Proyecto 3: Unintended Bias in Toxicity Classification

David Fajardo
Diego Restrepo
Andrés Motta

Descripción del modelo

1. CNN con la siguiente arquitectura:
    a. Embeddings dimension = 100
    b. Dropout rate = 0.5
    c. Learning rate 0.00001
    d. Epochs = 4
    e. Batch Size 200
    f. Optimizer = Nadam()
2. Optimizador:
3. Métrica: Categorical cross entropy
4. Link Colab
5. Resultado final:

Balanceo de los datos

Link

	False	True	Total	pFalse	pTrue	NewFalse
black	10223	4678	14901	0.686061	0.313939	24082.0
homosexual_gay_or_lesbian	7876	3121	10997	0.716195	0.283805	15011.0
white	18044	7038	25082	0.719400	0.280600	33568.0
muslim	16225	4781	21006	0.772398	0.227602	18836.0
psychiatric_or_mental_illness	3859	1030	4889	0.789323	0.210677	3694.0
jewish	6411	1240	7651	0.837930	0.162070	2682.0
male	37799	6685	44484	0.849721	0.150279	11224.0
female	46118	7311	53429	0.863164	0.136836	7496.0
christian	36750	3673	40423	0.909136	0.090864	-9815.0

Measuring and Mitigating Unintended Bias in Text Classification

Lucas Dixon
ldixon@google.com

John Li
jetpack@google.com

Jeffrey Sorensen
sorenj@google.com

Nithum Thain
nthain@google.com

Lucy Vasserman
lucyvasserman@google.com

Jigsaw

Abstract

We introduce and illustrate a new approach to measuring and mitigating unintended bias in machine learning models. Our definition of unintended bias is parameterized by a test set and a subset of input features. We illustrate how this can be used to evaluate text classifiers using a synthetic test set and a public corpus of comments annotated for toxicity from Wikipedia Talk pages. We also demonstrate how imbalances in training data can lead to unintended bias in the resulting models, and therefore potentially unfair applications. We use a set of common demographic identity terms as the subset of input features on which we measure bias. This technique permits analysis in the common scenario where demographic information on authors and readers is unavailable, so that bias mitigation must focus on the content of the text itself. The mitigation method we introduce is an unsupervised approach based on balancing the training dataset. We demonstrate that this approach reduces the unintended bias without compromising overall model quality.

Introduction

With the recent proliferation of the use of machine learning for a wide variety of tasks, researchers have identified unfairness in ML models as one of the growing concerns in the field. Many ML models are built from human-generated data, and human biases can easily result in a skewed distribution in the training data. ML practitioners must be proactive in recognizing and counteracting these biases, otherwise our models and products risk perpetuating unfairness by performing better for some users than for others.

Recent research in fairness in machine learning proposes several definitions of fairness for machine learning tasks, metrics for evaluating fairness, and techniques to mitigate unfairness. The main contribution of this paper is to introduce methods to quantify and mitigate unintended bias in text classification models. We illustrate the methods by applying them to a text classifier built to identify toxic comments in Wikipedia Talk Pages (Wulczyn, Thain, and Dixon 2017).

Initial versions of text classifiers trained on this data showed problematic trends for certain statements. Clearly non-toxic statements containing certain identity terms, such as "I am a gay man", were given unreasonably high toxicity scores. We call this *false positive bias*. The source of this bias was the disproportionate representation of identity terms in our training data: terms like "gay" were so frequently used in toxic comments that the models over-generalized and learned to disproportionately associate those terms with the toxicity label. In this work, we propose a method for identifying and mitigating this form of unintended model bias.

In the following sections, we describe related work, then discuss a working definition of unintended bias in a classification task, and distinguish that from "unfairness" in an application. We then demonstrate that a significant cause of unintended bias in our baseline model is due to disproportionate representation of data with certain identity terms and provide a way to measure the extent of the disparity. We then propose a simple and novel technique to counteract that bias by strategically adding data. Finally, we present metrics for evaluating unintended bias in a model, and demonstrate that our technique reduces unintended bias while maintaining overall model quality.

