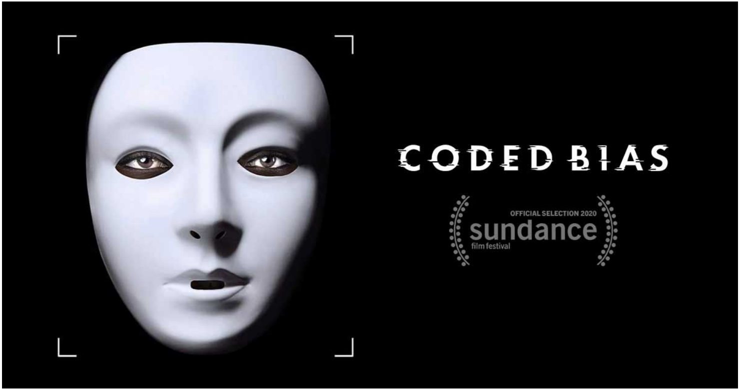
PREDICTING IF AN ONLINE COMMENT IS TOXIC AND REDUCING DATA BIASES

What is bias in machine learning?

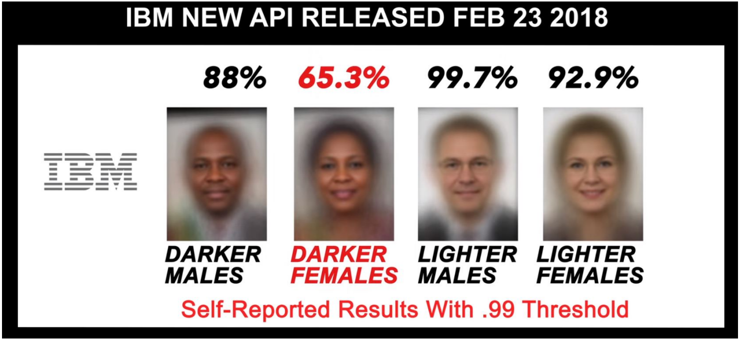
- **Systematic errors** due to **incorrect assumptions**
- Biases are **learned** from the data

- Active area of research in the ML community

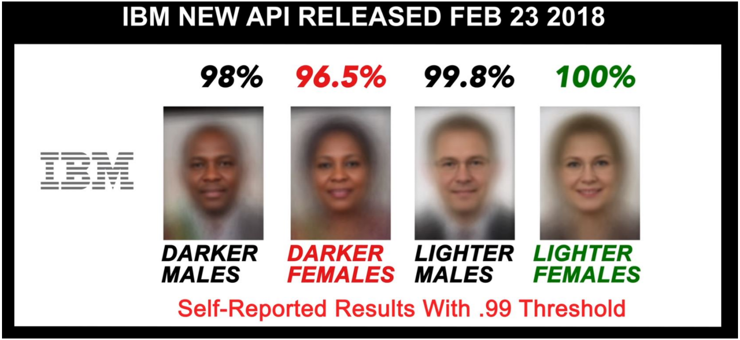
Introduction



Introduction



Introduction



Introduction

Introduction

Steve Wozniak
@stevewoz

Replying to @dhh and @AppleCard

The same thing happened to us. I got 10x the credit limit. We have no separate bank or credit card accounts or any separate assets. Hard to get to a human for a correction though. It's big tech in 2019.

