applied sciences

MDPI

Article

A Comparative Analysis of Active Learning for Rumor Detection on Social Media Platforms

Feng Yi *, Hongsheng Liu, Huaiwen He and Lei Su

School of Computer Science, Zhongshan Institute, University of Electronic Science and Technology of China, Zhongshan 528400, China; luhorsen@outlook.com (H.L.); hehw3@mail2.sysu.edu.cn (H.H.); sulei@zsc.edu.cn (L.S.)

* Correspondence: jmyf_bj@hotmail.com; Tel.: +86-0760-88227202

Abstract: In recent years, the ubiquity of social networks has transformed them into essential platforms for information dissemination. However, the unmoderated nature of social networks and the advent of advanced machine learning techniques, including generative models such as GPT and diffusion models, have facilitated the propagation of rumors, posing challenges to society. Detecting and countering these rumors to mitigate their adverse effects on individuals and society is imperative. Automatic rumor detection, typically framed as a binary classification problem, predominantly relies on supervised machine learning models, necessitating substantial labeled data; yet, the scarcity of labeled datasets due to the high cost of fact-checking and annotation hinders the application of machine learning for rumor detection. In this study, we address this challenge through active learning. We assess various query strategies across different machine learning models and datasets in order to offer a comparative analysis. Our findings reveal that active learning reduces labeling time and costs while achieving comparable rumor detection performance. Furthermore, we advocate for the use of machine learning models with nonlinear classification boundaries on complex environmental datasets for more effective rumor detection.

Keywords: rumor detection; active learning; active learning query strategy; social networks

Citation: Yi, F.; Liu, H.; He, H.; Su, L. A Comparative Analysis of Active Learning for Rumor Detection on Social Media Platforms. *Appl. Sci.* **2023**, *13*, 12098. https://doi.org/10.3390/app132212098

Academic Editor: Luis Javier Garcia Villalba

Received: 30 September 2023
Revised: 2 November 2023
Accepted: 4 November 2023
Published: 7 November 2023

1. Introduction

With the proliferation of social media platforms such as `Twitter` (https://twitter.com/, accessed on 3 November 2023) and `Sina Weibo` (https://weibo.com/, accessed on 3 November 2023) and rapid development of smart mobile devices, people increasingly tend to consume news from social media platforms rather than from traditional news sources [1]. According to a report by the Pew research center, more than half of Twitter user regularly access news on the site [2].

The anonymity and openness of social media enable users to consume and share news as well as to generate real-time information. When events such as earthquakes or accidents occur, smart mobile devices can act as real-time news sensors, allowing people to immediately upload information to social media. This has greatly changed the propagation and timeliness of traditional news media.

Nevertheless, the convenience of information dissemination on social media platforms facilitates the proliferation of rumors. Rumors often refer to information around which truth and sourcing are unreliable, and are likely to be generated under emergency situations [3]. Notably, most rumors exhibit distinct characteristics, enabling them to propagate faster, deeper, and further throughout social networks [4]. Beyond the inadvertent spreading of rumors, social media users may deliberately initiate and circulate rumors using sophisticated generative models, often motivated by commercial or political interests. Startlingly, it has been reported that more than a third of trending events on microblogs contain rumors [5].

The spread of rumors can pose significant threats to the credibility of the internet and have far-reaching real-life consequences, including causing public panic, disrupting the social order, eroding government credibility, and even endangering national security [6]. A notable case of rampant rumor propagation occurred during the 2016 U.S. presidential election. During the election, as many as 529 different rumor stories pertaining to presidential candidates Donald Trump and Hillary Clinton were spreading on Twitter, instantly reaching millions of voters and potentially influencing the election's outcome [7]. A more recent example revolves around the plethora of rumors regarding the COVID-19 pandemic [8]. These rumors on social platforms have significantly undermined the credibility and reliability of information shared on these platforms, consequently diminishing users' willingness to turn to social media for information. A 2021 survey conducted by the Pew Research Center [2] further underscores this decline in trust and reliance on social media for news. It revealed a decrease in the percentage of adult American users who frequently or occasionally obtain news from social media platforms, dropping from 53% in 2020 to 48% in 2021. This decline coincides with mounting criticism directed at social media and technology companies for their perceived inadequacy in curbing the spread of misleading information on their platforms. Therefore, it is of paramount importance to detect rumors spreading on social media platforms as early as possible.

Rumor detection has attracted significant attention from both social media platforms and researchers over the past decade. Typically, users on various social media platforms are encouraged to report or annotate suspicious posts as potential rumors. Subsequently, the accuracy of these possible rumors is verified with the assistance of human moderators and third-party fact-checkers. While this approach yields high-quality results, the substantial human effort required, including manual labeling and rumor verification, is challenging to reconcile with the sheer volume of emerging rumors. Therefore, there is a need for robust and efficient automated rumor detection approaches.

Automatic rumor detection is normally deemed a binary classification task, in which classifiers are employed to distinguish between rumors and non-rumors. These methods encompass a range of approaches, including traditional machine learning models [3,9] and neural network-based approaches [10–12], which all follow a supervised learning paradigm. In this paradigm, posts are first transformed into representations, which are then fed into a supervised learning model guided by ground-truth labels. Traditional machine learning-based approaches often rely on hand-crafted features, while neural network-based models automatically learn latent deep feature representations of rumors. However, both approaches require a sizable annotated dataset, such as RUMDECT [10] or PHEME [13], for training reliable classifiers.

While the aforementioned methods have demonstrated promising results, they face several significant challenges, as highlighted by previous research in the field of automatic rumor detection. One of the most critical challenge pertains to the labor-intensive and costly nature of constructing rumor datasets [14]. Labeling rumors within the ever-flowing stream of social media is a resource-intensive task associated with substantial costs. To illustrate this, consider the Sina Community Management Center's rumor reporting process (https://service.account.weibo.com/, accessed on 3 November 2023) depicted in Figure 1.

A social media user must navigate through three stages for rumor reporting: the reporting stage, the evidence stage, and the results announcement stage. The evidence stage demands that the reporting user provide proof that the post in question is indeed a rumor. Subsequently, this evidence is scrutinized by experts from the Sina platform. This process is both time-consuming and financially burdensome.

Moreover, the rapid advancement of artificial intelligence (AI), particularly the emergence of generative models such as Generative Adversarial Networks (GANs) and diffusion models, has led to an increase in manipulated multimodal rumors. These rumors may incorporate image, audio, and video data, rendering them increasingly challenging for ordinary social media users to differentiate from genuine content. A notable example is the use of DeepFakes, which leverage deep learning models to fabricate audio and video clips

of real individuals uttering or performing actions that never actually occurred. This makes rumors appear both more realistic and harder to discern [15,16].

Figure 1. The workflow of rumor reporting on the Weibo platform, showing the three stages by which a Sina user can confirm a rumor. The green English corresponds to the translation of the Chinese text just above.

Furthermore, certain rumors may contain domain-specific knowledge and can only be debunked by experts in the respective field. Annotation of previously unseen rumors often requires in-depth domain knowledge. A notable example occurred during the COVID-19 pandemic, when rumors such as "5G caused the virus" or "facemasks do not work" had to be confirmed as false by professional or authoritative medical experts rather than ordinary social media users. In more challenging scenarios, slight modifications to aspects of a non-rumor can lead to the creation of new and more convincing rumors. For instance, altering details such as the timing, location, or individuals associated with a non-rumor event can result in the fabrication of a convincing rumor. In such situations, it becomes significantly more arduous for experts to distinguish rumors from normal posts, making it a time-consuming and domain knowledge-intensive task.

Despite the growing volume of posts on social media platforms, including rumors, obtaining high-quality, large-scale, and authoritative benchmark datasets remains a daunting task. In comparison to benchmarks such as ImageNet [17], which contains 14,197,122 images and serves as a standard in visual object recognition, Table 1 shows that datasets used in recent research on rumor detection are relatively small in scale or confined to specific rumor categories.

Table 1. Common datasets use by state-of-the-art rumor detection research.

Dataset	Data Source	Rumor	Non-Rumor
KWON [18]	Twitter	47	55
PHEME [19]	Twitter	1972	3830
Medieval [20]	Twitter	9000	6000
Twitter15 [21,22]	Twitter	1116	374
Twitter16 [10,22]	Twitter	618	205
MULTI [23]	Sina Weibo	4749	4779
RUMDECT [10]	Twitter	498	494
	Sina Weibo	2313	2351

This discrepancy underscores the need for developing comprehensive benchmark datasets, particularly in the current revolutionary era in deep learning.

This epoch is frequently characterized by the phrase "Data is the new oil" [24], signifying the pivotal role of data in driving advancements across various tasks and applications through data-driven learning approaches. These approaches place heightened demands on

both the quality and quantity of data. It is crucial to recognize that the size and quality of datasets wield a profound influence on the performance and scalability of state-of-the-art (SOTA) rumor detection models [25].

In addition to the aforementioned challenges around labeling rumors and the limited scale of datasets, the performance of learned models may deteriorate due to conceptual drift. This phenomenon occurs when the distribution of features related to rumors undergoes changes over time. Typically, mitigating conceptual drift requires the continuous annotation of new datapoints and model updates. Unfortunately, this practice can be both costly and impractical. In summary, the field of automatic rumor detection faces a significant challenge in large-scale data annotation.

