3, 4 and 5. The three best AL strategy require only between 45 and 65 iterations to reach the same F1 Macro that random sampling reaches at iteration 100. More generally, given a certain F1 Macro score on FbMultiLingMisinfo, the three plots also indicate the number of iterations needed to reach that level of performance with the three best and the two worst AL strategies for GAT, GCN and GraphSAGE respectively. In the worst cases, random and Deep Unseen Sampling require up to 50% more iterations than Deep Error Sampling + Uncertainty, Deep Error Sampling + Diversity and Uncertainty + Diversity to reach the same F1 Macro score — and the gap seems to increase as the performance of the model increases. These promising results pave the way for a great reduction in time and money spent for annotating online news — thus making the training of GNN-based fake news detectors more affordable.

8. Conclusion & future work

In this work we presented an in-depth analysis of active learning strategies in the contest of automatic misinformation detection, we proposed a new deep active learning architecture that, when coupled with uncertainty sampling, performs equally or better than the most common AL strategies and the only proposed active learning procedure specifically targeting fake news detection. A key finding is that, in the

context GNN-based models for misinformation detection, compared to random sampling AL allows – on average – to achieve a 2% increase in AUC for the same percentage of third-party fact-checked news and to save up to 25% of labeling effort for a desired level of classification performance. While this direction seems promising, more ablation studies are needed to find the optimal number of URLs that should be labeled at every AL iteration. Experiments on much larger datasets would also help gauging the feasibility of our proposed method in a real world scenario. More in general, while hard to do, it would also make sense to jointly optimize the hyper-parameters of both the misinformation classifier and of the Deep AL architecture. Finally, the Deep AL model itself could be made much more complex, possibly leading to much greater improvements.

9. Ethical considerations

We acknowledge that automatic misinformation detection poses well-documented risks, including the marginalization of minority discourse through disparate false positive rates. At the same time, it also contributes to fighting misinformation campaigns that usually target marginalized groups, such as immigrants. The ethical considerations in this case affect all automated misinformation finding tools, and are not specific to our work, which uses well-established practices. The main subject of the work is in fact our Active Learning algorithm, the main purpose of which is to improve the performance of Misinformation Detection models. It is the use of the latter that can lead to ethical concerns and not our algorithm.

CRediT authorship contribution statement

Giorgio Barnabò: Conceptualization, Methodology, Software, Data curation, Writing – original draft. **Federico Siciliano:** Conceptualization, Methodology, Software, Writing – review & editing. **Carlos Castillo:** Supervision, Writing – review & editing. **Stefano Leonardi:** Supervision, Writing – review & editing, Funding acquisition. **Preslav Nakov:** Supervision, Writing – review & editing. **Giovanni Da San Martino:** Supervision, Writing – review & editing. **Fabrizio Silvestri:** Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Stefano Leonardi reports financial support was provided by European Commission. Stefano Leonardi reports financial support was provided by European Research Council. Stefano Leonardi reports financial support was provided by Government of Italy Ministry of Educaton niversity and Research. One of the co-authors — abrizio Silvestri — is in the editorial board of the journal

Data availability

Data can be obtained through a simple application.

Acknowledgments

This research was supported by the Italian Ministry of Education, University and Research (MIUR) under the grant "Dipartimenti di eccellenza 2018–2022" of the Department of Computer Science and the Department of Computer Engineering at Sapienza University of Rome. It was also partially supported by the European Research Council Advanced Grant 788893 AMDROMA "Algorithmic and Mechanism Design Research in Online Markets", the EC H2020RIA project "SoBigData++" (871042), the MIUR PRIN project ALGADIMAR "Algorithms, Games, and Digital Markets", and the project SERICS (PE00000014) under the NRRP MUR program funded by the European Union - NextGenerationEU. All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