4:51 PM · Nov 9, 2019 · Twitter Web App

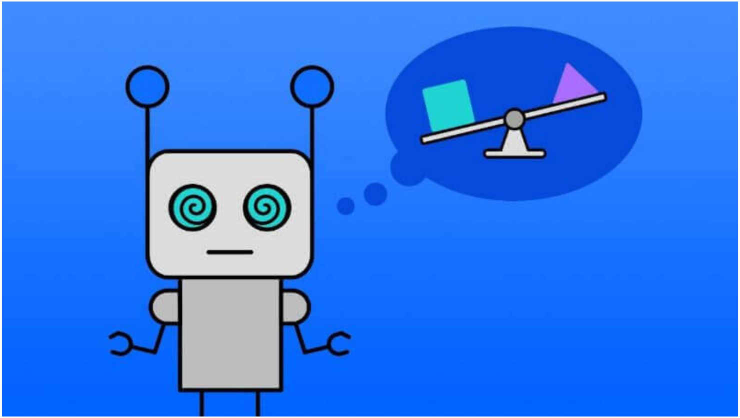
660 Retweets **3.8K** Likes

Bias in NLP

- **NLP:** Natural Language Processing

- Certain **identities** are overwhelmingly **referred to** in **offensive ways**
- Models **incorrectly learn** to associate frequently **attacked minorities** with **toxicity**

Bias in NLP



Bias in NLP

Goal:

- Attempt to **mitigate** this bias using a **solution** proposed by:

  Nuanced Metrics for Measuring Unintended Bias with Real Data for Text Classification (2019)

Dataset

- 2M comments from Wikipedia's talk page

- Each comment was given a **toxicity label** and 21 **identity attributes** by at least 10 annotators (up to thousands)
  - **Comment:** i'm a white woman in my late 60's and believe me, they are not too crazy about me either!!
  - **Toxicity Label:** 0.0
  - **Identity Labels:** female: 1.0, white: 1.0 (all others 0.0)

Dataset

Only **6 identities** will be used in the analysis:

- Female
- Homosexual_gay_or_lesbian
- Christian
- Jewish
- Muslim
- Black

Source: Kaggle competition - Jigsaw Unintended Bias in Toxicity Classification (2019)

Dataset

- Number of comments in dataset
- % of toxic comments within the number above

Female	Homosexual	Christian	Jewish	Muslim	Black	Total
Training Dataset						
42.9k (13.6%)	8.7k (28.9%)	30.2k (9.6%)	6.1k (16.2%)	16.6k (23.0%)	11.7k (32.3%)	**1.5M (7.9%)**
Validation Dataset						
7.6k (13.9%)	1.4k (30.1%)	5.2k (9.0%)	1.1k (17.8%)	2.9k (22.1%)	2.1k (32.0%)	**270K (7.9%)**
Test Dataset						
2k (13.2%)	491 (26.2%)	1.8k (10.3%)	405 (17.0%)	914 (25.6%)	699 (33.9%)	**97K (7.8%)**

Dataset

- Highly **imbalanced** dataset
- **Metric:** Subgroup AUC
  - AUC calculated within identity subgroups
  - Identity = 1

1. Introduction
2. Bias in NLP
3. Dataset
4. **Text Preprocessing Pipeline**
   - Tokenization
   - Normalization
   - Embeddings
5. Models
6. Results
7. Conclusion

Text Preprocessing Pipeline - Tokenization

Text

"NLP is a very exciting field!"

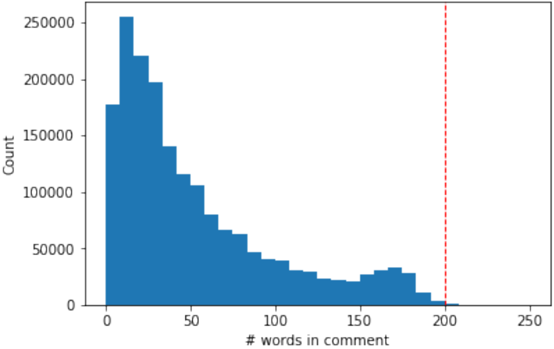
$\downarrow$

Tokens

["NLP", "is", "a", "very", "exciting", "field", "!"]

Text Preprocessing Pipeline - Normalization

- Pad/truncate comments to a pre-defined length



- Maximum length = 200

Text Preprocessing Pipeline - Normalization

- Example:

7 Tokens

[“NLP”, “is”, “a”, “very”, “exciting”, “field”, “!”]

