Relabeling A Minimal Training Subset to Flip a Prediction

Anonymous ACL submission

Abstract

When facing an unsatisfactory prediction from a machine learning model, it is crucial to investigate the underlying reasons and explore the potential for reversing the outcome. We ask: To flip the prediction on a test point x_t, how to identify the smallest training subset $\mathcal{S}_t$ we need to **relabel**? We propose an efficient procedure to identify and relabel such a subset via an extended influence function. We find that relabeling fewer than 2% of the training points can always flip a prediction. This mechanism can serve multiple purposes: (1) providing an approach to challenge a model prediction by altering training points; (2) evaluating model robustness with the cardinality of the subset (i.e., $|\mathcal{S}_t|$); we show that $|\mathcal{S}_t|$ is highly related to the noise ratio in the training set and $|\mathcal{S}_t|$ is correlated with but complementary to predicted probabilities; (3) revealing training points lead to group attribution bias. To the best of our knowledge, we are the first to investigate identifying and relabeling the minimal training subset required to flip a given prediction. [1]

1 Introduction

The interpretability of machine learning systems is a crucial research area as it aids in understanding model behavior, facilitating debugging, and enhancing performance (Adebayo et al., 2020; Han et al., 2020; Pezeshkpour et al., 2022; Teso et al., 2021; Marx et al., 2019). A common approach involves analyzing the model's predictions by tracing back to the training data (Hampel, 1974; Cook and Weisberg, 1980, 1982). Particularly, when a machine learning model produces an undesirable result, users might be interested in identifying the training points to modify to overturn the outcome. If the identified training points are wrongly labeled, the related determination should be overturned. For instance, consider a scenario where a machine learning model evaluates research papers and gives decisions. If an author receives a rejection and disagrees with the result, they might request insight into the specific papers examples used to train the model. If it turns out that correcting a few mislabeled training examples can change the prediction, then the original decision might need reconsideration, possibly accepting the paper instead. This concept is referred to contesting the predictions made by automatic models (Hirsch et al., 2017; Vaccaro et al., 2019). When using such models, users should have the right and ability to question and challenge results, especially when these results impact them directly (Almada, 2019). Our research is geared towards offering a mechanism for users to challenge these predictions by tracing back to the training data.

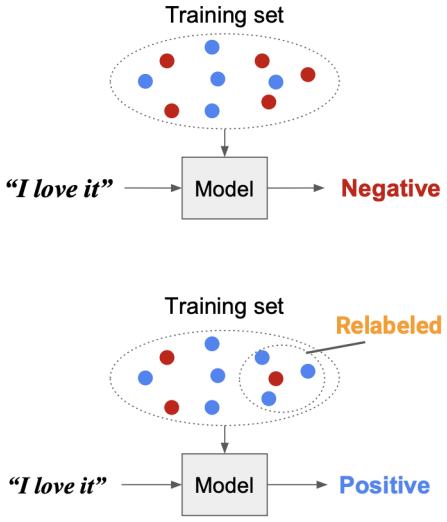

Figure 1: The question we seek to answer is: which is the smallest subset of the training data that needs to be relabeled in order to flip a specific prediction from the model?

In this paper, we study the question (visualized in Figure 1): *Given a test point x_t and its associated predicted label $\hat{y}_t$ by a model, how can we find the minimal training subset $\mathcal{S}_t$, if relabeled before*

[1] Code and data to reproduce all experiments will be available on the GitHub.

Test point			$\lvert\mathcal{S}_t\rvert$	Training points in $\mathcal{S}_t$		
Text	**Label**	**Prediction**		**Text**	**True Label**	**Labeled as**
The people who can stop it are the ones who pay their wages.	Non-hate	Hate	1	*Worker.*	Non-hate	Hate
We will never forget their heroism.	Non-hate	Hate	1	*TRUTH NO LIE.*	Non-hate	Hate
Cossack: There are no Russians with hook nose and dark hair in Russia, except for the jews.	Non-hate	Hate	2	*All of Ukraine is Russia/Poland/Romania, you fake Ukrainians.*	Hate	Non-hate
				The rest is Tatar.	Non-hate	Hate

Table 1: Examples showcase misclassified test points alongside the identified training set $\mathcal{S}_t$. For each test point, if those training points are relabeled prior to training, the test point can be correctly classified. These training points are intentional noise we manually introduced into the dataset.

training, would lead to a different prediction? [2]

Identifying $\mathcal{S}_t$ by enumerating all possible subsets of training examples, re-training under each, and then observing the resultant prediction would be inefficient and impractical. We thus introduce an algorithm for finding such sets efficiently using the extended *influence function*, which allow us to approximate changes in predictions expected as a result of relabeling subsets of training data (Koh et al., 2019; Warnecke et al., 2021; Kong et al., 2021).

The identified subset $\mathcal{S}_t$ can be harnessed for a variety of downstream applications. Firstly, we discover that $|\mathcal{S}_t|$ can be less than 5%, suggesting that relabeling a small fraction of the training data can markedly influence the test prediction. Secondly, we observe a correlation between $|\mathcal{S}_t|$ and the noise ratio in the training set. As the noise ratio increases from 0 to 0.5, $|\mathcal{S}_t|$ tends to decrease obviously. Thirdly, we find that $|\mathcal{S}_t|$ can be small when the model is high confident in a test prediction, so $|\mathcal{S}_t|$ serve as a measure of robustness that complements to the predicted probability. Lastly, our approach can light on points containing group attribution bias that caused biased determinations. We demonstrate that when such bias exists in the training set, the corresponding $\mathcal{S}_t$ will significantly overlap with the biased training data.

The contributions of this work are summarized as follows. (1) We introduce the problem: identifying the minimal subset $\mathcal{S}_t$ of training data, if relabeled, would result in a different prediction on test point x_t.; (2) We provide a computationally efficient algorithm for this task and report performance in binary classification problems; (3) We demonstrate that the size of the subset ($|\mathcal{S}_t|$) can be used to assess the robustness of the model and the training set. (4) We show that the composition of $\mathcal{S}_t$ can explain group attribution bias.

[2]We provide a way to investigate the training points instead of retraining the model.

2 Methods

This section first demonstrates the algorithm to find the minimal relabel set and shows a case to use the algorithm to challenge the model's prediction.

2.1 Algorithm

Consider a binary classification problem with a training set denoted as $Z^{\text{tr}} = \{z_1, ..., z_N\}$. Each data point $z_i = (x_i, y_i)$ comprises features $x_i \in \mathcal{X}$ and a label $y_i \in \mathcal{Y}$. We train a classification model $f_w : \mathcal{X} \to \mathcal{Y}$ by minimizing the empirical risk, which yields the estimated parameter $\hat{w}$, as defined by: $\hat{w} := \text{argmin}_w \mathcal{R}(w) = \text{argmin}_w \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}(z_i, w) + \frac{\lambda}{2} w^T w$. Here, λ serves as the hyperparameter for regularization. We assume that $\mathcal{R}$ is twice-differentiable and strongly convex in w, with $H_{\hat{w}} := \nabla_w^2 R(\hat{w}) = \frac{1}{N}\sum_{i=1}^{N} \nabla_w^2 \mathcal{L}(z_i, \hat{w}) + \lambda I$. Suppose we relabel a subset of training points $\mathcal{S} \subset Z^{\text{tr}}$ by relabeling y_i to y_i' for $(x_i, y_i) \in \mathcal{S}$ and re-estimate w, yielding new parameters $\hat{w}_{\mathcal{S}}$:

$$\hat{w}_S = \text{argmin}_w \{R(w) + \frac{1}{N} \sum_{(x_i, y_i) \in S} l\}, \quad (1)$$

where $l = -\mathcal{L}(x_i, y_i, w) + \mathcal{L}(x_i, y_i', w)$.

