Applying Iterative Nullspace Projection for Bias Mitigation in Toxicity Classification Task

Team 5:
20170906 Assem Zhunis, 20204880 Ern Chern Khor,
20170865 Bella Godiva, 20170900 Bauyrzhan Tokenov.

Abstract

Machine learning models can learn patterns in biased data and magnify the existing social inequality when they are used in real-world settings. In this project, we replicate Iterative Null-space Projection (INLP), a novel method for removing information from neural representations [9], for the purpose of bias mitigation in text classification. The method is based on repeated training of linear classifiers that predict a certain property, followed by projection of the representations on their null-space so that the classifiers become oblivious to that target property. We also improve the paper's results by applying the method on toxicity classification of text comments. The goal is to remove protected features of target property in the classifier, without sacrificing the accuracy of toxicity classification. We focused on three bias classes: gender bias, racial bias, and religious bias. The results show that the method can be successfully applied to gender and racial bias mitigation in toxicity classification.

1 Introduction

Machine learning (ML) based classification can be biased, as the models learn patterns from the existing biased data. As people use more ML models in real-world settings, biased models can amplify the existing social inequality. A large body of studies gives evidence on bias in ML-based classification. For example, gender bias is found captured by directions in word embeddings like man is to computer programmer and woman is to homemaker[2]. Racial bias is also found in word embeddings, like black is related more to criminals and white is related more to police[8]. Age bias is found in sentiment analysis, for example, sentences with words related to older age are more likely to be classified as negative sentiment, while sentences related to younger age are more likely to be marked positive[3].

A number of studies have contributed to bias mitigation in ML-based text classification. The solutions are usually divided into two categories: debiasing the datasets or debiasing the models. Modifying the training datasets is a straightforward solution but it is very costly to manually check and redo the annotation for the biased datasets. An example can be reweighting data points[5]. On the other hand, most of the studies choose to modify the word representations. One way is to zero out components in presupposed bias feature space[2], but this solution is not generalizable. Another way is to apply adversarial training[10]. However, this method is notoriously hard to train.

In this project, we use a novel method for removing information from neural representations for bias mitigation in text classification called Iterative Null-space Projection (INLP)[9]. We choose this method as it can be generalizable and need no retraining. We aim to use INLP to mitigate bias in the toxicity classification of text comments, so features related to gender, race, and religion can be protected without much sacrifice in the classifier performance. Our experiments can be found in the GitHub repository [1].

2 Approach

The paper's approach utilizes some features from two previous methods for selectively removing specific information: projections on a pre-specified, user-provided direction[2] and adding an adversarial object to an end-to-end training program[10]. As in the projection methods, it uses the notion of linear projection. As in the adversarial methods, it is data-driven in the direction it removes: it does not presuppose specific directions for protecting

[1] https://github.com/assemzh/NullitOut

attributes, but rather learns these directions.

2.1 Mathematical model

Iterative Nullspace projection. Given a set of vectors $x_i \in R^n$ and a set of corresponding attributes $z_i \in Z$, we are looking for a linear guarding function g such that it removes the linear dependence between Z and X.

The high level description of the approach is as follows: let c be a trained linear classifier, parametrized by a matrix $W \in R^{k \times d}$, that predicts a property z with some accuracy. It is possible to construct a projection matrix P such that $W(Px) = 0$ for all x, meaning that W is not able to predict attributes Z based on X. This single step in the algorithm is applied iteratively, training new classifiers W' and obtaining a new projection based on the previous projection. This procedure is repeated until no classifier W' can be trained, i.e. there is no linear relationship between Z and X. The projection matrix is obtained via nullspace projection, as described in the next subsection.

Nullspace Projection. The linear classification between W and a test point x has a simple geometric interpretation. x is projected onto the subspace spanned by the rows of W and it's classified according to the dot product between x and W's rows, which is proportional to the components of x in the direction of W's rowspace. Therefore, if the components of x were zero in the direction of W's rowspace, then W cannot infer any information from the projection of x. Since the nullspace is defined as the orthogonal subspace to the rowspace of a matrix, zeroing out the components of x is equivalent to mapping x onto the nullspace of W. The following figure illustrates the algorithm for the 2-dimensional binary classification.

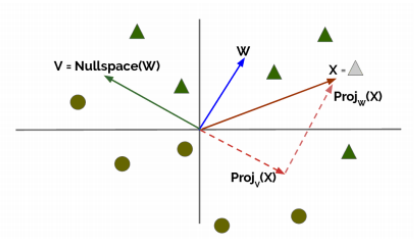

Figure 1: 2D nullspace

For an algebraic interpretation, the nullspace projection for W is defined as the space $N(W) = \{x|Wx = 0\}$. Given the basis vectors of $N(W)$ we can construct a projection $P_{N(W)}$ into $N(W)$, yielding $W(P_{N(W)}x) = 0$ for all x.