$\downarrow$

10 Tokens

[“NLP”, “is”, “a”, “very”, “exciting”, “field”, “!”, “<PAD>”, “<PAD>”, “<PAD>”]

Text Preprocessing Pipeline - Embeddings

Embeddings: Learned representation for text

Idea: Computers understand numbers not words (tokens)

Token
dog

$\downarrow$

Embedding
[0.23, 0.56, 0.67, 0.05, 0.98, ..., 0.13]

Text Preprocessing Pipeline - Embeddings

Embedding =
GloVe (300-dimensions) + fastText (300-dimensions)

Token
dog

$\downarrow$

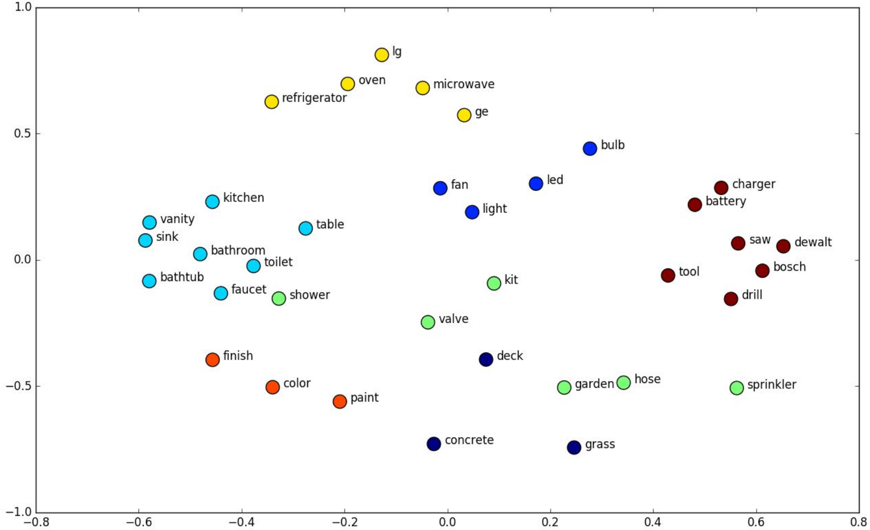
Embedding
$[x_1, x_2, ..., x_{300}, ..., x_{600}]$

Token
$<$PAD$>$

$\downarrow$

Embedding
$[0, 0, 0, ..., 0]$

Text Preprocessing Pipeline - Embeddings

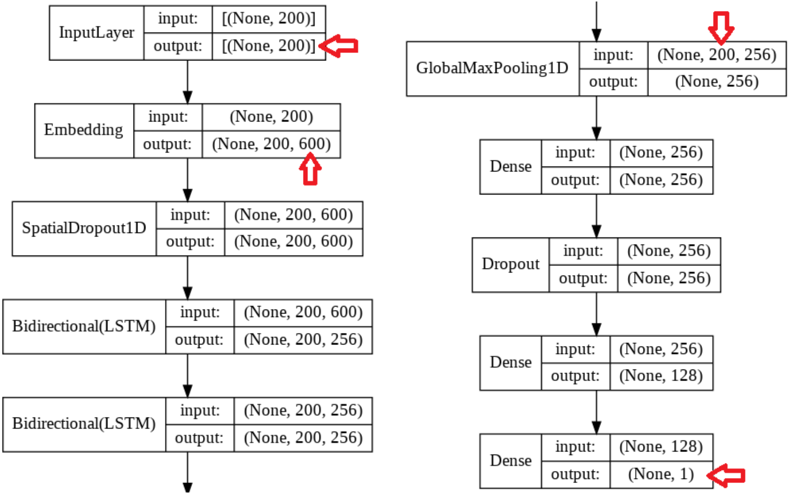


Models

- 2 models
  - **Model 1:** 1.2M trainable params
  - **Model 2:** 1.6M trainable params
- 2 losses (regular loss & custom loss)
- Total of **4 models**