Related Work

Researchers of fairness in ML have proposed a range of definitions for "fairness" and metrics for its evaluation. Many have also presented mitigation strategies to improve model fairness according to these metrics. (Feldman et al. 2015) provide a definition of fairness tied to demographic parity of model predictions, and provides a strategy to alter the training data to improve fairness. (Hardt, Price, and Srebro 2016) presents an alternate definition of fairness that requires parity of model performance instead of predictions, and a mitigation strategy that applies to trained models. (Kleinberg, Mullainathan, and Raghavan 2016) and (Friedler, Scheidegger, and Venkatasubramanian 2016) both compare several different fairness metrics. These works rely on the availability of demographic data about the object of classification in order to identify and mitigate bias. (Beutel et al. 2017) presents a new mitigation technique using adversarial training that requires only a small amount of labeled demographic data.

Very little prior work has been done on fairness for text classification tasks. (Blodgett and O'Connor 2017), (Hovy and Spruit 2016) and (Tatman 2017) discuss the impact of

Balanceo de los datos

	False	True	Total	pFalse	pTrue	NewFalse
black	10223	4678	14901	0.686061	0.313939	24082.0
homosexual_gay_or_lesbian	7876	3121	10997	0.716195	0.283805	15011.0
white	18044	7038	25082	0.719400	0.280600	33568.0
muslim	16225	4781	21006	0.772398	0.227602	18836.0
psychiatric_or_mental_illness	3859	1030	4889	0.789323	0.210677	3694.0
jewish	6411	1240	7651	0.837930	0.162070	2682.0
male	37799	6685	44484	0.849721	0.150279	11224.0
female	46118	7311	53429	0.863164	0.136836	7496.0
christian	36750	3673	40423	0.909136	0.090864	-9815.0

	False	True	Total	pFalse
psychiatric_or_mental_illness	7553	1030	8583	0.879995
jewish	9093	1240	10333	0.879996
homosexual_gay_or_lesbian	22887	3121	26008	0.879998
black	34305	4678	38983	0.879999
male	49023	6685	55708	0.879999
female	53614	7311	60925	0.880000
white	51612	7038	58650	0.880000
muslim	35061	4781	39842	0.880001
christian	36750	3673	40423	0.909136

- Se calculan los ‘nuevos falsos’ que se necesitan para que la muestra esté balanceada.
- Se agregan usando una BD de una competencia pasada con data de Wikipedia
- pFalse mejora para todos los labels

Balanceo de los datos

	bnsp_auc	bpsn_auc	subgroup	subgroup_auc	subgroup_size
7	0.956610	0.797312	heterosexual	0.802083	278
8	0.903317	0.864258	hindu	0.807895	115
17	0.920739	0.863042	transgender	0.808269	539
18	0.960948	0.780904	white	0.809513	5001
3	0.961562	0.772677	black	0.810990	3009
2	0.944649	0.850085	bisexual	0.812500	63
9	0.950213	0.802589	homosexual_gay_or_lesbian	0.814209	2212
14	0.958383	0.804230	muslim	0.826423	4238
1	0.933107	0.869534	atheist	0.843243	282
4	0.900079	0.871015	buddhist	0.846154	106
15	0.954467	0.854737	other_religion	0.851307	63
12	0.944769	0.857565	latino	0.858643	403
11	0.946890	0.855190	jewish	0.863981	1466
16	0.957296	0.850927	psychiatric_or_mental_illness	0.879133	967
6	0.939697	0.885807	female	0.884881	10696
13	0.945808	0.877540	male	0.885673	8936
5	0.928320	0.916149	christian	0.904672	8128

El AUC score aumenta para los labels subrepresentados

	bnsp_auc	bpsn_auc	subgroup	subgroup_auc	subgroup_size
4	0.935988	0.893363	jewish	0.887036	2073
1	0.943064	0.890266	female	0.892997	12128
3	0.920676	0.920539	christian	0.899249	8257
0	0.950421	0.891456	male	0.905964	11221
5	0.961619	0.887040	muslim	0.909287	7884
2	0.954809	0.911464	homosexual_gay_or_lesbian	0.919764	5183
7	0.964550	0.902771	white	0.927194	11758
8	0.955568	0.909859	psychiatric_or_mental_illness	0.930308	1711
6	0.966872	0.908666	black	0.936038	7702

Qué se puede hacer para aumentar el puntaje?

1. Cambiar el embedding de 100 por el de 300 dimensiones. Sin embargo, sería cambiar de un archivo de 350 mb a +1GB.
2. Parámetros de la CNN: epochs, batch, layers, neurons, activations, etc. Consume muchos recursos.
3. Inicializar el modelo con unos ‘buenos’ pesos iniciales

Gracias