This suggests the following solution for protecting attributes Z for a set of vectors X: train a linear classifier parametrized by W_0 to predict Z from X, calculate its nullspace, find the orthogonal projection matrix $P_{N(W_0)}$, and use this projection to remove the components of X, that were used to predict Z.

***Iterative* Projection**. One single projection of X onto the nullspace of W does not suffice to protect the attributes Z. Classifiers can often still be trained to recover Z from the projected X, as there are often multiple linear directions (hyperplanes) that can partially capture relations in multidimensional space. This can be solved by a series of projections. After obtaining $P_{N(W_0)}$, we train classifier W_1 on $P_{N(W_0)}X$, and obtain a projection matrix $P_{N(W_1)}$, train a classifier W_2 on $P_{N(W_1)}P_{N(W_0)}X$ and so on, until no classifier can be trained.

Algorithm 1 Iterative Nullspace Projection (INLP)

Input: (X, Z): a training set of vectors and protected attributes
n: number of rounds
Output: A projection matrix P

function GETPROJECTIONMATRIX(X, Z):
  $X_{projected} \leftarrow X$
  $P \leftarrow I$
  for $i \leftarrow 1$ **to** n **do**
    $W_i \leftarrow$TrainClassifier($X_{projected}, Z$)
    $B_i \leftarrow$GetNullSpaceBasis(W_i)
    $P_{N(W_i)} \leftarrow B_i B_i^T$
    $P \leftarrow P_{N(W_i)} P$
    $X_{projected} \leftarrow P_{N(W_i)} X_{projected}$
  end for
  return P
end function

3 Data & Experiments

In this work we have conducted experiments by applying Iterative Nullspace Projection (INLP) algorithm on 3 different tasks: (1) word vectors debiasing, (2) bias mitigation in profession classification of biographies and (3) bias mitigation in toxicity classification task. The first two tasks are related to the replication experiments. In the last task, we aim to contribute to the original paper by validating INLP algorithm on the 3 different bias domains, namely gender, race and religion.

Word Vectors Debiasing. The replication dataset for the first task consisted of GloVe word embeddings [11] 7,500 most male-biased, 7,500 most female-biased words (measured by the projection on the he—she direction as illustrated in Figure 2), and 7,500 most neutral vectors are used in this experiment which then are further randomly divided into training set (49%), development set (21%), and test set (30%) [2].

FastText Representation. The dataset used in the second task is obtained from short biographies scraped from the web and annotated by gender and profession. The data contains 393,423 biographies with 28 classes of profession and are split into training set (65%), development set (10%), and test set (10%) following the experiment set up of [1]. We only experimented with FastText [7] token representation of the words.

Target Dataset. For our target task we used Jigsaw Toxicity classification dataset which is available online on Kaggle platform [3]. It provides a large number of Wikipedia comments which have been labeled by human raters for toxic behavior. The types of toxicity are: toxic, severe toxic, obscene, threat, insult, and identity hate. To simplify the task we merged these labels into one *toxicity* label, setting it to *'toxic'* if a comment has at least one non-negative score in one of the aforementioned fields, otherwise if the sum of scores is equal to zero we set the label to *'non-toxic'*. The dataset also contains fields related to the gender, race and religion identities mentioned in the comment text. We have removed all null values and split the dataset into 3 files for separate experiments. You can find information about the dataset in the Table 1 and Figures 3, 6 and 9.

4 Results

4.1 Replication results

4.1.1 Word Vectors Debiasing

This experiment is done in an attempt to remove gender bias from GloVe word embeddings. We use a L2-regularized SVM classifier[4] trained to discriminate between the 3 classes: male-biased, female-biased and neutral. Using mean of projections of vectors on the gender unit vector as our evaluation metrics, we compare the score of masculine bias and feminine bias on GloVe word embeddings before and after projecting it through 35 iterations of INLP.

[2] https://github.com/shauli-ravfogel/nullspace_projection

[3] https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge/data

Identity class	# of comments	# toxic
Gender Class		
female	30,189	14,965
male	11,186	5,606
transgender	852	592
Race Class		
white	13,086	10,244
black	5,354	4,409
asian	1,114	519
latino	332	175
Religion Class		
christian	16,955	8,195
muslim	10,817	7,791
jewish	3,239	2,128
atheist	532	297
buddhist	144	85
hindu	84	42

Table 1: Dataset used for bias mitigation in toxicity classification task.