Due to the large number of possible subsets in the training set, it is computationally impractical to relabel and retrain models for each subset to observe prediction changes. Warnecke et al. (2021); Kong et al. (2021) derived the influence exerted by relabeling a training set $\mathcal{S}$ on the *loss* incurred for a test point t as:

$$\nabla_w \mathcal{L}(z_t, \hat{w})^\intercal \Delta_i w, \tag{2}$$

where $\Delta_i w = \frac{1}{N} H_{\hat{w}}^{-1} \sum_{(x_i,y_i)\in S} \nabla_w l$ is the change of parameters after relabeling training points in $\mathcal{S}$. Instead, we estimate the influence on *predicted probability* result by relabeling training subset $\mathcal{S}$ as:

$$\Delta_t f := \nabla_w f_{\hat{w}}(x_t)^\intercal \Delta_i w, \tag{3}$$

which is named as **IP-relabel**. Based on this metric and adopt the algorithm proposed by Broderick et al. (2020); Yang et al. (2023), we propose the Algorithm 1 to find a training subset $\mathcal{S}_t$ to relabel, which would result in flipping the test prediction $\hat{y}_t$ on x_t. Our approach initiates by approximating the change in predicted probability $\Delta_t f$ for a test point x_t, which results from the relabeling of each training point. Subsequently, we iterate through all the training points in a descending order of their influence—starting with the most decisive to the least. During each iteration, we accumulate the change in predicted probability $\Delta_t f$. When the cumulative change causes the output $\hat{y}_t$ to cross a predefined threshold, the algorithm identifies $\mathcal{S}_t$. If, however, the output fails to cross the threshold even after examining the entire training set, the algorithm is unable to find the set $\mathcal{S}_t$.

2.2 Case Study

In this section, we present an example to demonstrate how our method can be used to challenge the predictions of machine learning models. We employ the Hate Speech dataset (de Gibert et al., 2018), which encompasses instances of hate communication that target specific groups based on characteristics such as race, color, ethnicity, etc. On social media platforms, users found engaging in hate speech are typically banned.

We implement a linear regression model to classify hate speech on the internet. We intentionally introduced noise into the training dataset by mislabeling 1,000 data points (out of 9632, switching labels from 1 to 0 and vice versa). This deliberate noise in the training set can result in additional misclassifications during testing.

As demonstrated in Table 1, for each test instance, Algorithm 1 pinpoints the specific training data points that, when relabeled before training, could change the prediction of the test point. The table showcases three instances where the model misclassified test points. The corresponding training sets, $\mathcal{S}_t$, consist of training points that closely resemble the test cases but were erroneously labeled. Given that the classifications can be altered by relabeling a small subset of mislabeled training data, determinations based on these classifications, such as banning users, warrant careful reconsideration.

Algorithm 1: An algorithm to find a minimal subset to flip a test prediction

Input: f: Model; Z^{tr}: Full training set; N: number of total training points; $Z^{\text{tr}'}$: Relabeled full training set; $\hat{w}$: Parameters estimated Z^{tr}; $\mathcal{L}$: Loss function; x_t: A test point; τ: Classification threshold (e.g., 0.5)

Output: $\mathcal{S}_t$: minimal train subset identified to flip the prediction ($\emptyset$ if unsuccessful)

$H \leftarrow \nabla_w^2 \mathcal{L}(Z^{\text{tr}}, \hat{w})$
$\nabla_w l \leftarrow -\nabla_w \mathcal{L}(Z^{\text{tr}}, w') + \nabla_w \mathcal{L}(Z^{\text{tr}'}, w')$
$\Delta w \leftarrow \frac{1}{N} H^{-1} \nabla_w l$
$\Delta_t f \leftarrow \nabla_w f_{\hat{w}}(x_t)^\intercal \Delta w$
$\hat{y}_t \leftarrow f(x_t) > \tau$ `// Binary prediction`
`// Sort instances (and estimated output differences) in order of the current prediction`
`direction` $\leftarrow \{\uparrow$ if $\hat{y}_t$ else $\downarrow\}$
`indices` $\leftarrow$ `argsort`$(\Delta_t f,$ `direction`$)$
$\Delta_t f \leftarrow$ `sort`$(\Delta_t f,$ `direction`$)$
for $k = 1 \ldots |Z^{tr}|$ **do**
  $\hat{y}_t' = (f(x_t) + \text{sum}(\Delta_t f[:k])) > \tau$
  if $\hat{y}_t' \neq \hat{y}_t$ **then**
    return Z^{tr}[`indices`$[:k]$]
return $\emptyset$

3 Experiments

We provide an overview of our experiments:

1. We introduce our experimental setup and then validate Algorithm 1 in Sec 3.1 and 3.2. Our

Dataset	Features	Found $\mathcal{S}_t$	Flip successful
Loan	BoW	61%	49%
Movie reviews	BoW	100%	72%
	BERT	100%	73%
Essays	BoW	77%	40%
	BERT	76%	39%
Hate speech	BoW	99%	87%
	BERT	99%	86%
Tweet sentiment	BoW	100%	75%
	BERT	100%	68%

Table 2: Percentages of text examples for which Algorithm 1 successfully identified a set $\mathcal{S}_t$ (center) and for which upon flipping these instances and retraining the prediction indeed flipped (right).

results confirm that we can effectively change the test predictions by relabeling revealed points and subsequent model retraining.

2. Sec 3.3 analyzes the magnitude of $|\mathcal{S}_t|$ across various datasets and models, emphasizing its correlation with predicted probability and noise ratio. This showcases its utility in analyzing the robustness of training points and models.

3. We further delve into the integration of subset $\mathcal{S}_t$ in Sec 3.4, demonstrating its potential to highlight biased training data.

4. In Sec 3.5, we compare our method against other methods to alter training points to flip test prediction, illustrating that our method revealed a smaller training subset.

3.1 Experimental Setting

Datasets. We use a tabular dataset: Loan default classification (Surana, 2021), and text datasets: Movie review sentiment (Socher et al., 2013); Essay grading (Foundation, 2010); Hate speech (de Gibert et al., 2018); and Twitter sentiment (Go et al., 2009) to evaluate our method.

Models. We consider the ℓ_2 regularized logistic regression to fit the assumption on influence function. As features, we consider both bag-of-words and neural embeddings induced via BERT (Devlin et al., 2018) for text datasets. We report basic statistics describing our datasets and model performance in Section A.1.