To address the challenges associated with rumor detection, an intuitive idea is to selectively label valuable data instead of annotating the entire dataset for training rumor detection models. Active Learning (AL) has emerged as a promising solution to overcome the key challenges outlined earlier. As a subfield of machine learning, active learning aims to create efficient training datasets by iteratively enhancing model performance through strategic sample selection. The goal is to achieve or even surpass the expected model performance with as few labeled samples as possible [26].

Active learning recognizes that not all samples in a dataset are equally crucial for training a machine learning model. Therefore, it intelligently selects a subset of the dataset for labeling by an oracle, such as a human annotator, to optimize model performance. This approach mitigates the labeling bottleneck and minimizes the costs associated with acquiring labeled data. Consequently, active learning is well-suited for rumor detection scenarios, in which a surplus of unlabeled data is available from real social media streams while labeled data remain a costly resource.

Despite the existence of comparative studies across various tasks and domains, active learning has not been extensively explored in the context of rumor detection. In this work, we present a comparative analysis of active learning techniques for rumor detection on social media platforms, aiming to answer the following key questions:

1. Can active learning effectively reduce labeling costs in the context of rumor detection while maintaining high performance?
2. Which active learning query strategies are most suitable for specific rumor detection methods?

This research seeks to shed light on the potential of active learning in improving rumor detection while addressing the practical challenges associated with labeling large datasets. Hence, we evaluate the feasibility of utilizing active learning for rumor detection on social media platforms. To assess the effectiveness of active learning, we conduct a comparative analysis of multiple supervised machine learning methods. Our evaluation is performed on two distinct datasets, and we explore how active learning can reduce both the sample size and its influence on various supervised machine learning models. The significant contributions of our work can be summarized as follows:

- To the best of our knowledge, this is the first comprehensive and comparative investigation of rumor detection using active learning, addressing an important gap in the literature.
- We examine active learning query strategies suitable for different supervised learning models in the context of automatic rumor detection within pool-based scenarios.
- Through extensive evaluation on Twitter and Weibo datasets, we demonstrate that active learning achieves faster convergence with a limited amount of annotated data, offering practical benefits for rumor detection.

The rest this paper is organized as follows. In Section 2, we provide a comprehensive review of the related literature. Section 3 outlines the process of automatic rumor detection using active learning. Section 4 presents our experimental setup. In Section 5, we discuss our experiment results. Finally, Section 6 concludes the paper and discusses directions for future work.

2. Related Works

Though automated rumor detection is not a new phenomenon, it has been increasingly drawing public attention [27]. Researchers have made various efforts to develop different techniques for automated rumor detection. In this section, we briefly review existing work in the categories of traditional machine learning-based methods and active learning.

2.1. Traditional Machine Learning-Based Rumor Detection

The objective of automated rumor detection is to distinguish between rumors and normal posts, which is typically formulated as a binary classification problem. Consequently, early automated rumor detection methods often relied on traditional classifiers, including Logistic Regression (LR), Support Vector Machine (SVM), Decision Tree Classifier (DTC), Naive Bayes (NB), Random Forest Classifier (RFC), and K-Nearest Neighbours (KNN), among others. These models are heavily dependent on handcrafted features.

Much of the initial research into rumor detection focused on the selection and extraction of features. Commonly adopted features in state-of-the-art (SOTA) rumor detection methods include those derived from text content, user profiles, and propagation patterns. These features encompass counting the number of sentiment words, URLs, retweets, hashtags, etc. Supervised machine learning methods for rumor detection predominantly concentrate on characterizing one or a combination of these features.

For instance, Castillo et al. [9] analyzed features including text content, user information, propagation patterns, and Twitter memes (hashtags, URLs, and user mentions). Qazvinian et al. [28] built Bayes classifiers using Twitter meme-based features to detect rumors on Twitter, reporting that these features differ between rumor-related tweets and normal tweets. Kwon et al. [18] introduced temporal, structural, and linguistic features, then trained DTC, RFC, and SVM classifiers to identify rumors. Liang et al. [3] proposed a user behavior-based rumor identification scheme, treating user behaviors as hidden clues to identify potential rumormongers or rumor posts on microblogs. Their study incorporated user behavior features, content-based features, propagation features, and multimedia feature, and employed classifiers such as LR, SVM, DTC, NB, and KNN. Notably, the KNN model achieved higher precision when using user behavior features.

While these models have shown acceptable performance, they rely heavily on the quality of feature engineering. Certain distinguishing features, such as hashtags, may not be available in all datasets, and the characteristics of rumors can change over time, limiting the generalizability of these models. However, when conducting research on applying active learning in rumor detection, these models may offer sufficiently convincing evidence.

2.2. Active Learning

As mentioned previously, manually labeling datasets for rumor detection can be costly, and may demand domain-specific expertise. Active learning, which aims to achieve high accuracy with minimal labeled instances, has gained significant traction in various domains, including text classification [25,29], biomedical text mining [30], and computer vision tasks [31]. Uncertainty-based sampling models have demonstrated particularly promising results, making them a predominant choice when applying active learning to social media.

For instance, McCallum et al. [32] demonstrated how traditional text classifiers can reduce their need for labeled training data through active learning by leveraging a vast pool of unlabeled documents. Siddhant et al. [33] conducted a large-scale empirical study on deep active learning, confirming its efficacy in improving deep learning performance for text classification, especially through Bayesian active learning by disagreement.

While active learning is commonly employed in text classification, its application in rumor detection has been relatively limited. Most active learning studies in the context of rumor detection have focused on fake news detection. Bhattacharjee et al. [34] explored active learning for identifying the veracity of fake news using uncertainty-based probability of classification. This approach was later extended to domain-specific and context-rich

frameworks [35]. Hasan et al. [36] introduced a fake news detection framework incorporating active learning based on entropy sampling, achieving high accuracy with a limited training dataset (4% to 28% of available data). Sahan et al. [37] conducted a comparative study of various active learning strategies on different text embeddings for text classification and fake news detection. Although there are inherent differences between rumor detection and fake news detection, lessons from active learning in the context of fake news detection can inform and benefit rumor detection tasks.

Recently, Farinneya et al. [25] introduced an active transfer learning framework for rumor detection. Their work explored different pretrained language models, estimators, and active learning strategies. Extensive experiments on the PHEME dataset revealed that supervised machine learning rumor detection models using embedded representations with limited labeled data were able to achieve similar performance to models trained on the entire dataset. The authors' goal was to design a new rumor detection model based on pretrained language models and active learning. In contrast, the present study aims to conduct a comprehensive and comparative analysis of rumor detection approaches using active learning.

3. Methodology

The aim of this paper is to assess the effectiveness of various active learning query strategies in the context of rumor detection models. Specifically, we seek to determine whether active learning enhances rumor detection performance. Our study involves comparing different query strategies and their application to different datasets using various machine learning models. In particular, we aim to identify the most suitable query strategy for different rumor detection models. In this section, the following three aspects are introduced: active learning, active learning query strategies, and rumor detection classifiers.

3.1. Active Learning

Active learning is a subfield of machine learning and artificial intelligence. It falls under the category of semi-supervised machine learning, where a learning model can iteractively request information from the user or another information source to obtain desired outputs at new datapoints [26]. In the statistics literature, it is sometimes known as "query learning" or "optimal experimental design" [38]. Active learning encompasses various problem scenarios in which a machine learning model can ask queries, such as membership query synthesis, stream-based selective sampling, and pool-based active learning. In the context of rumor detection, as discussed earlier, it is often possible to gather a substantial volume of unlabeled data, aligning with the common scenario in pool-based active learning.

Figure 2 illustrates the typical workflow cycle of active learning in pool-based scenarios. The raw dataset for rumor detection contains a small portion of labeled data and a large amount of unlabeled data, designated D_u. The labeled dataset is divided into an initialized training dataset D_l and a test dataset D_t based on a certain proportion. Let (x, y) be an instance in the raw dataset, where x is a d-dimensional feature vector and y is its corresponding label. A machine learning model, denoted as P_θ, begins with the small labeled training dataset D_l and undergoes standard supervised training to establish an initial model. This initial model is then evaluated on the unlabeled dataset D_u, with $\hat{y}$ representing the predicted label for instance x.

A query strategy is employed to compute a measurement criterion with $\hat{y}$, which is used to select one or a few of instances from D_u. The selected unlabeled instances are typically informative or representative samples, and are referred to as query instances. The query instances are then sent to an oracle for labeling. When the query instances have been labeled by querying the oracle, they are added to the training dataset D_l. The machine learning model P_θ is then retrained with the updated labeled dataset D_l and tested on the test dataset D_t to evaluate its current performance. This process is repeated until the model achieves satisfactory performance on D_t or until specific preset conditions are met.

The primary objective of active learning is to maximise the model's effectiveness by minimizing the number of samples that require manual labelling. The main challenge lies in identifying informative or representative query instances that facilitate the rapid convergence of model training.