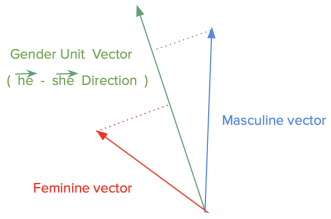

Figure 2: Gender vector

	Our Result	Paper Report
Masculine Bias - Before	1.007	1.007
Masculine Bias - After	0.0009	0.0005
Feminine Bias - Before	-0.973	-0.973
Feminine Bias - After	-0.0004	-0.00008

Table 2: Measure of masculine bias and feminine bias before and after applying INLP

4.1.2 FastText Representation

This experiment attempts to mitigate gender bias in profession classification of biographies using INLP. Multiclass logistic regression classifier are trained to predict the profession of the biography's subject based on BWV input representation where FastText token representations[6] of the words in the biography are summed. To debias the profession classifier, INLP is applied to make the clas-

sifier oblivious to gender information. In another word, after debiasing, classifier would not be able to predict gender information from the input data correctly. After applying INLP for 150 iterations on the classifier, decrease in accuracy of gender classifier is seen, which marks the success of the debiasing of the classifier. However, as a trade-off, profession classifier experience a decrease in performance, but this degradation of performance is small compared to the debiasing performance we obtained from applying INLP.

	Apply 150 iterations of INLP	Accuracy	
		Our Result	Paper Report
Profession Classifier	Before INLP	77.2%	78.1%
	After INLP	71.7%	73.0%
Gender Classifier (Used for algo)	Before INLP	98%	98%
	After INLP	57.5%	57.3%

Table 3: Accuracy of profession classifier and gender classifier before and after applying INLP

4.2 Results of improved approach

4.2.1 Gender bias mitigation

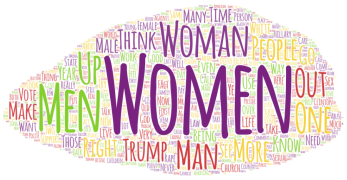

Figure 3: Word Cloud from Gender comments

In the dataset labeled with gender identity, there are three gender groups: male, female, and transgender. The accuracy graph below shows that the approach performs well, as the gender classifier has its accuracy drops a lot after iterations, while there is insignificant drop for the toxicity classifier. This means the gender-related features are successfully protected from the toxicity classification. The confusion matrices for each gender group before and after iterations provide a straightforward view of the results, as the color of each gender group's confusion matrix becomes more similar to each other. This shows that the gender gap for the toxicity classification is reduced. We found that comments containing identity 'male' are more likely to be mistreated as toxic comments compared to other genders (higher false positive), but the proportion of false positives decreases after 15 iterations. Besides, the gap of recall scores between the three gender groups also decreases significantly, indicating less biased results in finding out toxic comments.

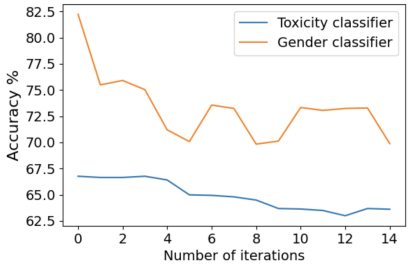

Figure 4: Graph of accuracy of toxicity classifier and gender classifier throughout INLP iterations

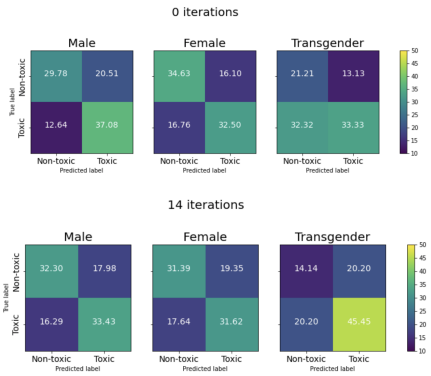

Figure 5: Confusion matrix in % (Gender)

		Male	Female	Transgender
Recall $\frac{TP}{TP+FN}$	0 iteration	74.6%	66.0%	50.8%
	14 iteration	67.2%	64.2%	69.2%

Table 4: Recall of toxicity classifier before and after applying INLP for 14 iterations in debiasing gender

4.2.2 Racial bias mitigation

Figure 6: Word cloud from Race comments

Four racial groups are included in the dataset labeled with racial identity. Similar to gender bias, as shown in the accuracy graph, there is not much decrease in the toxicity classifier but there is a large decrease in the race classifier. This means that the racial features are successfully protected after the

iterations, without much sacrifice in the main task performance. As shown in the confusion matrices of the toxicity classification results, the colors of the matrices became similar between each racial group. This shows that the racial gap is reducing for the toxicity classification. We also found that the gap of recall scores decreases significantly between white, black, and asian groups, showing less biased results in finding out toxic comments, after INLP approach.