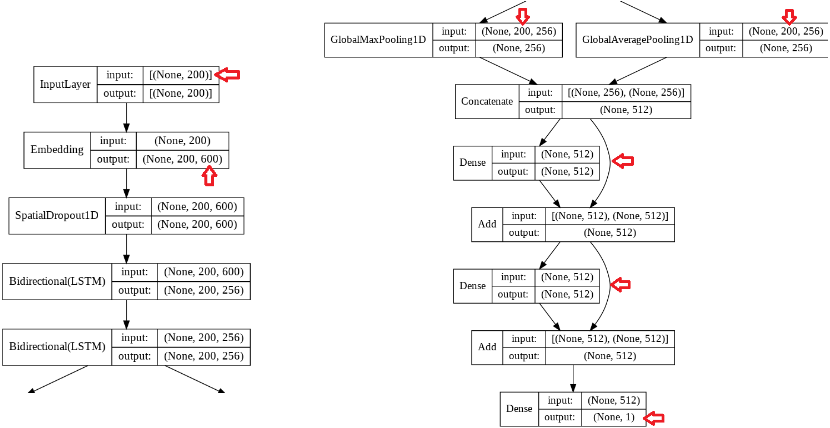
Models

Model 1:



Models

Model 2:



Models - Hyperparameters

- Batch size: 512
- Max length: 200
- Optimizer: Adam
- Learning rate: 0.001

Models - Losses

- **Binary Cross Entropy (*BCE*)**
  - True toxicity label y
  - Predicted toxicity label $\hat{y}$

$$BCE = -\frac{1}{N}\sum_{i}^{N} y_i \, log \hat{y}_i + (1 - y_i) \, log \, (1 - \hat{y}_i)$$

Models - Losses

- **Custom Loss (simplified)**

$$Loss = \begin{cases} 1 * BCE, & \text{if } y = 0 \;\&\; identity = 0 \\ 1.5 * BCE, & \text{if } y = 1 \\ 2 * BCE, & \text{if } y = 0 \;\&\; identity = 1 \end{cases}$$

  - Separability within subgroup
  - Reduce *false positives*
  - Reduce *false negatives*

Results - Metrics

Model 1:
Validation AUC

Female	Homosexual	Christian	Jewish	Muslim	Black	Total
Model 1 (BCE loss)						
92.8	84.0	93.6	86.6	86.1	84.6	92.4
Model 1 (Custom loss)						
93.0	84.6	93.8	87.7	86.7	84.9	92.7

AUC (custom loss) - AUC (BCE loss) $\approx$ 0.3

Results - Metrics

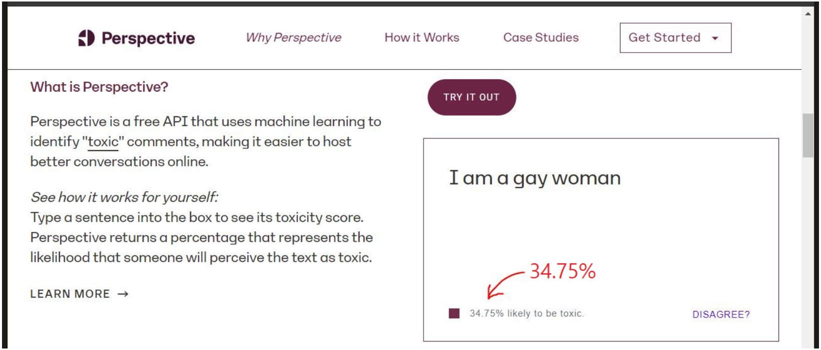
Model 2:
Validation AUC

Female	Homosexual	Christian	Jewish	Muslim	Black	Total
Model 2 (BCE loss)						
93.1	84.7	93.6	87.8	86.5	85.4	92.4
Model 2 (Custom loss)						
93.3	84.1	93.8	88.3	86.9	85.6	92.5

AUC (custom loss) - AUC (BCE loss) $\approx$ 0.3

Conclusion

- Models with the **custom loss** perform slightly **better** than regular **BCE loss** ($\sim$ 0.3 Subgroup AUC)



Conclusion

- In my opinion, **time** and **resources** to train new models make it **impractical**
- **Worth it** in a **competition** (Kaggle) but not for real-world applications

- **Bias** is a **big issue** and more efforts should go towards it


Thank you!