3.2 Algorithm Validation

How effective our algorithm find $\mathcal{S}_t$ and flip the corresponding prediction? As shown in Table 2, the frequency of finding $\mathcal{S}_t$ varies greatly among datasets. For the movie reviews and tweet datasets, Algorithm 1 returns a set $\mathcal{S}_t$ for approximately 100% of test points. On the other hand, for the simpler loan data, it only returns $\mathcal{S}_t$ for approximately 60% of instances. Results for other datasets fall between these two extremes. When the algorithm successfully finds a set $\mathcal{S}_t$, relabeling all $(x_i, y_i) \in \mathcal{S}_t$ almost enables the re-trained model to flip the prediction $\hat{y}_t$ (as indicated in the right-most column of Table 2).

Comparision with other methods. We draw comparisons between IP-relabel and several other methods (Pezeshkpour et al., 2021), including IP-remove (Yang et al., 2023), influence function (Koh and Liang, 2017), and three gradient-based instance attribution methods on a logistic regression model to the movie review dataset (Barshan et al., 2020; Charpiat et al., 2019):

1. $RIF = cos(H^{-\frac{1}{2}}\nabla_w \mathcal{L}(x_t), H^{-\frac{1}{2}}\nabla_w \mathcal{L}(x_i))$
2. $GD = \langle \nabla_w \mathcal{L}(x_t), \nabla_w \mathcal{L}(x_i) \rangle$
3. $GC = cos(\nabla_\theta \mathcal{L}(x_t), \nabla_\theta \mathcal{L}(x_i))$

We also randomly select subsets of training data and relabel them. We graph the average change in predicted probability for 100 randomly chosen test points in Figure 2. These probabilities are from the model trained before and after relabeling the top k training points ranked on the scores above. Our analysis indicates that IP-relabel shows a more significant impact in the test predicted probability compared to the impact of removing training points as ranked by other methods.

Running time of Algorithm 1. We recorded the average running time of Algorithm 1 to find $\mathcal{S}_t$ for test points in different datasets in Table 3 on Apple M1 Pro CPUs. For one test point, it just takes milliseconds to go through the whole training set (the training set sizes are provided in A.1) to find $\mathcal{S}_t$.

Dataset	BoW (ms)	BERT (ms)
Movie Reviews	19.04	140.51
Essays	160.01	265.09
Hate speech	103.70	299.46
Tweet	58.42	260.75
Loan	63.97	/

Table 3: Average running time (in milliseconds) of Algorithm 1 to find $\mathcal{S}_t$ for a test point in different datasets.

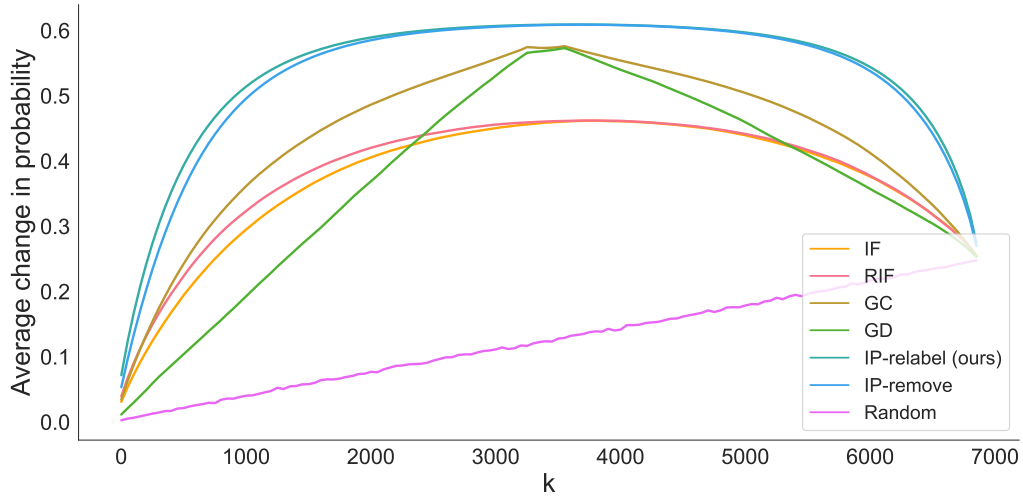

Figure 2: The relationship between the average of absolute difference on predicted probabilities for sampled test points results from relabeled $k = |\mathcal{S}_t|$ training points, using different methods on movie review dataset.

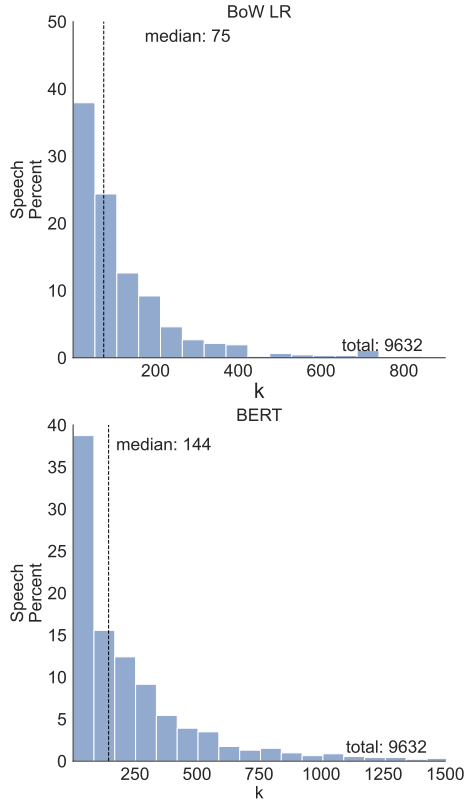

Figure 3: The histogram shows the distribution of $k = |\mathcal{S}_t|$ on the hate speech dataset, i.e. the minimal number of points that need to be relabeled from the training data to change the prediction $\hat{y}_t$ of a specific test example x_t.

3.3 $|\mathcal{S}_t|$ Quantifies Model Robustness

Relabel less than 2% training data can usually flip a prediction. The empirical distributions of k values for subsets $\mathcal{S}_t$ identified by Algorithm 1 can be seen in Figure 3 for the representative hate speech datasets (full results are in the Appendix). The key observation is that when $\mathcal{S}_t$ is found, its size is often relatively small compared to the total number of training instances. In fact, for many test points, relabeling less than 2% instances would have resulted in a flipped prediction.

BERT demonstrates greater robustness than LR based on $|\mathcal{S}_t|$ measures. For a proficiently trained model, the need to relabel a larger subset of training data in order to alter a correct test prediction suggests greater model robustness. In Figure 4, we present a comparison of the average values of $|\mathcal{S}_t|$ for common test data points where both BERT and LR model predictions were successfully altered using our method. The results indicate that BERT typically demands the relabeling of more training data points than the LR models do. This observation supports the utility of our method in gauging the relative robustness of different models.