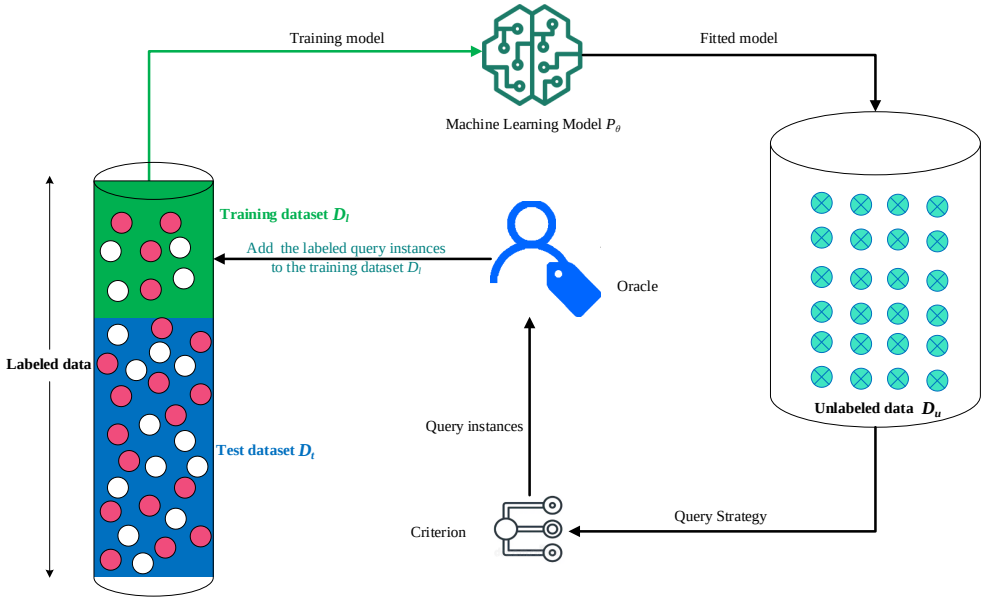


Figure 2. The workflow cycle of active learning in pool-based active learning scenarios.

3.2. Active Learning Query Strategy

As mentioned before, the critical challenge in the workflow cycle of active learning lies in selecting an appropriate query strategy, known as a selector. The query strategy evaluates the "worthiness" of unlabeled samples using a specific criterion and determines whether a sample is worthy of annotation based on its suitability. Therefore, choosing the right query strategy is pivotal in enabling the model to converge effectively with minimal training data. The choice of query strategy holds significant implications in active learning.

To date, numerous strategies have been proposed in the literature for querying unlabeled instances. These query strategies can be categorized into three main groups based on the nature of the instances they select: informative-based, representative-based, and both of these in combination. Informative-based strategies focus on the informativeness of unlabeled instances, prioritizing those with higher information content for labeling by the oracle. Typically, the informativeness of unlabeled data is assessed based on the model's uncertainty. However, informative-based strategies may overlook relationships among unlabeled instances, and often lead to the selection of multiple instances of a similar type.

On the other hand, representative strategies aim to make efficient use of the structure within the unlabeled data when selecting candidate query instances. Additionally, they strive to address the challenges encountered by informative query strategies. Representative strategies can help to alleviate issue of sampling bias by selecting instances from diverse regions within the input space. Combining informative and representative strategies can strike a balance between measures of informativeness and representativeness. It is worth noting that an increase in the informativeness of a selected instance may come at the cost of reduced representativeness.

In the following subsections, we provide a detailed description of the strategies employed in this paper.

3.2.1. Uncertainty Sampling

Uncertainty sampling is a typical informative-based strategy and is the most popular query strategy for active learning. It assumes that the uncertainty samples provide more information for training a machine learning model if they are labeled. The rationale for this

is that instances with lower certainty are typically located near the decision boundary of the classification, while highly certain instances are usually far from the decision boundary. Therefore, instances that are distant from the decision boundary are often considered redundant. The uncertainty sampling strategy selects an instance with the lowest confidence predicted by the current machine learning model as the query instance.

Common criteria for evaluating the uncertainty include least confidence, uncertainty margin, and entropy.

Least confidence is a strategy based on the prediction uncertainty. It measures uncertainty as the level of confidence in the most likely label. It is based on the probability of the top-class label with the highest posterior probability for a given instance. The uncertainty of an unlabeled instance is defined by Equation (1):

$$x_s = \arg \max_{x \in D_U} \{1 - P_\theta(\hat{y}|x)\} = \arg \min_{x \in D_U} P_\theta(\hat{y}|x) \quad (1)$$

where $P_\theta(\hat{y}|x)$ is the probability of the top-class label $\hat{y}$ with the highest posterior probability (for instance, x), D_U represents the unlabelled data pool, and x_s is the uncertainty score of the query instance. The least confidence criterion strives to find the most indistinguishable instance of the current model as the query instance.

Least confidence only considers the probability of the best prediction class label, ignoring the information from other class labels. As an improved query strategy, margin sampling can calculate the difference between the two most confident posterior probabilities, as defined by Equation (2):

$$x_s = \arg \min_{x \in D_U} \{P_\theta(\hat{y}_1|x) - P_\theta(\hat{y}_2|x)\}, \quad (2)$$

where $P_\theta(\hat{y}_1|x)$ and $P_\theta(\hat{y}_2|x)$ are the top-1 and top-2 posterior probabilities. The instance with the smallest difference is defined as a hard-to-classify instance for labeling.

In order to further consider the information of all class, another more typical measure of uncertainty is entropy, which is defined by Equation (3):

$$x_s = \arg \max_{x \in D_U} -\sum_{i}^{C} P_\theta(\hat{y}_i|x) \log P_\theta(\hat{y}_i|x) = \arg \min_{x \in D_U} \sum_{i}^{C} P_\theta(\hat{y}_i|x) \log P_\theta(\hat{y}_i|x), \quad (3)$$

where C is the number of classes. The entropy measures the purity of a class for one sample, with larger entropy denoting higher uncertainty. The instance with the largest entropy is selected as query instance.

3.2.2. Query by Committee (QBC)

QBC is another typical informative-based strategy; it is based on the inconsistency of ensemble learning. In this strategy, a committee is composed by training multiple classifiers on different subsets of instances drawn from the labeled dataset. The fundamental assumption of QBC is that different classifiers should exhibit consistency with the provided labeled data instances. Hence, the query instance is selected based on the unlabeled instance that demonstrates the highest disagreement among the committee members in label prediction.

There are two ways to construct this committee, namely, boosting and bagging. In query by bagging, a committee of m classifiers is created by applying bootstrap aggregating, which involves randomly sampling with replacement m times from the labeled training data. In query by boosting, the random instances are bootstrapped with replacement from available labeled training data.

There are two kinds of indicators for measuring disagreement, namely, vote entropy and the average Kullback–Leibler (KL) divergence. Vote entropy identifies the instances with the largest entropy among the predicted class labels. Such instances are considered hard samples, and are selected as query instances for labeling. The average KL divergence

measure identifies the most informative query as the one with the largest average difference between the label distributions of any one committee member and the consensus [26].

3.2.3. Expected Error Reduction (EER)

EER is a informative-based strategy which selects the next instance that maximally reduces the generalization "error" or "loss" in expectation [39]. It takes into account the uncertainty or informativeness of unlabeled instances and measures the potential impact of querying them on the overall error reduction of the learning model.

The key idea behind EER is to estimate the expected reduction in error that can be achieved by labeling specific instances. It involves selecting those samples expected to have the greatest impact on improving the model's performance. This is achieved by considering the uncertainty or lack of confidence in the current predictions made by the learning model. The intuition is that by querying instances that are difficult to classify or that lie near the decision boundary, the model can obtain crucial information to refine its decision-making process.

3.2.4. Graph Density Strategy

The graph density strategy is a representative-based strategy that employs a graph structure to identify the most representative unlabeled datapoints. The underlying intuition of the graph density strategy is that representative data points for a specific class are typically well-embedded in the graph structure, resulting in many edges $\gg k$ with high weights. To implement the graph density strategy [40], a k-nearest neighbor graph is constructed in which $e_{ij} = 1$ if $d(x_i, x_j)$ is one of the k smallest distances of x_i with Manhattan distance d. The strategy uses a weighted matrix with a Gaussian kernel, to rank all data points based on their representativeness, as defined in Equation (4):

$$W_{ij} = e_{ij} \exp\left\{\frac{-d(x_i, x_j)}{2\sigma^2}\right\}. \tag{4}$$

3.2.5. Querying Informative and Representative Examples (QUIRE)

QUIRE combines the informative and representative strategies, taking a min–max view of active learning and providing a systematic way to measure and combine informativeness and representativeness. QUIRE measures both the informativeness and representativeness of an instance; specifically, the informativeness of an instance x is measured using its prediction uncertainty based on the labeled data, while the representativeness of x is measured by its prediction uncertainty based on the unlabeled data [41].

3.2.6. Information Density Weighted Strategy

The information density weighted strategy [42] is another combination of informative and representative strategies. Informative-based strategies may tend to select unlabeled instances that lie along the classification boundaries even when these instances are outliers that are not representative of the broader distribution in the input space. This strategy introduces the concept of information density (ID), as defined Equation (5):

$$\phi_{ID}(x) = \phi_{SE}(x) \times \left(\frac{1}{|D_u|} \sum_{i=1}^{|D_u|} sim(x, x_u)\right)^{\beta}, \tag{5}$$

where $\phi_{SE}(x)$ measures the "base" informativeness of an unlabeled instance x; the terms in parentheses in Equation (5) represent the similarity of x to all other unlabeled instances in D_u, while the parameter β controls the relative importance of the representative term. The information density weighted strategy effectively combines uncertainty and diversity in active selection.