BACK-UP

Text Preprocessing Pipeline - Embeddings

GloVe embedding

- Open-source pre-trained embedding (5.5 GBs)
- 1.9M words
- 300-dimensional embeddings
- Trained on word-word co-occurrence statistics
- Corpus: Common Crawl dataset

Text Preprocessing Pipeline - Embeddings

fastText embedding

- Open-source pre-trained embedding (4.4 GBs)
- 2M words
- 300-dimensional embeddings
- Trained using modified word2vec algorithm
- Corpus: Common Crawl dataset

GloVe

"The dog ran after the man"

	the	dog	ran	after	man
the	0	1	0	1	1
dog	1	0	1	0	0
ran	0	1	0	1	0
after	1	0	1	0	0
man	1	0	0	0	0

GloVe

Probability and Ratio	$k = solid$	$k = gas$	$k = water$	$k = fashion$
$P(k\|ice)$	1.9×10^{-4}	6.6×10^{-5}	3.0×10^{-3}	1.7×10^{-5}
$P(k\|steam)$	2.2×10^{-5}	7.8×10^{-4}	2.2×10^{-3}	1.8×10^{-5}
$P(k\|ice)/P(k\|steam)$	8.9	8.5×10^{-2}	1.36	0.96

fastText

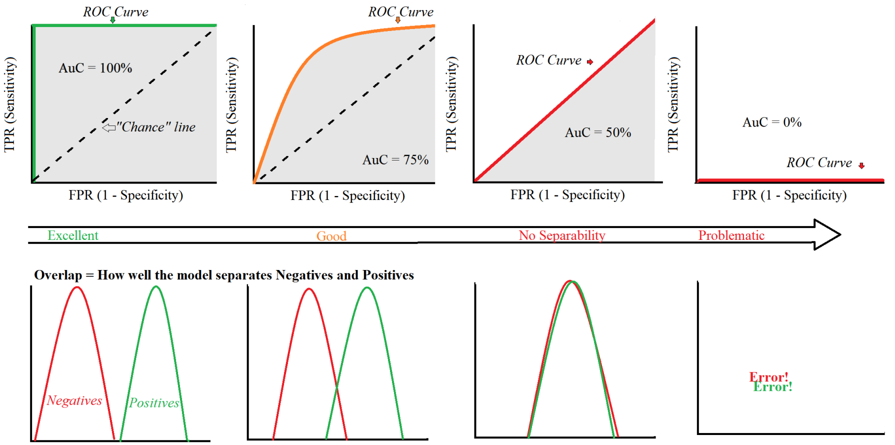
n = 3

"artificial" = $<$ar, art, rti, tif, ifi, fic, ici, ial, al$>$

Common Crawl Dataset

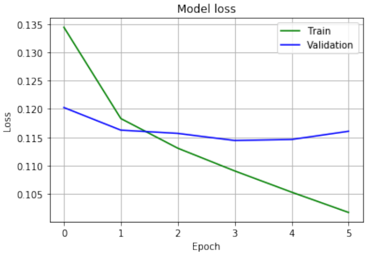
Component	Raw Size	Weight	Epochs	Effective Size	Mean Document Size
Pile-CC	227.12 GiB	18.11%	1.0	227.12 GiB	4.33 KiB
PubMed Central	90.27 GiB	14.40%	2.0	180.55 GiB	30.55 KiB
Books3†	100.96 GiB	12.07%	1.5	151.44 GiB	538.36 KiB
OpenWebText2	62.77 GiB	10.01%	2.0	125.54 GiB	3.85 KiB
ArXiv	56.21 GiB	8.96%	2.0	112.42 GiB	46.61 KiB
Github	95.16 GiB	7.59%	1.0	95.16 GiB	5.25 KiB
FreeLaw	51.15 GiB	6.12%	1.5	76.73 GiB	15.06 KiB
Stack Exchange	32.20 GiB	5.13%	2.0	64.39 GiB	2.16 KiB
USPTO Backgrounds	22.90 GiB	3.65%	2.0	45.81 GiB	4.08 KiB
PubMed Abstracts	19.26 GiB	3.07%	2.0	38.53 GiB	1.30 KiB
Gutenberg (PG-19)†	10.88 GiB	2.17%	2.5	27.19 GiB	398.73 KiB
OpenSubtitles†	12.98 GiB	1.55%	1.5	19.47 GiB	30.48 KiB
Wikipedia (en)†	6.38 GiB	1.53%	3.0	19.13 GiB	1.11 KiB
DM Mathematics†	7.75 GiB	1.24%	2.0	15.49 GiB	8.00 KiB