Correlation between k and the predicted probability. Does the size of $\mathcal{S}_t$ tell us anything beyond what we might infer from the predicted probability $p(y_t = 1)$? In Fig 5 we show a scatter of $k = |\mathcal{S}_t|$ against the distance of the predicted probability from 0.5 on speech dataset. There are test instances of the model being confident, but relabeling a small set of training instances would overturn the prediction. In Sec A.3, there are datasets where the k can be highly correlated with probability.

How is $|\mathcal{S}_t|$ correlated with the noise ratio? Figure 6 shows how $|\mathcal{S}_t|$ and the model's accuracy vary when we increase the noise ratio from 0 to 0.9. We

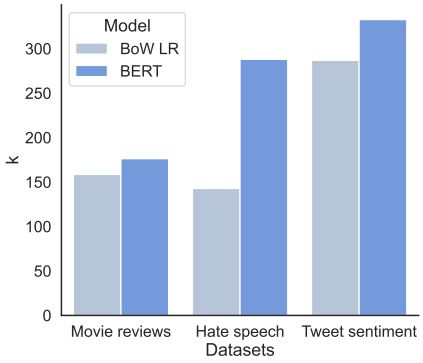

Figure 4: Comparison of the average $k = |\mathcal{S}_t|$ values for shared test points under both BERT and LR models that were successfully flipped by our method.

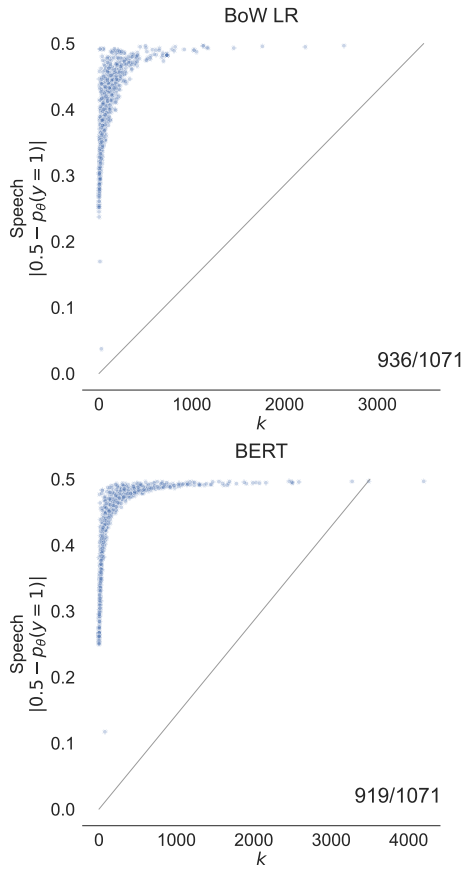

Figure 5: The correlation between the predicted probabilities of certain test examples and $k = |\mathcal{S}_t|$ on the hate speech dataset. For test examples where the model is highly certain about its prediction, the prediction can be flipped by relabeling a small number of data points from the training set.

introduce noise to the training set by incrementally relabeling a portion of training points, from 0 to 0.9 in steps of 0.1. When the noise ratio increases from 0 to 0.5, we observe a decline in $|\mathcal{S}_t|$. However, as the noise ratio rises from 0.5 to 0.9, $|\mathcal{S}_t|$ starts to increase. Interestingly, within the noise ratio interval of 0 to 0.3, the model's accuracy does not demonstrate a noticeable decline. This suggests that $|\mathcal{S}_t|$ can be an additional metric for assessing the model's robustness complementary to accuracy under different noise ratios.

3.4 Composition of $\mathcal{S}_t$ Contributes Bias Explanation

Group attribution bias in machine learning refers to a model's inclination to link specific attributes to a particular group, potentially resulting in biased predictions. We show that the integration of $\mathcal{S}_t$ is associated with group attribution biased in training data. As a case, we manually introduce group attribution bias into the loan default dataset (Surana, 2021), designed to predict potential defaulters for a consumer loan product. We augment a dataset containing basic consumer features with a manually added discrete "tag" feature, arbitrarily assigning 40% as "tag X" and 60% as "tag Y" We then introduce bias by relabeling 90% of the qualified "tag X" as "default." This biased set is defined as $\mathcal{B}$, where the wrong label tightly links with the feature "tag X." A logistic regression model is subsequently trained with this modified dataset.

We apply Algorithm 1 to misclassified test points and compute the proportion in each resulting subset $\mathcal{S}_t$ belonging to $\mathcal{B}$. The average proportions are 60% for "tag X" and 23% for "tag Y" misclassified data. The higher proportion in "tag X" suggests that the misclassification of eligible "tag X" individuals mainly results from the biased training set $\mathcal{B}$, whereas for "tag Y" individuals may be due to other reasons like model oversimplification. Thus, our approach can highlight training points contributing to group attribution bias.

3.5 Comparison between Removal and Relabeling

In this section, we compare two ways to alter training points such that the alternation can result in the flipping of a test point: relabeling and removal. We show that the relabeling mechanism can reveal a smaller training subset, thus saving the cost of investigating suspicious training points.

Kong et al. (2021) firstly propose an algorithm to find the training subset to remove to flip a test prediction for economy models, which we denote as "Removal Alg1" in Table 7. Yang et al. (2023) employ the same algorithm on machine learning

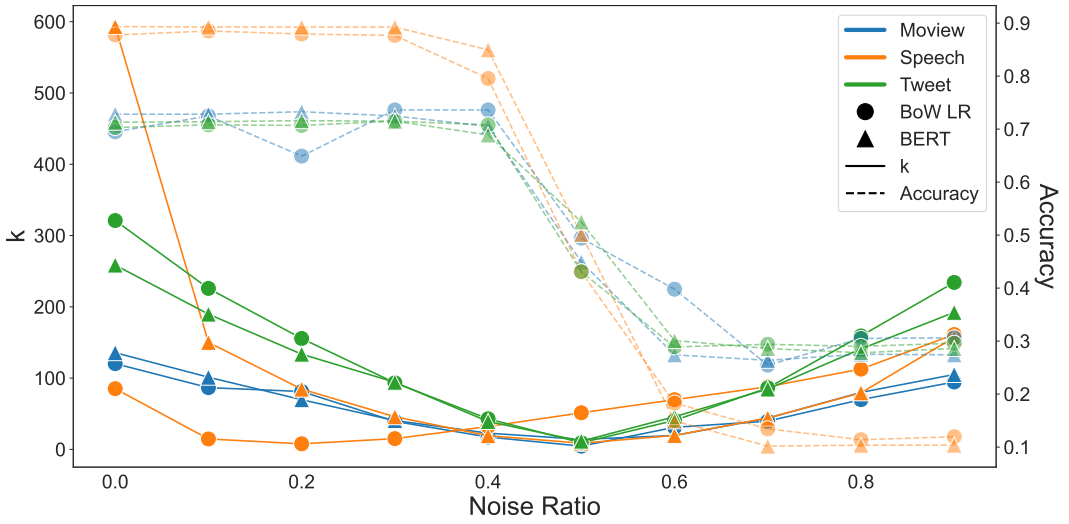

Figure 6: Average of $k = |\mathcal{S}_t|$ (solid line) and model's accuracy (dashed line) for the test dataset with noise ratio from 0 to 0.9. When the noise ratio increases from 0 to 0.3, k decreases apparently, while the model's accuracy does not demonstrate a noticeable decline.