3.3. Rumor Detection Classifier

In this paper, we explore a wide range of supervised learning classification models for rumor detection and subject them to extensive study using different active learning strategies. These classifiers LR, SVM, DTC, NB, RFC, KNN, the Gaussian Process (GP) classifier, Multi-Layer Perceptron (MLP), Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA), and AdaBoost (Ada). Unless otherwise specified, all model parameters were set to their default values.

We employed two LR models: one trained with the standard approach and the other utilizing Stochastic Gradient Descent (SGD). For simplicity, we refer to these as LR and LR(SGD), respectively. Additionally, we employed three SVM classifier models: one with a linear kernel, denoted as SVM(Linear), another with a Radial Basis Function (RBF) kernel, denoted as SVM(RBF), and the third with an RBF kernel trained using SGD, which we denote as SVM(SGD). It is important to note that, unlike LR, SVM models with a linear kernel or trained with SGD lack the ability to predict probabilities for object classes. This limitation confines their usage to representative-based query strategies. Furthermore, we assessed the performance of DTC using both the Gini and Entropy criteria, which are denoted as DTC(Gini) and DTC(Entropy), respectively.

Feature extraction is one of the most crucial phases of supervised machine learning, and has a significant influence on classification accuracy. Researchers seeking to achieve better rumor classification performance have experimented with combinations of various features and supervised machine learning classifiers. Several common features, including content-based, user-based, propagation-based and behavior-based features, were selected for experimentation in our study. The diverse features extracted from online social media posts play a vital role in rumor detection using machine learning models.

4. Experimental Setup

In this section, we discuss the datasets and the implementation details of our experiments.

4.1. Dataset

We conducted experiments on two publicly available datasets, PHEME and RUMDECT, collected from Twitter and Weibo, respectively. These datasets are widely recognized and commonly used in the field of rumor detection.

As shown in Table 1, the PHEME dataset [13] sourced from Twitter comprises 1972 instances of rumors and 3830 instances of non-rumors. This benchmark contains five newsworthy cases of breaking news that provoked a high number (exceeding 100) of retweets. As these five events propagated through social media, four distinct rumors emerged from the conversations surrounding them. Table 2 illustrates typical examples of rumor and non-rumor tweets from this dataset. Additionally, this benchmark contains rich features, including post text, user information, and release timestamps.

Table 2. Typical tweets in the PHEME dataset.

Events	Rumor	Non-Rumor
Ottawa Shooting	#Ottawa police confirm shooting at War Memorial. Reports say victim may be a soldier.	Canine unit running to Parliament Hill
	One person shot outside Centre Block, a second wounded inside the building in Parliament Hill shooting.	I covered Polytechnique, Concordia and Dawson shootings. Remember, at least half of what you hear about Ottawa shootings will prove untrue.

Table 2. *Cont.*

Events	Rumor	Non-Rumor
Sydney siege	BREAKING: Live coverage of hostage situation unfolding in Sydney's Martin Place.	Any lone nutjob who wants international attention for a crime now just has to wave a black flag around. Voila #sydney
	Black Islamic flag being held up in window of #lindt chocolate store in Martin Place, Sydney - hostages inside.	If you have any lingering doubts that the threat of radical Islam is global as well as lethal ...

The RUMDECT [10] consists of data from Weibo and Twitter, although our experiments exclusively focused on the Weibo data. The Weibo dataset comprises 4664 labeled events collected from the Sina Weibo rumor debunking service, encompassing 2313 rumors and 2321 non-rumors. Each event in the Weibo dataset provides text information, related forwarding details, posting and reposting location information, user information of the poster and reposter, and other features. Table 3 presents examples of typical rumor and non-rumor Weibo posts from this dataset.

Table 3. Typical Weibo data from the RUMDECT dataset.

	Event Text	Reposter Text
Rumor	#每日一帖\#[生活百科][生活常识]牙膏底部的短线，绿色天然的。蓝色天然加药物，红色药物加化学，黑色纯化学。今后购买要记得好好看看了，真心有用！转！ Translation:#One post per day\#[Life Encyclopedia][common sense of life]The short line at the bottom of toothpaste: green means natural. Blue means natural with added chemicals. Red means chemicals with added medication. Black means pure chemicals. Remember to pay attention when buying in the future, it's really useful! Share!	真的假的啊[晕] Translation:Is it true or not! [Geez]
Non-rumor	什么叫做真正的街舞！全世界的街舞都被这帮人玩遍了 2.42秒彻底崩溃 这才是街舞啊 你懂得 http://t.cn/hbIn6n。 Translation:What is real street dance! They've mastered street dance from all over the world in just 2.42 seconds, completely mind-blowing. This is real street dance, you know. http://t.cn/hbIn6n.	好帅啊~~~~~~~~~~~~ Translation:So handsome~~~~~~~~~~~~

4.2. Features Used in Supervised Machine Learning Methods

Based on the information contained in the two datasets mentioned above, we conducted feature extraction on both the PHEME [10] and RUMORDECT [9] datasets. Specifically, the features listed in Table 4 were extracted from these datasets, along with their descriptions, to serve as input for the traditional machine learning models. Following the definitions outlined in Table 4, we created two new datasets that exclusively contained the feature data extracted from the original datasets. These datasets formed a crucial component of our subsequent research analysis.

Table 4. The features used for rumor detection in the experiment.

Dataset	Feature Name	Description
RUMDECT	Number of sentiment words	The number of positive and negative words in a post
	Number of URLs	The number of URLs in a post
	Number of comment count	The number of comments on a post
	Account type	The type of current account: personal or organization
	Registration age	The registration age of current account
	Count followers	Number of users following current account
	Number of posts	The number of posts by the account
	Number of repost	The number of total repost
	Count followees	Number of accounts which current account followed

Table 4. *Cont.*

Dataset	Feature Name	Description
PHEME	Length of characters	The number of all characters contained in a post
	Number of words	The number of words contained in a post
	Count uppercase letters	Fraction of capital letters in the tweet
	Sentiment Score	Sum of ±0.5 for weak positive/negative words, ±1.0 for strong ones
	Number of URLs	The number of URLs in a post
	Registration age	The registration age of current account
	Count followers	Number of users following current account
	Is verified	Is current account verified
	Statuses count	The number of tweets at posting time
	Sentiment positive words	The number of positive words in the text
	Sentiment negative words	The number of negative words in the text
	Contains multi mark	Contains a question mark '?'
	Contains pronoun first	Contains a personal pronoun in 1st person
	Contains pronoun second	Contains a personal pronoun in 2st person
	Contains pronoun third	Contains a personal pronoun in 3st person

4.3. Active Learning Tools and Machine Learning Tools

In order to study the impact of different active learning strategies on various supervised machine learning models, we employed the following mature active learning tools: ALiPy (https://github.com/NUAA-AL/ALiPy, accessed on 3 November 2023) (Active Learning in Python) [43] provides a module-based implementation of an active learning framework, enabling users to conveniently evaluate, compare, and analyze the performance of active learning methods. The library offers several commonly used strategies for instance selection, including Uncertainty, Query by Committee, and QUIRE, among others.

We experimentally compared the performance of different query strategies, including Uncertainty, Query By Committee (QBC), Expected Error Reduction (ERR), Graph Density, and Querying Informative and Representative Examples (QUIRE). For the uncertainty, we investigated three different uncertainty measures: least confidence, margin, and entropy. For QBC, bagging was used to create a committee and vote entropy was selected as the measure of disagreement of committees. When referring to these strategies in the legend of the experimental results, we use the abbreviations Unc(Lc) for uncertainty with least confidence, Unc(Ma) for uncertainty with margin, Unc(En) for uncertainty with entropy, GD for Graph Density, and DW for the information density weighted strategy.

For machine learning models, we utilized the corresponding implementations provided by the Scikit-learn library (https://scikit-learn.org/, accessed on 3 November 2023) [44].

All experiments were performed on an Intel (R) Xeon (R) Silver 4116 CPU @ 2.10 GHz with 48 CPUs and 128 GB of RAM.

4.4. Data Splitting and Initial Labeled Dataset

Each experiment followed a well-defined process. First, we performed random division of the dataset into three subsets: a test dataset comprising 30% of the data, an initial labeled dataset consisting of ten randomly selected data instances, and an unlabeled data pool constituting approximately 70% of the data.

Within each experiment, we conducted ten rounds, and each round involved the selection of ten labeled instances chosen at random. These labeled instances were used to train the initial model from scratch. Subsequently, based on different query strategies, additional unlabeled instances were selected from the unlabeled pool for annotation. Importantly, the instances in the unlabeled pool already had labels, allowing us to retrieve the label corresponding to each selected instance. These newly labeled instances were then appended to the labeled dataset and the model was retrained using the updated labeled dataset. This iterative process continued until the query budget was exhausted. The final results reported in our experiments are the average outcomes across all ten rounds.

Our experimental design incorporated a query budget, which was set to 200. This choice was informed by careful experimentation and empirical observations that suggested this budget as sufficient for ensuring that most models converge effectively. Unless otherwise specified, we maintained the remaining parameters of the query strategy in the active learning and machine learning models at their default values.

4.5. Metrics

After examining several metrics, including the accuracy, precision, recall, and F1 score, and with LR and SVM using the three uncertainty queries, we observed that the different metrics exhibited similar behaviors. Hence, for the sake of experimental simplicity, in this paper we employ the accuracy as the primary metric. For all comparisons, we report the averaged accuracy across ten runs using different random seeds. The accuracy was computed by relating FP (False Positives), FN (False Negatives), TP (True Positives), and TN (True Negatives), as defined in Equation (6).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{6}$$

5. Experimental Results and Discussion

5.1. Baselines

The primary objective of this paper is to assess the potential of active learning for achieving comparable performance to models trained on the entire dataset while using a significantly smaller amount of data. To establish a performance baseline, we employed a model trained on the complete training dataset.