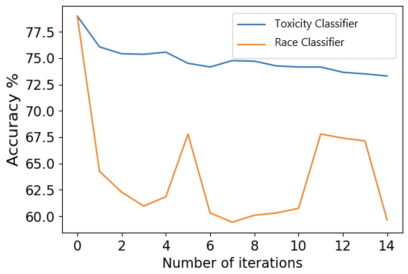

Figure 7: Graph of accuracy of toxicity classifier and race classifier throughout INLP iterations

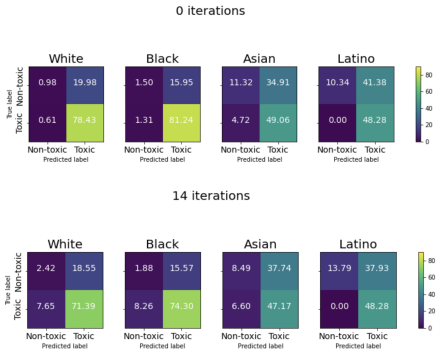

Figure 8: Confusion matrix in % (Race)

		White	Black	Asian	Latino
Recall $\frac{TP}{TP+FN}$	0 iteration	99.2%	98.4%	91.2%	100.0%
	14 iteration	90.3%	90.0%	87.7%	100.0%

Table 5: Recall of toxicity classifier before and after applying INLP for 14 iterations in debiasing race

4.2.3 Religion bias mitigation

Figure 9: Word Cloud from Religion comments

We also applied INLP to comments that come with religion label. We used a total of 31,771 comments with religion and toxic proportions as listed in Table 1. The graph below shows how the performance of toxicity classifier and religion classifier changes with regard to number of iterations. We can see that the performance of religion classifier lowers much more than the performance of toxicity classifier. It shows that INLP works well in guarding the religion feature of the comments with side effect of small degradation of toxicity classifier performance.

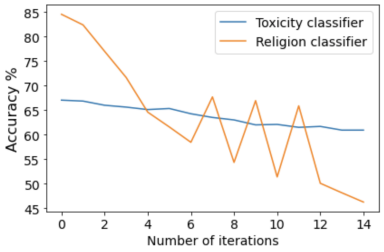

Figure 10: Graph of accuracy of toxicity classifier and religion classifier throughout INLP iterations

As before, we are trying to evaluate bias through confusion matrix. When metrics are calculated separately for each religion, we can compare model performance for each group and see whether it differs across religion. If there is difference in the model performance, it means the model is biased. Here it shows how the confusion matrix changes with regard to number of iterations. We calculate the model precision for each religion group and we can see that initially the model does not perform equally for each religion. Considering that only Christian, Jewish and Muslim group had enough training data, we compare only three of them.

Initially the performance of model in classifying toxicity for Christian is worse than for the Muslim and Jewish groups. The model tends to classify comments that associate with Christian to be toxic even though they are not. After the iteration, the toxicity classifier performs similarly for the Jewish and Muslim groups with 70.4%. However, performance of the Christianity group lowers a lot and its gap with other religions still exists, which is not what we expected from the work of INLP. This might be because of the unbalanced dataset where a larger portion of comments stood for Christianity group. Also, Christianity group had a more balanced ratio between non-toxic and toxic comments while both Muslim and Jewish groups had more

toxic comments.

		Atheist	Buddhist	Christian	Hindu	Jewish	Muslim
Precision $\frac{TP}{TP+FP}$	0 iteration	61.3%	80%	**65.1%**	83.3%	**71.8%**	**72.4%**
	14 iteration	61.7%	72.7%	**52.2%**	83.3%	**70.4%**	**70.4%**

Table 6: Precision of toxicity classifier before and after applying INLP for 14 iterations in debiasing religion