	Noisy points in $\mathcal{S}_{t1}$			Normal points in $\mathcal{S}_{t2}$		
	Loan	**Movie reviews**	**Speech**	**Loan**	**Movie reviews**	**Speech**
Removal Alg1	47.9	1.8	146.8	30.6	2.1	31.9
Removal Alg2	45.6	1.8	104.2	27.0	2.1	21.0
Relabeling (ours)	**11.6**	**0.8**	**55.8**	**22.9**	**1.3**	**8.2**

Table 4: Average number of points to relabel and remove to flip a test prediction, categorized by noisy and normal points. Relabeling consistently leads to smaller sets of both noisy and normal points being altered.

models and improve it to return a smaller training set, denoted as "Removal Alg2".

We aim to show that when noise is present in the training set, the relabeling mechanism consistently uncovers a smaller subset of influential points from the noisy training set while affecting fewer standard points. To demonstrate this, we introduced a 30% noise factor into the training set by flipping labels of normal points, denoted as $\mathcal{N}$, which increased misclassified test points. We identified the training set $\mathcal{S}_t$ using the three methods for these misclassified test points. We divided the identified training points $\mathcal{S}_t$ into two categories: training points belonging to the noise set $\mathcal{S}_{t1} = \mathcal{S}_t \cap \mathcal{N}$, and those that do not belong to the noise set $\mathcal{S}_{t2} = \mathcal{S}_t \setminus \mathcal{N}$. The results presented in Table 4 demonstrate that both the S_1 and S_2 subsets identified through the relabeling process are smaller than those identified through removal. This suggests that considering relabeling training points can more effectively discern fewer noisy and regular training points, saving the cost to investigate more suspicious points. We also show the conclusion holds when there is no noise in the training set in Sec A.2.

4 Related Work

The holding of model predictions. Several studies have explored the changes of a model behavior and its factors. Ilyas et al. (2022) analyzed model behavior changes based on different training data. Harzli et al. (2022) studied the change of a specific prediction by finding a smallest informative feature set to analize economy models. Additionally, research on *counterfactual examples* aims to explain predicted outcomes by identifying the feature values that caused the given prediction (Kaushik et al., 2019). Recent studies investigated the influence function in machine learning to answer the question of "How many and which training points need to be removed to alter a specific prediction?" (Broderick et al., 2020; Yang et al., 2023). We follow these two works and propose an alternative way to alter the training points by asking, "How many and which training points would need to be relabeled

to change this prediction?"

Trustworthy machine learning is important in today's era, given the pervasive adoption of artificial intelligence systems in our everyday lives. Previous work emphasizes contestability as a key facet of trustworthiness, advocating for individuals' right to challenge AI predictions (Vaccaro et al., 2019; Almada, 2019). This may involve providing evidence or alternative perspectives to challenge AI-derived conclusions (Hirsch et al., 2017). Our mechanism offers a way to draw upon training data as evidence when contest AI determination. In line with advancing model fairness, it's crucial to address training data related to noise (Wang et al., 2018; Kuznetsova et al., 2020) and biases (Osoba and Welser IV, 2017; Howard and Borenstein, 2018). Our research shows that, despite different noise ratios, the model's accuracy remains relatively consistent, yet there is a significant variation in the size of the subset $\mathcal{S}_t$. Furthermore, we demonstrate that in scenarios where group attribution bias is present, our method can aid in identifying the associated training points.

Influence function offers tools for identifying training data most responsible for a particular test prediction (Hampel, 1974; Cook and Weisberg, 1980, 1982). By uncovering mislabeled training points and/or outliers, influence can be used to debug training data and provide insight for the result generated by neural networks (Koh and Liang, 2017; Adebayo et al., 2020; Han et al., 2020; Pezeshkpour et al., 2022; Teso et al., 2021). Warnecke et al. (2021) extend influence function to measure the influence of alternation in training points' feature and label and apply it to machine unlearning. Furthermore, Kong et al. (2021) also extended influence on the effect of relabeling training points but utilized this measure to identify and recycle noisy training samples, leading to enhanced model performance at the training stage. Our research emphasizes utilizing this measure to determine which training subsets should be relabeled to question machine learning model predictions, and we delve into the factors influencing the integration and size of the identified subsets.

5 Discussion and Future Work

In today's landscape dominated by large language models (LLMs), researchers are trying to integrate machine learning models into various decision-making processes, ranging from medical diagnoses (Shaib et al., 2023) to legal judgments (Jiang and Yang, 2023) and academic paper reviews (Liang et al., 2023). However, LLMs are black and hard to explain despite their immense capabilities. They are prone to challenges including, but not limited to, social biases (Hutchinson et al., 2020; Bender et al., 2021; Abid et al., 2021; Weidinger et al., 2021; Bommasani et al., 2022) and the spread of misinformation (Evans et al., 2021; Lin et al., 2022). These immediate issues might be precursors to more profound, long-term risks for making decisions based on AI systems.

As we harness these models to make critical decisions, it becomes imperative to delve into the root causes of any erroneous determinations. As outlined in our research, our proposed method offers a pathway to trace the origins of such errors back to specific training data points. As the first to state this problem, we primarily focus on linear regression and BERT with a classifier. In the future, we envision our methodology applying to even more complex models. A recent study extends the influence function to LLMs to understand how training data alterations can impact model predictions (Grosse et al., 2023). Building upon this foundation, adapting our approach for LLMs is promising for future exploration.

6 Conclusions

In this work, we introduce the problem of identifying a minimal subset of training data, $\mathcal{S}_t$, which, if relabeled before training, would result in a different test prediction. We introduce a computationally efficient algorithm to address this task and evaluate its performance within binary classification problems. In the experiment, we illustrate that the size of the subset $|\mathcal{S}_t|$ can serve as a measure of the model and the training set's robustness. Lastly, we indicate that the composition of $\mathcal{S}_t$ can reveal training points that cause group attribution bias.

7 Limitations and Risks

In our study, we've extensively used influence functions to solve the problem. However, being aware of fundamental limitations is crucial: they tend to be only effective in convex loss. The overarching goal of pinpointing a minimal subset within the training data, such that a change in labels leads to a reversal in prediction, isn't exclusively achievable via approximations rooted in influence functions.

This approach is favored in our work due to its intuitive nature and wide use. In addition, while Algorithm 1 currently shows less than optimal performance on the essay dataset, this presents an opportunity for further investigation. Specific characteristics unique to this dataset might influence the performance, opening up a valuable avenue for future research.

There exists an inherent risk wherein the same approach could be exploited to engender biased determinations. Specifically, by intentionally mislabeling genuine training data and subsequently retraining the model, actors with malicious intent might be able to invert just determinations, thereby compromising the model's integrity and fairness. To counteract this risk, strategies such as regular data integrity checks, stringent access control, and employing model robustness techniques can be integrated, thereby ensuring the preservation of model authenticity and shielding against adversarial exploits.