In this case, we random partitioned the preprocessed dataset into a training dataset comprising 70% of the whole dataset and a test dataset representing the remaining 30%. The machine learning models in our experiments underwent initial training using the training dataset and subsequent evaluation on the test data. The reported results are an average of the outcomes from ten rounds of experimentation.

The results are reported in Table 5. Notably, we observed that RFC consistently demonstrated superior performance across both datasets. Machine learning models such as MLP, AdaBoost, and KNN exhibited higher performance compared to LR, DTC, and NB. This observation can be attributed to the constrained flexibility of the LR, DTC, and NB models, which tend to establish a linear decision boundary between the two classes. This inherent constraint can impede their ability to effectively distinguish between rumors and non-rumors.

Table 5. Baseline accuracy score, best-performing accuracy score, and corresponding strategy name among all model and strategy combinations.

Model	PHEME			RUMDECT		
	Baseline	Best Accuracy	Best Strategy	Baseline	Best Accuracy	Best Strategy
LR	0.725	0.726 ± 0.01	ERR	0.889	0.864 ± 0.01	Unc (Lc)
LR (SGD)	0.755	0.676 ± 0.03	QBC	0.889	0.856 ± 0.01	QBC
NB	0.407	0.553 ± 0.08	Unc (Ma)	0.876	0.851 ± 0.01	QUIRE
GP	0.726	0.710 ± 0.01	Unc (Lc)	0.892	0.865 ± 0.02	QBC
DTC (Gini)	0.679	0.690 ± 0.01	ERR	0.874	0.859 ± 0.02	QBC
DTC (Entropy)	0.677	0.709 ± 0.02	ERR	0.872	0.860 ± 0.02	QBC
LDA	0.771	0.673 ± 0.03	DW	0.892	0.875 ± 0.02	QBC
QDA	0.589	0.627 ± 0.10	DW	0.881	0.876 ± 0.02	DW
Ada	0.736	0.677 ± 0.02	QBC	0.903	0.883 ± 0.01	QBC
KNN	0.771	0.706 ± 0.01	QBC	0.895	0.849 ± 0.01	QBC
MLP	0.778	0.706 ± 0.01	Unc (Lc)	0.904	0.875 ± 0.01	Unc (Lc)
RFC	0.805	0.719 ± 0.01	Unc (Lc)	0.915	0.902 ± 0.01	Unc (Ma)
SVM (RBF)	0.778	0.714 ± 0.01	Unc (En)	0.895	0.884 ± 0.02	QUIRE
SVM (Linear)	0.775	0.719 ± 0.03	QBC	0.895	0.879 ± 0.02	QBC
SVM (SGD)	0.765	0.685 ± 0.03	QBC	0.894	0.854 ± 0.03	QBC

5.2. Results

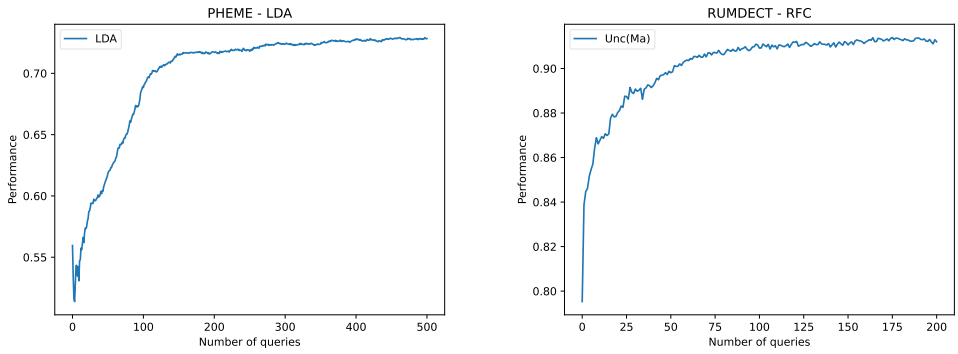
In the following, we delve into and discuss the comparative results of our experiments in detail from multiple perspectives.

5.2.1. Analysis of the Effectiveness of Active Learning in Rumor Detection

Table 5 provides all of the baseline accuracy scores for the experimental models along with the best-performing accuracy scores among all query strategies.

For the PHEME dataset, several models using active learning achieved performance surpassing that of the baseline. Even LDA, the model with the lowest performance among them, reached 87.3% of the baseline performance. Figure 3a illustrates that its performance can be further improved to 0.741 with the addition of query datapoints, reaching up to 96% of the baseline performance. This demonstrates that by carefully selecting an optimal number of query instances, machine learning models can be trained effectively with a smaller dataset compared to the entire training set while achieving the desired performance levels.

For the RUMDECT dataset, all models using active learning exceeded the baseline performance achieved by the models trained using the entire dataset, achieving a performance level of 95%. Likewise, several models, including DTC (Gini), DTC (Entropy), GP, LDA, and Ada, outperformed the baseline. Under appropriate query strategies, these models can converge quickly relative to baseline performance. For example, as shown in the Figure 3b, the RFC model uses the Unc(Margin) strategy to quickly converge within just 100 query instances.



(**a**) (**b**)

Figure 3. (**a**) On the PHEME dataset, the DW strategy achieved the best performance for the LDA model; however, its performance was the worst among all methods. As the query instances continue to increase, its performance approaches the baseline performance when the total number reaches 500. (**b**) The RFC model, which exhibited the best baseline performance on the RUMDECT dataset, demonstrates the ability to approach baseline performance with remarkable efficiency when utilizing the Unc(Ma) query strategy, converging in just 100 query instance.

In summary, the results in Table 5 demonstrate that all models can approach or achieve baseline performance using no more than 200 data instances when paired with appropriate active learning query strategies. These data instances represent only 5% and 6% of the training sets for the PHEME and RUDECT datasets, respectively. This highlights the effectiveness of active learning in significantly reducing labeling costs while maintaining high-performance results in the context of rumor detection.

5.2.2. Model Comparison

Figures 4 and 5 display the accuracy scores of various machine learning models using different strategies on the PHEME and RUMDECT datasets, respectively. These diagrams demonstrate that most models reach a performance plateau with just 50–150 query instances, showcasing their ability to achieve commendable accuracy with a relatively small dataset size. Notably, most models exhibit consistent behavior and substantial improvements

when provided with 50–150 well-selected data samples through active learning. Overall, the performance on the PHEME dataset lags behind that on the RUMDECT dataset, possibly due to the higher level of noise and class imbalance present in the PHEME dataset.

Examining the results further, RFC yields the highest accuracy score in baseline and best convergence score in active learning among our models on both datasets. Although different query strategies may result in varying performance, they all demonstrate consistent convergence for RFC, which is true for models such as Ada, LR, GP, SVM(RBF) and SVM(Linear) as well.

Several models, including LDA, MLP, QDA, and models trained by SGD. such as LR(SGD) and SVM(SGD), exhibited noticeable oscillations across various query strategies. In essence, these models experience fluctuations in their performance as the number of samples increases. Notably, the LR(SGD) and SVM(SGD) models display oscillations throughout the entire sampling process. This behavior may arise from a mismatch between the query strategies and the fundamental principle of random sampling behind SGD-based training. To mitigate this issue, it is possible to adopt query strategies that leverage gradient information, such as the Expected Gradient Length (EGL) strategy [45].

In the case of the LDA model applied to the PHEME dataset, the model's performance initially decreases during the early stages of training, then increases, and eventually converges. However, this phenomenon does not manifest on the RUMDECT dataset. We attribute this difference to the class imbalance in the PHEME dataset and the relatively small initial labeled dataset used for model training. To address this, we increased the number of initial datapoints in the PHEME dataset to 25, resulting in the disappearance of the fluctuations, as depicted in the Figure 6a. A similar trend was observed for the QDA models, which involve a greater number of model parameters. For both LDA and QDA models, active learning necessitates a larger labeled dataset to prevent fluctuations.

In the case of the MLP model used with active learning, an interesting observation emerges as the query budget nears exhaustion, with a significant decrease in performance becoming evident. In situations where the sampling budget is increased to 500, as depicted in Figure 6b, two such performance dips are apparent. These dips do not disappear when changing the number of hidden units. We leave the investigation of the causes of this phenomenon to future work, along with the exploration of potential solutions.

In the case of the Decision Tree Classifier (DTC) model, there is minimal variation between the Gini and Entropy criteria across all query strategies applied to the two datasets. However, it is noteworthy that an intriguing pattern emerges for the query strategies Unc(Lc), Unc(Ma), and Unc(En) as the number of samples increases, where the performance of the DTC model starts to decline. The DTC model relies on a diverse set of representative data points to construct effective classification rules; however, the uncertainty query strategy primarily refines existing rules rather than introducing new ones, which may explain the performance decline. A similar phenomenon was observed with the KNN model on the PHEME dataset. Therefore, these three query strategies should not be chosen for DTC or KNN models. The QBC query strategy, which considers the disagreement of multiple classifiers and tends to select query instances for introducing new rules, is applicable for DTC(Gini), DTC(Entorpy), and KNN models.