References

[1] F. Monti, F. Frasca, D. Eynard, D. Mannion, M.M. Bronstein, Fake news detection on social media using geometric deep learning, 2019, arXiv preprint arXiv:1902.06673.
[2] T. Bian, X. Xiao, T. Xu, P. Zhao, W. Huang, Y. Rong, J. Huang, Rumor detection on social media with bi-directional graph convolutional networks, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, (01) 2020, pp. 549–556.
[3] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, 2016, arXiv preprint arXiv:1609.02907.
[4] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, Y. Bengio, Graph attention networks, 2017, arXiv preprint arXiv:1710.10903.
[5] W.L. Hamilton, R. Ying, J. Leskovec, Inductive representation learning on large graphs, in: Proceedings of the 31st International Conference on Neural Information Processing Systems, 2017, pp. 1025–1035.
[6] R.M. Monarch, Human-in-the-Loop Machine Learning: Active Learning and Annotation for Human-Centered AI, Simon and Schuster, 2021.
[7] P. Kumar, A. Gupta, Active learning query strategies for classification, regression, and clustering: A survey, J. Comput. Sci. Tech. 35 (4) (2020) 913–945.
[8] K. Shu, D. Mahudeswaran, S. Wang, D. Lee, H. Liu, FakeNewsNet: A data repository with news content, social context, and spatiotemporal information for studying fake news on social media, Big Data 8 (3) (2020) 171–188.
[9] G. Barnabò, F. Siciliano, C. Castillo, S. Leonardi, P. Nakov, G. Da San Martino, F. Silvestri, FbMultiLingMisinfo: Challenging large-scale multilingual benchmark for misinformation detection, in: International Joint Conference on Neural Networks, IJCNN, IEEE, 2022 pp. 1–8.
[10] S. Vosoughi, D. Roy, S. Aral, The spread of true and false news online, Science 359 (6380) (2018) 1146–1151.
[11] Y. Ren, B. Wang, J. Zhang, Y. Chang, Adversarial active learning based heterogeneous graph neural network for fake news detection, in: 2020 IEEE International Conference on Data Mining, ICDM, IEEE, 2020, pp. 452–461.
[12] J. Yu, Q. Huang, X. Zhou, Y. Sha, IARnet: An information aggregating and reasoning network over heterogeneous graph for fake news detection, in: 2020 International Joint Conference on Neural Networks, IJCNN, IEEE, 2020, pp. 1–9.
[13] V.-H. Nguyen, K. Sugiyama, P. Nakov, M.-Y. Kan, FANG: Leveraging social context for fake news detection using graph representation, in: Proceedings of the 29th ACM International Conference on Information & Knowledge Management, 2020, pp. 1165–1174.
[14] S. Chandra, P. Mishra, H. Yannakoudakis, M. Nimishakavi, M. Saeidi, E. Shutova, Graph-based modeling of online communities for fake news detection, 2020, arXiv preprint arXiv:2008.06274.
[15] Y.-J. Lu, C.-T. Li, GCAN: Graph-aware co-attention networks for explainable fake news detection on social media, 2020, arXiv preprint arXiv:2004.11648.
[16] A. Silva, Y. Han, L. Luo, S. Karunasekera, C. Leckie, Propagation2Vec: Embedding partial propagation networks for explainable fake news early detection, Inf. Process. Manage. 58 (5) (2021) 102618.
[17] Y. Han, S. Karunasekera, C. Leckie, Graph neural networks with continual learning for fake news detection from social media, 2020, arXiv preprint arXiv:2007.03316.
[18] C. Song, K. Shu, B. Wu, Temporally evolving graph neural network for fake news detection, Inf. Process. Manage. 58 (6) (2021) 102712.
[19] Y. Liu, Y.-F.B. Wu, Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks, in: Thirty-Second AAAI Conference on Artificial Intelligence, 2018.
[20] Y. Dou, K. Shu, C. Xia, P.S. Yu, L. Sun, User preference-aware fake news detection, in: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Association for Computing Machinery, New York, NY, USA, 2021, pp. 2051–2055, URL https://doi.org/10.1145/3404835.3462990.
[21] Q. Huang, J. Yu, J. Wu, B. Wang, Heterogeneous graph attention networks for early detection of rumors on twitter, in: 2020 International Joint Conference on Neural Networks, IJCNN, IEEE, 2020, pp. 1–8.
[22] Y. Ren, J. Zhang, Fake news detection on news-oriented heterogeneous information networks through hierarchical graph attention, in: 2021 International Joint Conference on Neural Networks, IJCNN, IEEE, 2021, pp. 1–8.
[23] K. Shu, S. Wang, H. Liu, Beyond news contents: The role of social context for fake news detection, in: Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, 2019, pp. 312–320.
[24] J. Zhang, B. Dong, S.Y. Philip, FakeDetector: Effective fake news detection with deep diffusive neural network, in: 2020 IEEE 36th International Conference on Data Engineering, ICDE, IEEE, 2020, pp. 1826–1829.
[25] C. Yuan, Q. Ma, W. Zhou, J. Han, S. Hu, Early detection of fake news by utilizing the credibility of news, publishers, and users based on weakly supervised learning, 2020, arXiv preprint arXiv:2012.04233.
[26] E. Lughofer, On-line active learning: A new paradigm to improve practical useability of data stream modeling methods, Inform. Sci. 415 (2017) 356–376.
[27] Ł. Korycki, A. Cano, B. Krawczyk, Active learning with abstaining classifiers for imbalanced drifting data streams, in: 2019 IEEE International Conference on Big Data, Big Data, IEEE, 2019, pp. 2334–2343.