Ubuntu IRC	5.52 GiB	0.88%	2.0	11.03 GiB	545.48 KiB
BookCorpus2	6.30 GiB	0.75%	1.5	9.45 GiB	369.87 KiB
EuroParl†	4.59 GiB	0.73%	2.0	9.17 GiB	68.87 KiB
HackerNews	3.90 GiB	0.62%	2.0	7.80 GiB	4.92 KiB
YoutubeSubtitles	3.73 GiB	0.60%	2.0	7.47 GiB	22.55 KiB
PhilPapers	2.38 GiB	0.38%	2.0	4.76 GiB	73.37 KiB
NIH ExPorter	1.89 GiB	0.30%	2.0	3.79 GiB	2.11 KiB
Enron Emails†	0.88 GiB	0.14%	2.0	1.76 GiB	1.78 KiB
The Pile	**825.18 GiB**			**1254.20 GiB**	**5.91 KiB**

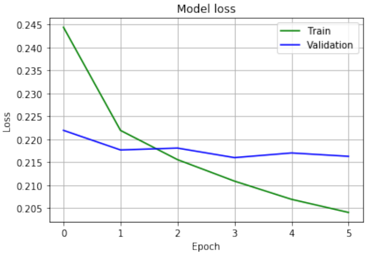
AUC


ROC Curve
AuC = 100%
"Chance" line
TPR (Sensitivity)
FPR (1 - Specificity)
ROC Curve
AuC = 75%
ROC Curve
AuC = 50%
AuC = 0%
ROC Curve
Excellent
Good
No Separability
Problematic
Overlap = How well the model separates Negatives and Positives
Negatives
Positives
Error!
Error!

Losses

Model 1:

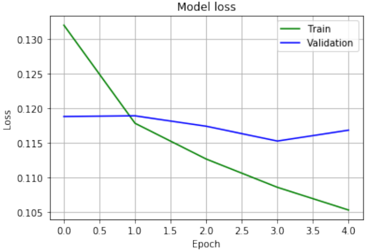


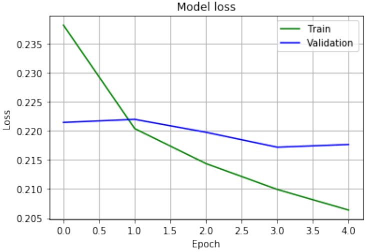


Left: BCE loss (Biased). Right: Custom loss (Unbiased).

Losses

Model 2:





Left: BCE loss (Biased). Right: Custom loss (Unbiased).

Text Preprocessing Pipeline - Normalization

- Lower case conversion

  Dog $\rightarrow$ dog
  Cat $\rightarrow$ cat

- Remove punctuation and special characters (Greek letters, mathematical symbols)

- Expand common abbreviations

  you're $\rightarrow$ you are
  aren't $\rightarrow$ are not

Models - Custom Loss

- **Custom Loss**

  $$Loss = BCE\ (1\ +\ 0.5\,(SUB\ +\ BPSN\ +\ BNSP))$$

  where,

  - **Subgroup (SUB):**
    - SUB=1, if $identity$=1
    - Separability within subgroup
  - **Background Positive Subgroup Negative (BPSN):**
    - $BPSN$=1, if y=1 & $identity$=0
    - Reduce *false positives*
  - **Background Negative Subgroup Positive (BNSP):**
    - $BNSP$=1, if y=0 & $identity$=1
    - Reduce *false negatives*

Source: Nuanced Metrics for Measuring Unintended Bias with Real Data for Text Classification (2019)