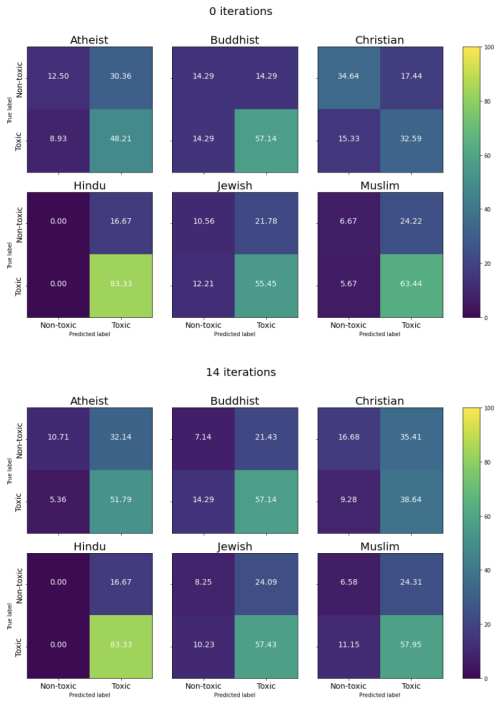

Figure 11: Confusion matrix in % (Religion)

4.3 Interpretation of results

From the three bias groups that we experimented with, we see that the accuracy of religion classifier drops significantly compared to other bias groups. However, we see from the confusion matrix that it does not perform well in decreasing bias between religion classes and cannot maintain the main toxicity task performance well. The possible explanation for this phenomenon is that comments containing information about religion are more complex compared to gender and race. We found that comments in the religion dataset are usually 10% longer and have more unique words on average than the comments with from other bias groups. Also, from the Table 1 we can see that dataset we used for the analysis is not balanced, and since for the religion group there are more number of religious classes the classification task becomes more complex as well.

5 Discussions

There are two major limitations of this study: linearity constraint and data limitation. Since the INLP method utilizes linear projections it cannot capture non-linear biases incorporated in more complex models. Therefore, one of the significant limitations of the INLP approach is that it can only be applied to the linear classifiers. The second limitation is that this method works well only in the setting with balanced training dataset. We can see from the Table 1, that data we used for our experiments is not equally distributed across classes, thus algorithms did not work well in some scenarios where data is limited.

To address these limitation we propose collecting more data for the minority groups by distant supervision techniques. Since identity information was extracted directly from the comment text, more data can be collected and automatically labeled by minority classes before the manual toxicity annotation step.

6 Conclusion

INLP is a simple but powerful approach for bias mitigation in ML classification tasks. In this work we have shown how INLP method can be generalized to different bias mitigation domains. We improved the chosen paper by applying the method to the Jigsaw Toxicity dataset, which contains toxic and non-toxic comments labeled with the mentioned demographic identity in the comments. Three sub-datasets were preprocessed and included in our experiments, with each of them represents gender, race and religion classes respectively.

In all experiments, the accuracy of identity classifier decreases significantly without sacrificing much of the performance of toxicity classifier. This indicates the bias identity features are successfully protected with INLP method. From confusion matrices, we can see the biases between identity groups in gender and race classes are decreased after debiasing. We observed that religion bias mitigation is a more complex task than the gender or racial bias mitigation tasks. Further analysis with more balanced datasets is required to make more solid conclusions about the differences between bias groups.

References

[1] Maria De-Arteaga et al. "Bias in bios: A case study of semantic representation bias in a high-stakes setting". In: *proceedings of the Conference on Fairness, Accountability, and Transparency*. 2019, pp. 120–128.

[2] Tolga Bolukbasi et al. "Man is to computer programmer as woman is to homemaker? debiasing word embeddings". In: *Advances in Neural Information Processing Systems* (2016).

[3] Mark Dıaz et al. "Addressing age-related bias in sentiment analysis". In: *2018 CHI Conference on Human Factors in Computing Systems, CHI 2018*. Association for Computing Machinery. 2018.

[4] Marti A. Hearst et al. "Support vector machines". In: *IEEE Intelligent Systems and their applications* 13.4 (1998), pp. 18–28.

[5] Heinrich Jiang and Ofir Nachum. "Identifying and correcting label bias in machine learning". In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2020, pp. 702–712.

[6] Armand Joulin et al. "Bag of Tricks for Efficient Text Classification". In: *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*. 2017, pp. 427–431.

[7] Armand Joulin et al. "Fast linear model for knowledge graph embeddings". In: *arXiv preprint arXiv:1710.10881* (2017).

[8] Thomas Manzini et al. "Black is to Criminal as Caucasian is to Police: Detecting and Removing Multiclass Bias in Word Embeddings". In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. 2019, pp. 615–621.

[9] Shauli Ravfogel et al. "Null It Out: Guarding Protected Attributes by Iterative Nullspace Projection". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 2020, pp. 7237–7256.

[10] Qizhe Xie et al. "Controllable invariance through adversarial feature learning". In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 2017, pp. 585–596.

[11] Jieyu Zhao et al. "Learning gender-neutral word embeddings". In: *arXiv preprint arXiv:1809.01496* (2018).