[28] L. Cui, X. Tang, S. Katariya, N. Rao, P. Agrawal, K. Subbian, D. Lee, ALLIE: Active learning on large-scale imbalanced graphs, in: Proceedings of the ACM Web Conference 2022, 2022, pp. 690–698.
[29] P. Ren, Y. Xiao, X. Chang, P.-Y. Huang, Z. Li, B.B. Gupta, X. Chen, X. Wang, A survey of deep active learning, ACM Comput. Surv. 54 (9) (2021) 1–40.
[30] K. Madhawa, T. Murata, Active learning for node classification: An evaluation, Entropy 22 (10) (2020) 1164.
[31] J. Liu, Y. Wang, B. Hooi, R. Yang, X. Xiao, Active learning for node classification: The additional learning ability from unlabelled nodes, 2020, arXiv preprint arXiv:2012.07065.
[32] S.D. Bhattacharjee, A. Talukder, B.V. Balantrapu, Active learning based news veracity detection with feature weighting and deep-shallow fusion, in: 2017 IEEE International Conference on Big Data, Big Data, IEEE, 2017, pp. 556–565.
[33] K. Shu, D. Mahudeswaran, S. Wang, H. Liu, Hierarchical propagation networks for fake news detection: Investigation and exploitation, in: Proceedings of the International AAAI Conference on Web and Social Media, vol. 14, 2020, pp. 626–637.
[34] K. Shu, G. Zheng, Y. Li, S. Mukherjee, A.H. Awadallah, S. Ruston, H. Liu, Leveraging multi-source weak social supervision for early detection of fake news, 2020, arXiv preprint arXiv:2004.01732.
[35] X. Zhou, R. Zafarani, Network-based fake news detection: A pattern-driven approach, ACM SIGKDD Explor. Newsl. 21 (2) (2019) 48–60.
[36] X. Yang, Y. Lyu, T. Tian, Y. Liu, Y. Liu, X. Zhang, Rumor detection on social media with graph structured adversarial learning, in: IJCAI, 2020, pp. 1417–1423.
[37] A. Lao, C. Shi, Y. Yang, Rumor detection with field of linear and non-linear propagation, in: Proceedings of the Web Conference 2021, 2021, pp. 3178–3187.
[38] A. D'Ulizia, M.C. Caschera, F. Ferri, P. Grifoni, Fake news detection: A survey of evaluation datasets, PeerJ Comput. Sci. 7 (2021) e518.
[39] J. Nørregaard, B.D. Horne, S. Adalı, NELA-GT-2018: A large multi-labelled news dataset for the study of misinformation in news articles, in: Proceedings of the International AAAI Conference on Web and Social Media, vol. 13, 2019, pp. 630–638.
[40] M. Gruppi, B.D. Horne, S. Adalı, NELA-GT-2020: A large multi-labelled news dataset for the study of misinformation in news articles, 2021, arXiv preprint arXiv:2102.04567.
[41] S. Messing, C. DeGregorio, B. Hillenbrand, G. King, S. Mahanti, Z. Mukerjee, C. Nayak, N. Persily, B. State, A. Wilkins, Facebook privacy-protected full URLs data set, 2020, http://dx.doi.org/10.7910/DVN/TDOAPG.
[42] N. Reimers, I. Gurevych, Sentence-BERT: Sentence embeddings using siamese BERT-networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2019, URL http://arxiv.org/abs/1908.10084.