Notably, NB experienced a continuous decrease in performance when trained on the PHEME dataset, and showed only a minimal improvement on RUMDECT. This suggests that query strategies may not adequately address this model's need for diverse datapoints, highlighting an area of focus for active learning in generative models.

These findings can shed light on the complex behaviors of different models under active learning conditions, and emphasize the importance of understanding the interplay between model complexity, dataset characteristics, and query strategies.

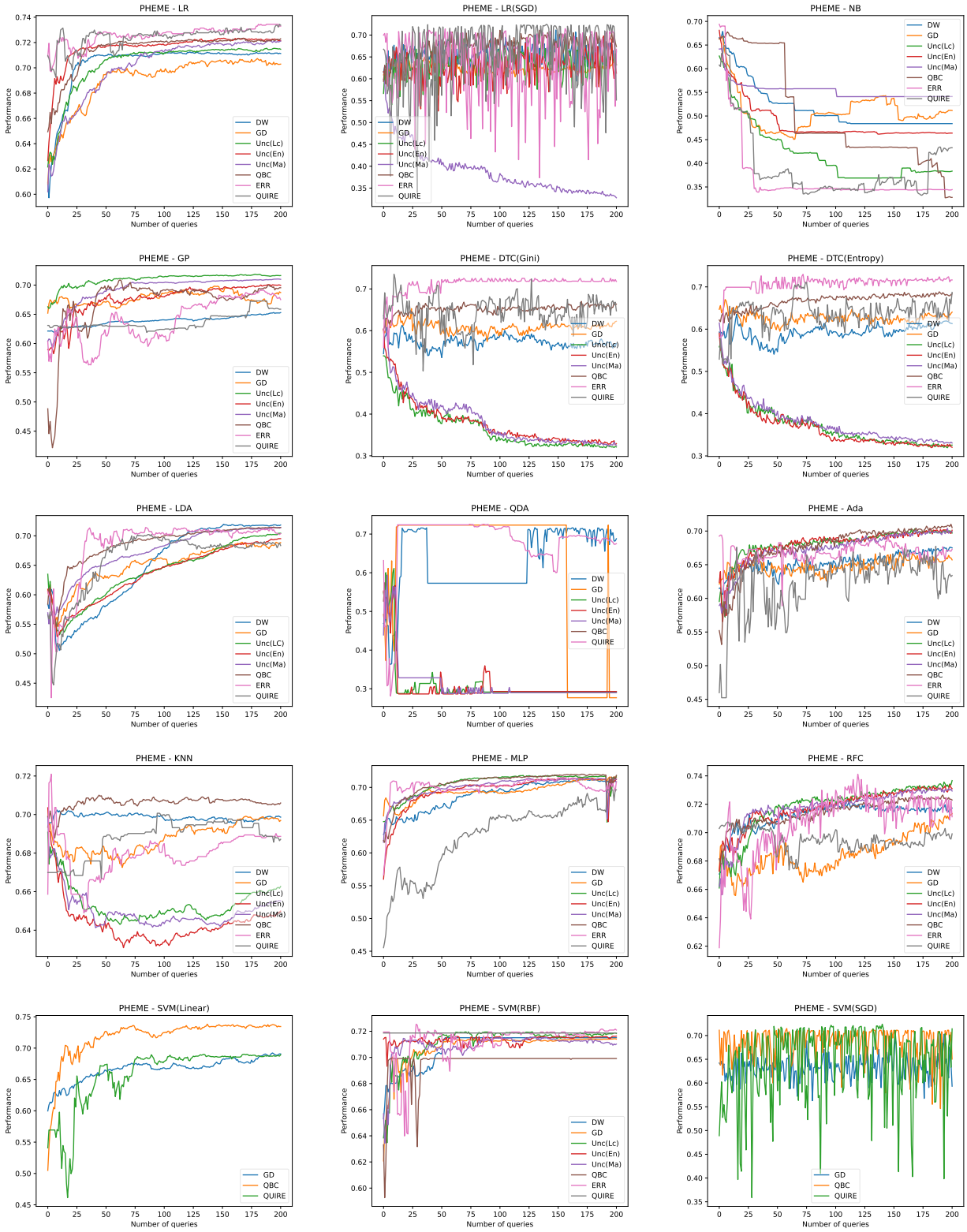


Figure 4. Performance of different strategies used with various machine learning models on the PHEME dataset. The vertical axis shows the accuracy and the horizontal axis shows the number of data samples used for training during active learning. Subfigures without legends share the same legend as the first subfigure in the same row.

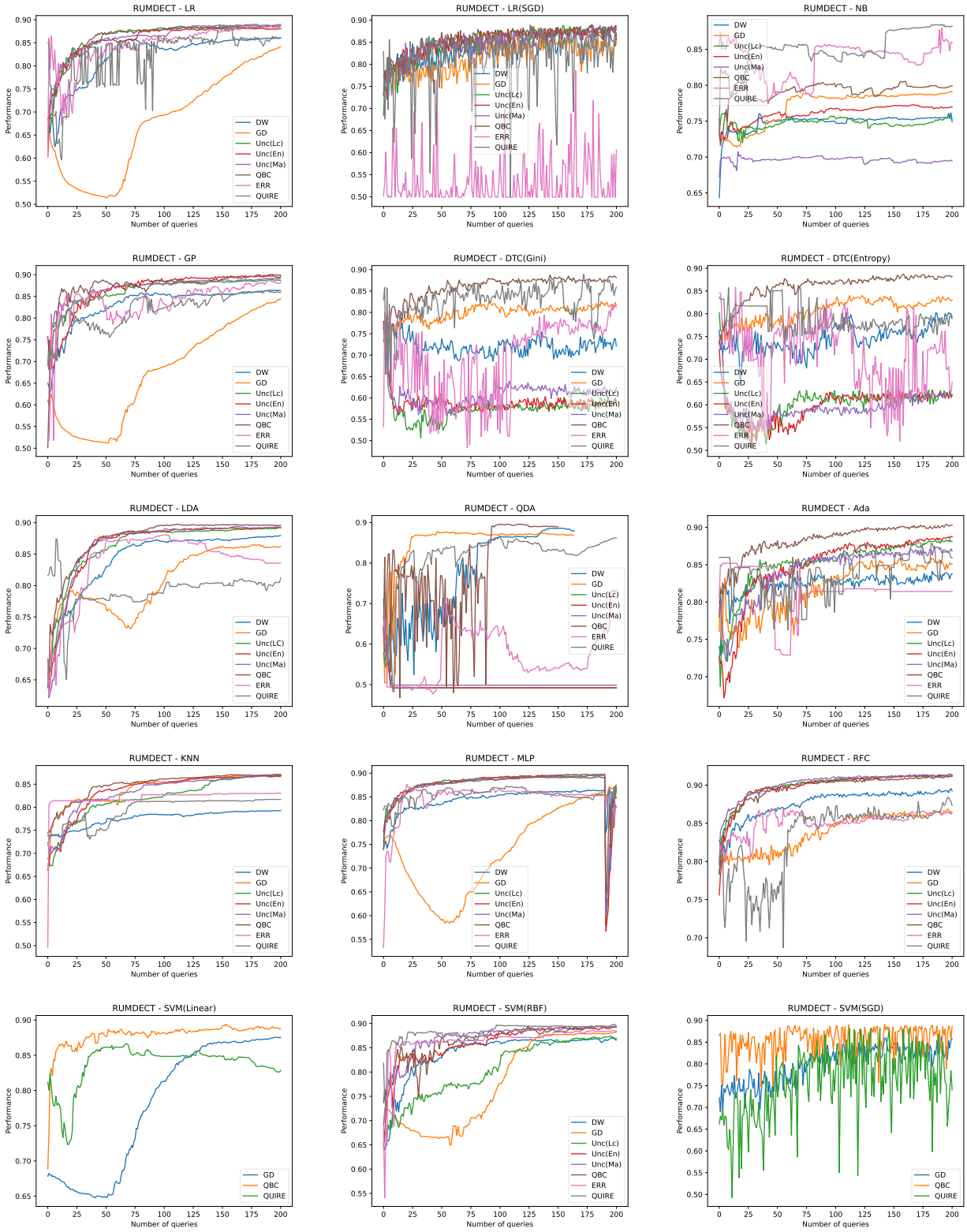
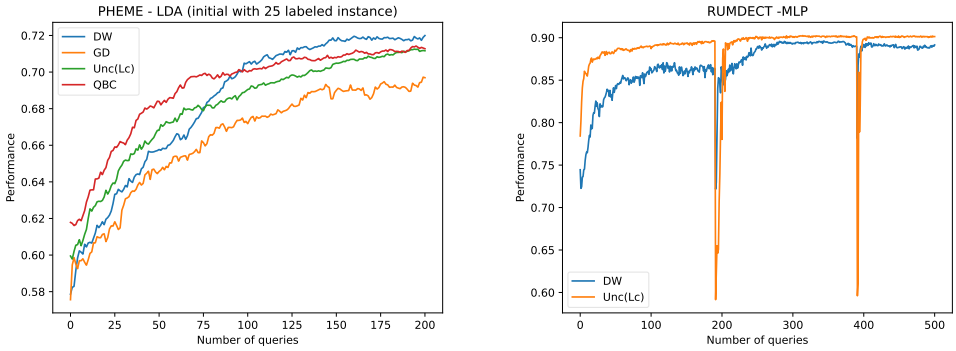


Figure 5. Performance of different strategies used with various machine learning models on the RUMDECT dataset. The meanings of the vertical axis and horizontal axis are the same as in Figure 4. The horizontal gray dashed line represents the model's baseline performance. Subfigures without legends share the same legend as the first subfigure in the same row.