References

Abubakar Abid, Maheen Farooqi, and James Zou. 2021. Persistent anti-muslim bias in large language models. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pages 298–306.

Julius Adebayo, Michael Muelly, Ilaria Liccardi, and Been Kim. 2020. Debugging tests for model explanations. *arXiv preprint arXiv:2011.05429*.

Marco Almada. 2019. Human intervention in automated decision-making: Toward the construction of contestable systems. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Law*, pages 2–11.

Elnaz Barshan, Marc-Etienne Brunet, and Gintare Karolina Dziugaite. 2020. Relatif: Identifying explanatory training samples via relative influence. In *International Conference on Artificial Intelligence and Statistics*, pages 1899–1909. PMLR.

Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big?. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 610–623.

Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dora Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, Omar Khattab, Pang Wei Koh, Mark Krass, Ranjay Krishna, Rohith Kuditipudi, Ananya Kumar, Faisal Ladhak, Mina Lee, Tony Lee, Jure Leskovec, Isabelle Levent, Xiang Lisa Li, Xuechen Li, Tengyu Ma, Ali Malik, Christopher D. Manning, Suvir Mirchandani, Eric Mitchell, Zanele Munyikwa, Suraj Nair, Avanika Narayan, Deepak Narayanan, Ben Newman, Allen Nie, Juan Carlos Niebles, Hamed Nilforoshan, Julian Nyarko, Giray Ogut, Laurel Orr, Isabel Papadimitriou, Joon Sung Park, Chris Piech, Eva Portelance, Christopher Potts, Aditi Raghunathan, Rob Reich, Hongyu Ren, Frieda Rong, Yusuf Roohani, Camilo Ruiz, Jack Ryan, Christopher Ré, Dorsa Sadigh, Shiori Sagawa, Keshav Santhanam, Andy Shih, Krishnan Srinivasan, Alex Tamkin, Rohan Taori, Armin W. Thomas, Florian Tramèr, Rose E. Wang, William Wang, Bohan Wu, Jiajun Wu, Yuhuai Wu, Sang Michael Xie, Michihiro Yasunaga, Jiaxuan You, Matei Zaharia, Michael Zhang, Tianyi Zhang, Xikun Zhang, Yuhui Zhang, Lucia Zheng, Kaitlyn Zhou, and Percy Liang. 2022. On the opportunities and risks of foundation models.

Tamara Broderick, Ryan Giordano, and Rachael Meager. 2020. An automatic finite-sample robustness metric: When can dropping a little data make a big difference? *arXiv preprint arXiv:2011.14999*.

Guillaume Charpiat, Nicolas Girard, Loris Felardos, and Yuliya Tarabalka. 2019. Input similarity from the neural network perspective. *Advances in Neural Information Processing Systems*, 32.

R Dennis Cook and Sanford Weisberg. 1980. Characterizations of an empirical influence function for detecting influential cases in regression. *Technometrics*, 22(4):495–508.

R Dennis Cook and Sanford Weisberg. 1982. *Residuals and influence in regression*. New York: Chapman and Hall.

Ona de Gibert, Naiara Perez, Aitor García-Pablos, and Montse Cuadros. 2018. Hate Speech Dataset from a White Supremacy Forum. In *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*, pages 11–20, Brussels, Belgium. Association for Computational Linguistics.

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

Owain Evans, Owen Cotton-Barratt, Lukas Finnveden, Adam Bales, Avital Balwit, Peter Wills, Luca Righetti, and William Saunders. 2021. Truthful ai: Developing and governing ai that does not lie.

Hewlett Foundation. 2010. The hewlett foundation: Automated essay scoring.

Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *CS224N project report, Stanford*, 1(12):2009.

Roger Grosse, Juhan Bae, Cem Anil, Nelson Elhage, Alex Tamkin, Amirhossein Tajdini, Benoit Steiner, Dustin Li, Esin Durmus, Ethan Perez, Evan Hubinger, Kamilė Lukošiūtė, Karina Nguyen, Nicholas Joseph, Sam McCandlish, Jared Kaplan, and Samuel R. Bowman. 2023. Studying large language model generalization with influence functions.

Frank R Hampel. 1974. The influence curve and its role in robust estimation. *Journal of the american statistical association*, 69(346):383–393.

Xiaochuang Han, Byron C Wallace, and Yulia Tsvetkov. 2020. Explaining black box predictions and unveiling data artifacts through influence functions. *arXiv preprint arXiv:2005.06676*.

Ouns El Harzli, Bernardo Cuenca Grau, and Ian Horrocks. 2022. Minimal explanations for neural network predictions. *arXiv preprint arXiv:2205.09901*.

Tad Hirsch, Kritzia Merced, Shrikanth Narayanan, Zac E Imel, and David C Atkins. 2017. Designing contestability: Interaction design, machine learning, and mental health. In *Proceedings of the 2017 Conference on Designing Interactive Systems*, pages 95–99.

Ayanna Howard and Jason Borenstein. 2018. The ugly truth about ourselves and our robot creations: the problem of bias and social inequity. *Science and engineering ethics*, 24(5):1521–1536.

Ben Hutchinson, Vinodkumar Prabhakaran, Emily Denton, Kellie Webster, Yu Zhong, and Stephen Denuyl. 2020. Social biases in nlp models as barriers for persons with disabilities.

Andrew Ilyas, Sung Min Park, Logan Engstrom, Guillaume Leclerc, and Aleksander Madry. 2022. Datamodels: Understanding predictions with data and data with predictions. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 9525–9587. PMLR.

Cong Jiang and Xiaolei Yang. 2023. Legal syllogism prompting: Teaching large language models for legal judgment prediction. In *Proceedings of the Nineteenth International Conference on Artificial Intelligence and Law*, pages 417–421.

Divyansh Kaushik, Eduard Hovy, and Zachary C Lipton. 2019. Learning the difference that makes a difference with counterfactually-augmented data. *arXiv preprint arXiv:1909.12434*.

Pang Wei Koh and Percy Liang. 2017. Understanding black-box predictions via influence functions. In *International conference on machine learning*, pages 1885–1894. PMLR.

Pang Wei W Koh, Kai-Siang Ang, Hubert Teo, and Percy S Liang. 2019. On the accuracy of influence functions for measuring group effects. *Advances in neural information processing systems*, 32.

Shuming Kong, Yanyan Shen, and Linpeng Huang. 2021. Resolving training biases via influence-based data relabeling. In *International Conference on Learning Representations*.

Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. 2020. The open images dataset v4. *International Journal of Computer Vision*, 128(7):1956–1981.

Weixin Liang, Yuhui Zhang, Hancheng Cao, Binglu Wang, Daisy Ding, Xinyu Yang, Kailas Vodrahalli, Siyu He, Daniel Smith, Yian Yin, Daniel McFarland, and James Zou. 2023. Can large language models provide useful feedback on research papers? a large-scale empirical analysis.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Truthfulqa: Measuring how models mimic human falsehoods.

Charles Marx, Richard Phillips, Sorelle Friedler, Carlos Scheidegger, and Suresh Venkatasubramanian. 2019. Disentangling influence: Using disentangled representations to audit model predictions. *Advances in Neural Information Processing Systems*, 32.

Osonde A Osoba and William Welser IV. 2017. *An intelligence in our image: The risks of bias and errors in artificial intelligence*. Rand Corporation.

Pouya Pezeshkpour, Sarthak Jain, Sameer Singh, and Byron Wallace. 2022. Combining feature and instance attribution to detect artifacts. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1934–1946, Dublin, Ireland. Association for Computational Linguistics.

Pouya Pezeshkpour, Sarthak Jain, Byron C Wallace, and Sameer Singh. 2021. An empirical comparison of instance attribution methods for nlp. *arXiv preprint arXiv:2104.04128*.

Chantal Shaib, Millicent L Li, Sebastian Joseph, Iain J Marshall, Junyi Jessy Li, and Byron C Wallace. 2023. Summarizing, simplifying, and synthesizing medical evidence using gpt-3 (with varying success). *arXiv preprint arXiv:2305.06299*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank.

In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Ssubham Surana. 2021. Loan prediction based on customer behavior.

Stefano Teso, Andrea Bontempelli, Fausto Giunchiglia, and Andrea Passerini. 2021. Interactive label cleaning with example-based explanations. *Advances in Neural Information Processing Systems*, 34:12966–12977.

Kristen Vaccaro, Karrie Karahalios, Deirdre K Mulligan, Daniel Kluttz, and Tad Hirsch. 2019. Contestability in algorithmic systems. In *Conference Companion Publication of the 2019 on Computer Supported Cooperative Work and Social Computing*, pages 523–527.

Fei Wang, Liren Chen, Cheng Li, Shiyao Huang, Yanjie Chen, Chen Qian, and Chen Change Loy. 2018. The devil of face recognition is in the noise. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 765–780.

Alexander Warnecke, Lukas Pirch, Christian Wressnegger, and Konrad Rieck. 2021. Machine unlearning of features and labels. *arXiv preprint arXiv:2108.11577*.

Laura Weidinger, John Mellor, Maribeth Rauh, Conor Griffin, Jonathan Uesato, Po-Sen Huang, Myra Cheng, Mia Glaese, Borja Balle, Atoosa Kasirzadeh, Zac Kenton, Sasha Brown, Will Hawkins, Tom Stepleton, Courtney Biles, Abeba Birhane, Julia Haas, Laura Rimell, Lisa Anne Hendricks, William Isaac, Sean Legassick, Geoffrey Irving, and Iason Gabriel. 2021. Ethical and social risks of harm from language models.

Jinghan Yang, Sarthak Jain, and Byron C Wallace. 2023. How many and which training points would need to be removed to flip this prediction? *arXiv preprint arXiv:2302.02169*.

A Appendix

A.1 Datasets and model details

We present basic statistics describing our text classification datasets in Table 5. We set the threshold for the hate speech data as 0.25 ($\tau = 0.25$) to maximize the F1 score on the training set. For other datasets, we set the threshold as 0.5. For reference, we also report the hyperparameters and predictive performance realized by the models considered on the test sets of datasets in Table 6.

A.2 Comparison between removal and relabeling on clean training set

When there is no noise in the training set, we run Removal Alg1, Removal Alg2, and Algorithm 1 to compare the average returned training set size in Table 7. It shows that considering training points to relabel can result in smaller training sets than removing them.

Dataset	# Train	# Test	% Pos
Loan	21120	2800	0.50
Movie reviews	6920	872	0.52
Essay	11678	1298	0.10
Hate speech	9632	1071	0.11
Tweet sentiment	18000	1000	0.50

Table 5: Dataset information.

Models	Accuracy	F1-score	AUC	l2
Loan				
LR	0.79	0.80	0.88	100
Movie reviews				
BoW	0.79	0.80	0.88	1000
BERT	0.82	0.83	0.91	500
Essay				
BoW	0.97	0.80	0.99	1
BERT	0.98	0.87	0.99	10
Hate speech				
BoW	0.87	0.40	0.81	10
BERT	0.89	0.63	0.88	10
Tweet sentiment				
BoW	0.70	0.70	0.75	500
BERT	0.75	0.76	0.84	1000

Table 6: The model performance under different datasets.

	Loan	Reviews	Speech
Removal Alg1	965.4	712.8	768.6
Removal Alg2	440.4	636.8	411.6
Relabeling (ours)	**67.0**	**138.5**	**49.3**

Table 7: The comparison of average on $k = |\mathcal{S}_t|$ values over a random subset of test points x_t, result by removal (Algorithm 1 and Algorithm 2 (Yang et al., 2023)) and relabel. Relabel always finds a smaller $\mathcal{S}_t$ compared with removal.

A.3 Full Plots

We present the distribution of $\mathcal{S}_t$ across various datasets in Tables 7 and 9. Additionally, the correlation between predicted probability and the size of $\mathcal{S}_t$, denoted by $|\mathcal{S}_t|$, for different datasets is showcased in Tables 8 and 10.

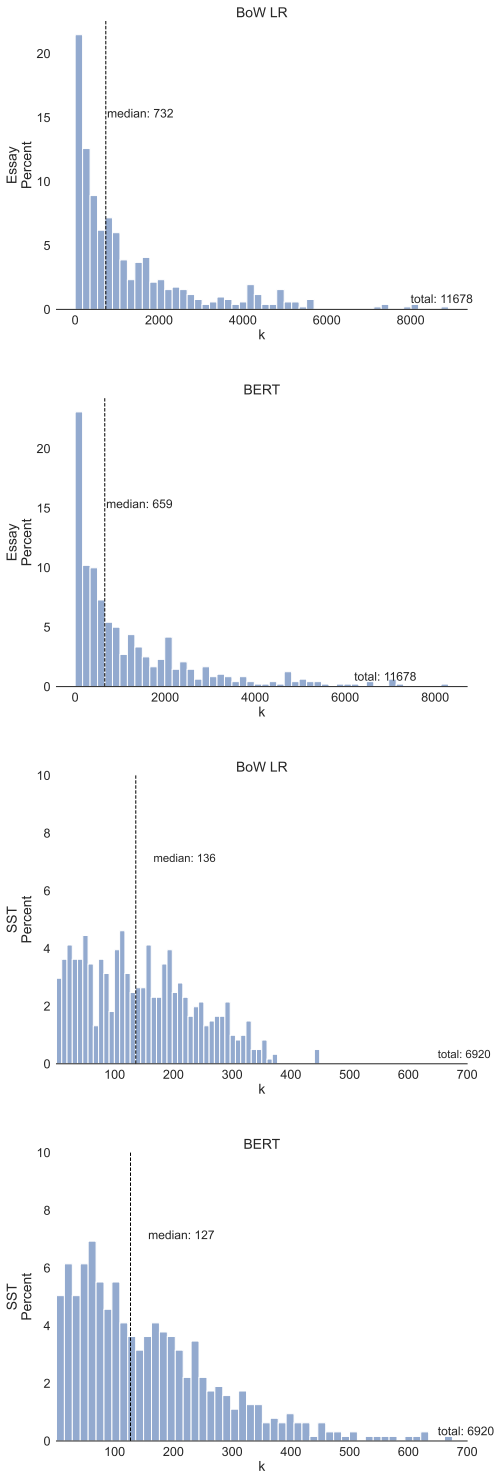

Figure 7: The histogram shows the distribution of $k = |\mathcal{S}_t|$, i.e. the number of points that need to be relabeled from the training data to change the prediction $\hat{y}_t$ of a specific test example x_t.