(**a**) (**b**)

Figure 6. (**a**) After increasing the initial amount of label data to 25, the LDA model does not have fluctuations. (**b**) After the MLP model increases the query budget to 500, two fluctuations appear.

5.2.3. Analysis of Query Strategies

In this subsection, we conduct a comprehensive comparative analysis of various active learning query strategies with the goal of gaining deeper insights into their respective advantages and disadvantages. This analysis aims to shed light on rumor detection for enhancing active learning query strategies in future research.

On one hand, information-based query strategies have proven to be highly effective, especially for discriminative models characterized by linear classification boundaries. On the other hand, representative-based query strategies are better suited for models that rely on data diversity, such as DTC models. Interestingly, despite having the advantages of both characteristics, the combination of information-based and representative-based strategies does not yield superior performance.

The uncertainty-based strategies perform well across most models on both datasets, and show no significant differences in performance. Based on the experimental results in Figures 4 and 5, these strategies are highly effective for discriminative models characterized by linear boundaries. However, they rely on estimating prediction uncertainties, and consequently cannot be applied to non-probabilistic models such as SVM(Linear) and SVM(SGD). Additionally, they tend to select non-representative instances in the feature space, such as outliers or noisy instances;as a result, their performance is lower on the noisy and imbalanced PHEME dataset compared to the RUMDECT dataset. Furthermore, because these strategies may exhibit a bias towards regions in the feature space where the model already has high confidence, they might inadvertently neglect crucial areas that require further exploration. As a result, a performance degradation phenomenon is observed for the DTC(Gini) and DTC(Entropy) models on both datasets, as well as for the KNN model on the PHEME dataset.

Figures 4 and 5 illustrate that the QBC strategy consistently achieves the best performance across multiple models on the two datasets. Notably, even in challenging scenarios such as the DTC(Gini), DTC(Entropy), and KNN model on the PHEME dataset, the QBC strategy outperforms most other strategies. This outcome aligns with expectations, as QBC tends to select instances in which ensemble models exhibit the highest levels of disagreement, thereby promoting diversity in the selected instances. QBC is characterized by its stable performance and rapid convergence, making it adaptable to a wide range of machine learning models.

In contrast, the ERR strategy is model-dependent and performs suboptimally when the model cannot provide reliable uncertainty estimates. Even in situations where uncertainty estimates are available, ERR tends to perform worse than QBC. This observation might be attributed to the fact that ERR is better suited for complex models dealing with high-dimensional data.

The GD strategy, as a representative-based query strategy, does not consistently outperform the other strategies on either dataset. Its convergence is relatively slow, and its perfor-

mance appears to depend on the dataset's characteristics. For instance, on the RUMDECT dataset, multiple models (including LR, GP, LDA, and MLP) using the GD strategy initially experienced a performance decrease, followed afterwards by an increase. However, this phenomenon did not manifest on the PHEME dataset.

Regarding DW and QUIRE, which use a combination of the informative and representative strategies, our experiments indicate that they do not significantly outperform purely informative or representative approaches.

In terms of running time, we can consider the example of LR and RFC running for ten rounds on two datasets. Table 6 provides a comparison of their running times. It is evident that the running times for the ERR and QUIRE strategies are significantly longer than those for the other strategies. This is primarily because QUIRE involves measuring both informativeness and representativeness for each unlabeled instance, making it more computationally intensive and time consuming, especially for larger datasets. During our experiments, we observed that running a model with the ERR or QUIRE strategy under the hardware configuration used in this study took nearly a week. Such long processing times could make these strategies impractical for real-world rumor detection tasks. In contrast, while the running time for the QBC strategy is longer than for most other strategies (excluding ERR and QUIRE), it falls within an acceptable range.

When considering both performance and running time, QBC emerges as the most suitable query strategy for most machine learning models in rumor detection.

Table 6. Running times (in seconds) of LR and RFC on the PHEME and RUMDECT datasets when employing different query strategies.

Model	Dataset	Unc (Lc)	Unc (Ma)	Unc (En)	QBC	ERR	GD	QUIRE	DW
LR	PHEME	360	214	352	600	125,347	358	547,705	484
	RUMDECT	381	239	370	417	176,304	380	625,944	845
RFC	PHEME	783	847	813	3458	684,065	463	544,903	752
	RUMDECT	553	511	558	4042	465,379	523	763,072	987

6. Conclusions

In this paper, we have conducted a comprehensive experimental comparative analysis assessing various rumor detection models in conjunction with diverse active learning query strategies. Our experimental results indicate that most supervised machine learning models can achieve model training with significantly fewer datapoints than the full training set, and that their performance can match or even surpass models trained on the complete dataset. Different supervised learning models exhibit varying performance under different active learning query strategies. This article presents experimental findings that identify the query strategies that enable different machine learning models to converge most rapidly. Our experimental results illustrate that RFC achieves the best performance on both datasets, while QBC emerges as the most suitable query strategy for most machine learning models in rumor detection. In addition to comparing model performance, this article discusses the runtimes of different strategies in order to further assess the pros and cons of different machine learning models when using various query strategies. This work can provide experimental guidance for the application of active learning in the field of rumor detection.

As part of our future work, we plan to expand upon our current research by incorporating deep neural networks such as Recurrent Neural Networks (RNNs), Convolutional Neural Networks (CNNs), and Graph Neural Networks (GNNs) for rumor detection. Additionally, we aim to explore the application of active learning in rumor detection within dynamic data streaming scenarios.

Author Contributions: Conceptualization, F.Y. and L.S.; methodology, F.Y. and H.H.; software, H.L.; validation, F.Y., H.L. and H.H.; formal analysis, F.Y.; investigation, L.S.; resources, H.L.; data curation, F.Y. and H.L.; writing—original draft preparation, F.Y.; writing—review and editing, F.Y., L.S. and

H.H.; visualization, H.L.; supervision, F.Y.; project administration, F.Y.; funding acquisition, F.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the 'Guangdong Province Overseas Renowned Teacher' project of department of science and technology of Guangdong province and the Zhongshan City Social Welfare and Basic Research Project "Research on High-Precision Mark Point Localization Algorithm Based on Convolutional Neural Networks and Transfer Learning" (No. 2022B2009).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All data are contained within the article and the referenced papers.

Acknowledgments: We express our gratitude to Liang Shangsong from Sun Yat-sen University for offering invaluable guidance and support throughout our thesis research. Additionally, we extend our appreciation to Li Gang from Deakin University for providing extensive constructive feedback and valuable suggestions during the paper's writing process.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

AL	Active Learning
SGD	Stochastic Gradient Descent
RBF	Radial Basis Function
LR	Logistic Regression classifier
LR(SGD)	Logistic Regression classifier trained with SGD
SVM	Support Vector Machine
SVM(Linear)	SVM classifier with a Linear kernel
SVM(RBF)	SVM classifier with an RBF kernel
SVM(SGD)	SVM(RBF) trained with SGD
DTC	Decision Tree Classifier
DTC(Gini)	Decision Tree Classifier using Gini criterion
DTC(Entropy)	Decision Tree Classifier using Entropy criterion
NB	Gaussian Naive Bayes
RFC	Random Forest Classifier
KNN	K-Nearest Neighbours
GP	Gaussian Process classifier
MLP	Multi-layer Perceptron
LDA	Linear Discriminant Analysis
QDA	Quadratic Discriminant Analysis
Ada	AdaBoost classifier
Unc(Lc)	Query strategy: uncertainty measured by least confidence criterion
Unc(Ma)	Query strategy: uncertainty measured by margin criterion
Unc(Lc)	Query strategy: uncertainty measured by entropy criterion
QBC	Query strategy: query by committee
ERR	Query strategy: expected error reduction strategy
GD	Query strategy: graph density strategy
QUIRE	Query strategy: querying informative and representative examples
DW	Query strategy: information density weighted strategy

References

1. Shu, K.; Sliva, A.; Wang S.; Tang J.; Liu H. Fake news detection on social media: A data mining perspective. *SIGKDD Explor. Newsl.* **2017**, *19*, 22–36. [CrossRef]
2. Walker, M.; Matsa, K.E. *News Consumption across Social Media in 2021*; Technical Report; Pew Research Center: Washington, DC, USA, 2021.

3. Liang, G.; He, W.; Xu, C.; Chen L.; Zeng, J. Rumor identification in microblogging systems based on users' behavior. *IEEE Trans. Comput. Soc. Syst.* **2015**, *2*, 99–108. [CrossRef]
4. Vosoughi, S.; Roy, D.; Aral, S. The spread of true and false news online. *Science* **2018**, *359*, 1146–1151. [CrossRef] [PubMed]
5. Zhao, Z.; Resnick P.; Mei, Q. Enquiring minds: Early detection of rumors in social media from enquiry posts. In Proceedings of the 24th International Conference on World Wide Web, Florence, Italy, 18–22 May 2015; pp. 1395–1405.
6. Friggeri, A.; Adamic L.; Eckles, D.; Cheng, J. Rumor cascades. In Proceedings of the Eighth International Conference on Weblogs and Social Media, ICWSM, Ann Arbor, MI, USA, 1–4 June 2014; The AAAI Press: Washington, DC, USA, 2014.
7. Jin, Z.; Cao, J.; Guo, H.; Zhang, Y.; Wang, Y.; Luo, J. Detection and analysis of 2016 us presidential election related rumors on twitter. In *Social, Cultural, and Behavioral Modeling: 10th International Conference, SBP-BRiMS 2017, Washington, DC, USA, 5–8 July 2017, Proceedings 10*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 14–24.
8. Islam, M.S.; Sarkar, T.; Khan, S.H.; Kamal, A.H.M.; Hasan, S.M.; Kabir, A.; Dalia, Y.; Islam, M.A.; Chowdhury, K.I.A.; Anwar, K.S.; et al. COVID-19–related infodemic and its impact on public health: A global social media analysis. *Am. J. Trop. Med. Hyg.* **2020**, *103*, 1621. [CrossRef] [PubMed]
9. Castillo, C.; Mendoza, M.; Poblete, B. Information credibility on twitter. In Proceedings of the 20th International Conference on World Wide Web, Hyderabad, India, 28 March–1 April 2011; pp. 675–684.
10. Ma, J.; Gao, W.; Mitra, P.; Kwon, S.; Jansen, B.J.; Wong, K.F.; Cha, M. Detecting rumors from microblogs with recurrent neural networks. In Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, New York, NY, USA, 9–15 July 2016; pp. 3818–3824.
11. Bian, T.; Xiao, X.; Xu, T.; Zhao, P.; Huang, W.; Rong, Y.; Huang, J. Rumor detection on social media with bi-directional graph convolutional networks. In Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; pp. 549–556.
12. Sun, M.; Zhang, X.; Zheng, J.; Ma, G. DDGCN: Dual dynamic graph convolutional networks for rumor detection on social media. In Proceedings of the Thirty-Sixth AAAI Conference on Artificial Intelligence, Virtual Event, 22 February–1 March 2022; pp. 4611–4619.
13. Zubiaga, A.; Liakata, M.; Procter, R.; Wong Sak Hoi, G.; Tolmie, P. Analysing how people orient to and spread rumours in social media by looking at conversational threads. *PLoS ONE* **2016**, *11*, e0150989. [CrossRef]
14. Karisani, P.; Karisani, N. Semi-supervised text classification via self-pretraining. In Proceedings of the Fourteenth ACM International Conference on Web Search and Data Mining, Virtual Event, Israel, 8–12 March 2021; pp. 40–48.
15. Guo, B.; Ding, Y.; Yao, L.; Liang, Y.; Yu, Z. The future of false information detection on social media: New perspectives and trends. *ACM Comput. Surv.* **2020**, *53*, 1–36. [CrossRef]
16. Nguyen, T.T.; Nguyen, Q.V.H.; Nguyen, D.T.; Nguyen, D.T.; Huynh-The, T.; Saeid, N.; Nguyen, T.T.; Quoc-Viet, P.; Nguyen, C.M. Deep learning for deepfakes creation and detection: A survey. *Comput. Vis. Image Underst.* **2022**, *223*, 103525. [CrossRef]
17. Deng, J.; Dong, W.; Socher, R.; Li, LJ.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), Miami, FL, USA, 20–25 June 2009; pp. 248–255.
18. Kwon, S.; Cha, M.; Jung, K.; Chen, W.; Wang, Y. Prominent features of rumor propagation in online social media. In Proceedings of the 2013 IEEE 13th International Conference on Data Mining ICDM 2013, Dallas, TX, USA, 7–10 December 2013; pp. 1103–1108.
19. Zubiaga, A.; Liakata, M.; Procter, R. Learning reporting dynamics during breaking news for rumour detection in social media. *arXiv* **2016**, arXiv:1610.07363.
20. Christina, B.; Symeon, P.; Yiannis, K.; Steve, S.; Nic, N. Challenges of computational verification in social multimedia. In Proceedings of the 23rd International World Wide Web Conference, Seoul, Republic of Korea, 7–11 April 2014; pp. 743–748.
21. Liu, X.; Nourbakhsh, A.; Li, Q.; Fang, R.; Shah, S. Real-time rumor debunking on twitter. In Proceedings of the 24th ACM International Conference on Information and Knowledge Management, Melbourne, VIC, Australia, 19–23 October 2015; pp. 1867–1870.
22. Ma, J.; Gao, W.; Wong, K.F. Detect rumors in microblog posts using propagation structure via kernel learning. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Vancouver, BC, Canada, 30 July– 4 August 2017; pp. 708–717.
23. Jin, Z.; Cao, J.; Guo, H.; Zhang, Y.; Luo, J. Multimodal fusion with recurrent neural networks for rumor detection on microblogs. In Proceedings of the 2017 ACM on Multimedia Conference, Mountain View, CA, USA, 23–27 October 2017; pp. 795–816.
24. Jia, R.; Dao, D.; Wang, B.; Hubis, F.A.; Gurel, N.M.; Li, B.; Zhang, C.; Spanos, C.J.; Song, D. Efficient task-specific data valuation for nearest neighbor algorithms. *Proc. VLDB Endow.* **2019**, *12*, 1610–1623. [CrossRef]
25. Farinneya, P.; Pour, M.M.A.; Hamidian, S.; Diab, M. Active learning for rumor identification on social media. In *Findings of the Association for Computational Linguistics: EMNLP 2021*; Association for Computational Linguistics: Punta Cana, Dominican Republic, 2021; pp. 4556–4565.
26. Settles, B. *Active Learning Literature Survey*; Technical Report 1648; University of Wisconsin-Madison: Madison, WI, USA, 2009.
27. Varshney, D.; Vishwakarma, D.K. A review on rumour prediction and veracity assessment in online social network. *Expert Syst. Appl.* **2021**, *168*, 114208. [CrossRef]

28. Qazvinian, V.; Rosengren, E.; Radev, D.; Mei, Q. Rumor has it: Identifying misinformation in microblogs. In Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, Edinburgh, UK, 27–31 July 2011; John McIntyre Conference Centre: Edinburgh, UK, 2011; pp. 1589–1599.
29. Karisani, P.; Karisani, N.; Xiong, L. Multi-view active learning for short text classification in user-generated data. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2022, Abu Dhabi, United Arab Emirates, 7–11 December 2022; pp. 6441–6453.
30. Naseem, U.; Khushi, M.; Khan, S.K.; Shaukat, K.; Moni, M.A. A comparative analysis of active learning for biomedical text mining. *Appl. Syst. Innov.* **2021**, *4*, 23. [CrossRef]
31. Wu, M.; Li, C.; Yao, Z. Deep active learning for computer vision tasks: Methodologies, applications, and challenges. *Appl. Sci.* **2022**, *12*, 8103. [CrossRef]
32. McCallum, A.; Nigam, K. Employing EM and pool-based active learning for text classification. In Proceedings of the Fifteenth International Conference on Machine Learning (ICML 1998), Madison, WI, USA, 24–27 July 1998; pp. 350–358.
33. Siddhant, A.; Lipton, Z.C. Deep bayesian active learning for natural language processing: Results of a large-scale empirical study. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; pp. 2904–2909.
34. Bhattacharjee, S.D.; Talukder, A.; Balantrapu, B.V. Active learning based news veracity detection with feature weighting and deep-shallow fusion. In Proceedings of the 2017 IEEE International Conference on Big Data (IEEE BigData 2017), Boston, MA, USA, 11–14 December 2017; pp. 556–565.
35. Bhattacharjee, S.D.; Tolone, W.J.; Paranjape, V.S. Identifying malicious social media contents using multi-view context-aware active learning. *Future Gener. Comput. Syst.* **2019**, *100*, 365–379. [CrossRef]
36. Hasan, M.S.; Alam, R.; Adnan, M.A. Truth or lie: Pre-emptive detection of fake news in different languages through entropy-based active learning and multi-model neural ensemble. In Proceedings of the IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, The Hague, The Netherlands, 7–10 December 2020; pp. 55–59.
37. Sahan, M.; Smidl, V.; Marik, R. Active learning for text classification and fake news detection. In Proceedings of the International Symposium on Computer Science and Intelligent Control, Rome, Italy, 12–14 November 2021; pp. 87–94.
38. Fedorov, V. Optimal experimental design. *Wiley Interdiscip. Rev. Comput. Stat.* **2010**, *2*, 581–589. [CrossRef]
39. Roy, N.; McCallum, A. Toward optimal active learning through sampling estimation of error reduction. In Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001), Williamstown, MA, USA, 28 June–1 July 2001; pp. 441–448.
40. Ebert, S.; Fritz, M.; Schiele, B. RALF: A reinforced active learning formulation for object class recognition. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition CVPR 2012, Providence, RI, USA, 16–21 June 2012; pp. 3626–3633.
41. Huang, S.J.; Jin, R.; Zhou, Z.H. Active learning by querying informative and representative examples. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *36*, 1936–1949. [CrossRef] [PubMed]
42. Settles, B. Curious Machines: Active Learning with Structured Instances. Ph.D. Thesis, University of Wisconsin–Madison, Madison, WI, USA, 2008.
43. Tang, Y.P.; Li, G.X.; Huang, S.J. Alipy: Active learning in python. *arXiv* **2019**, arXiv:1901.03802.
44. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
45. Settles, B.; Craven, M.; Ray, S. Multiple-instance active learning. *Adv. Neural Inf. Process. Syst.* **2007**, *20*.