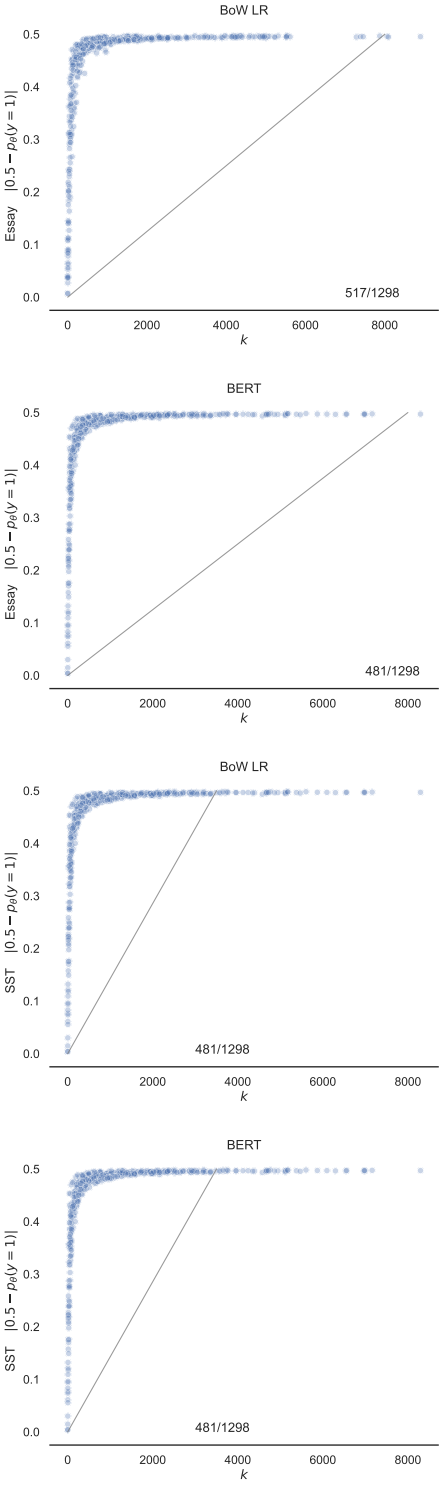

Figure 8: The plot displays the correlation between the predicted probabilities of certain test examples and $k = |\mathcal{S}_t|$ .There are some test examples where the model is reasonably or highly certain about its prediction, yet by removing a limited number of data points from the training set, the prediction can be altered.

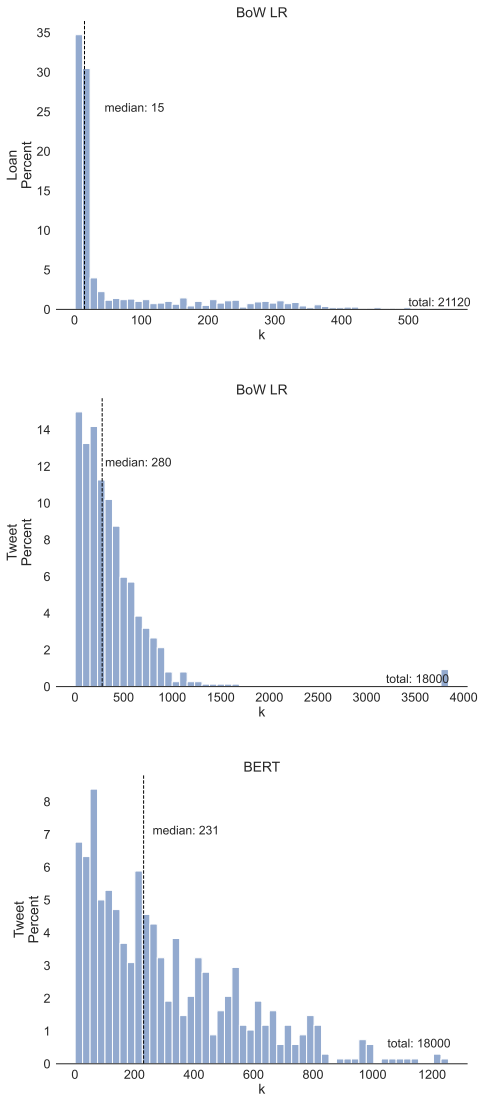

Figure 9: The histogram shows the distribution of $k = |\mathcal{S}_t|$, i.e. the number of points that need to be relabeled from the training data to change the prediction $\hat{y}_t$ of a specific test example x_t.

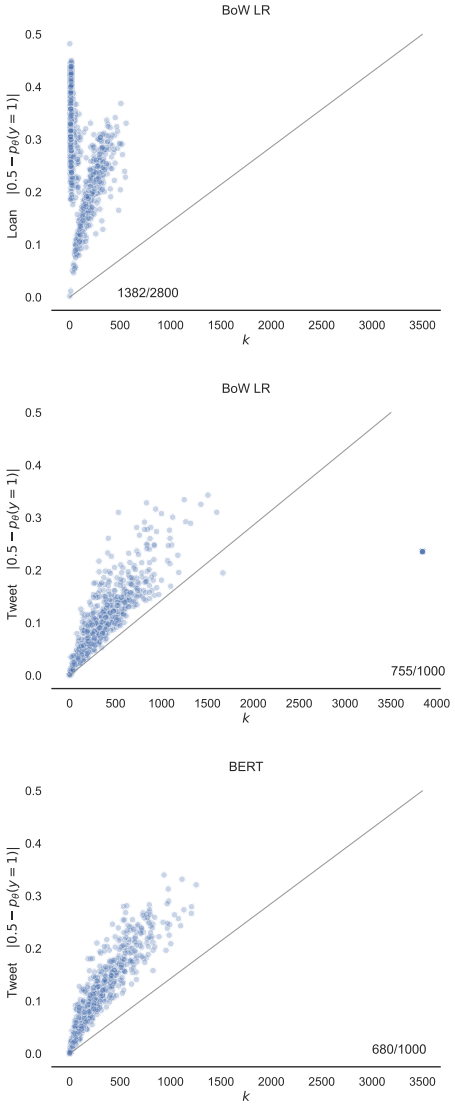

Figure 10: The plot displays the correlation between the predicted probabilities of certain test examples and $k = |\mathcal{S}_t|$. Thethere are some test examples where the model is reasonably or highly certain about its prediction, yet by removing a limited number of data points from the training set, the prediction can be altered.