Does Unlabelled data improve comment toxicity classification?

Anonymous

1. Introduction

While the proliferation of the internet has allowed for open communication in online spaces, toxic comments can lead to a withdrawal in participation of individuals and a loss of discourse. With this increase in online spaces comes an increase in a magnitude of comments and as such, it has been important to develop automatic classifiers that can detect toxic comments to minimize human moderation and so that these online spaces are safe for individuals to express their opinions in a constructive way. In a toxicity classification competition by Kaggle, models were improved when utilizing comments in the existing training set, translating it into other languages, and back to English to be used as additional training data [1]. Following this, we wanted to explore whether the use of unlabeled data in semi-supervised learning utilizing extra training data could improve classification of toxic comments.

1.1. Related Works

Research into toxic word classifications typically delve into either feature engineering or deep learning using neural networks. Logistic Regression has been found to be perform well as a supervised model without deep learning [2]. Research has been done into Long Short-Term Memory (LSTM) and Bidirectional Encoder Representations from Transformers (BERT) neural networks which have been shown to outperform Naïve Bayes models [3][4]. Following this, Ensemble methods have also been proposed which combine the aforementioned neural networks and have been shown to increase F1 scores compared to single model methods [4][5]. RNN have also been used to create custom embeddings which have been compared to existing embedding models such as GloVe [6].

2. Dataset and Features

2.1. Dataset

A modified version of the Jigsaw "Unintended Bias in Toxicity Classification" dataset from Kaggle was used for this project, which was split into training, development, test, and unlabeled sets consisting of 140000, 15000, 15000 and 200000 instances respectively [1]. Instances included a comment text, toxicity and identity labels with values of 1 indicating toxicity and whether the identity was found in the comment text for the training data respectively. The unlabeled data only included the text comments or features. The distribution of the training data was unbalanced as seen in figure 1, comprising mainly of non-toxic comments.

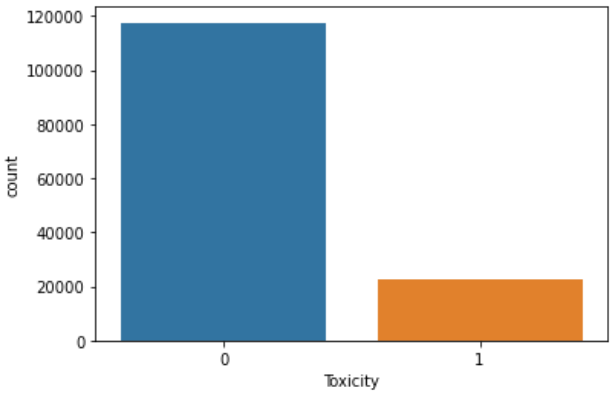

Figure 1. Class distribution of the training set

2.2. Features

Comments were pre-processed into features using two different methods for experimentation, term frequency-inverse document frequency (TFIDF) and embedding.

2.2.1 TFIDF

Comments were first cleaned by removing all stop words and 1000-dimensional feature vectors were

obtained containing the top 1000 highest TFIDF values.

2.2.2 Embedding

Comments were mapped to a 384-dimensional embedding using Sentence Transformer, a pre-trained language model which attempts to distil the meaning of each comment with similar comments located closer together [7].

3. Methods

3.1. Models

Here we propose the baseline model with two supervised and two unsupervised models, along with a semi-supervised model to be used after obtaining evaluation results from the supervised and unsupervised models.

3.1.1 Baseline

Zero R. Zero R is used as a baseline model and works by assigning all test instances to the majority class of the training data.

3.1.2 Supervised

Naïve Bayes. Naïve bayes (NB) is a classification model which assumes that all features are independent of each other. For the embedding features we used Gaussian NB and for the TFIDF features we used Multinomial NB. While this assumption is not true regarding comments where words are intertextually important, the NB model works surprisingly well given that expression of some words are more indicative of toxic comments, which is why we have chosen it as a model.

Logistic Regression. While it is named regression, logistic regression is a classification model that is used often for binary classification tasks. It has been found to work well as a non-neural network model for toxic comment classification which is why we are comparing it against NB [1].

3.1.3 Unsupervised

K-means. K-means is a simple unsupervised classification model used to try to cluster unlabeled data. The hyperparameter number of clusters was specified to 2 for toxic and non-toxic.

T-SNE and DBSCAN. Due to the curse of dimensionality where distances converge to the same value as the number of dimensions increase, we wanted to use a dimensionality reduction technique so that we could perform DBSCAN to effectively form distinct clusters which we could then assign to labels.

3.2. Semi-supervised Model

For the semi-supervised model, we plan to use the supervised and unsupervised models with the highest evaluation scores. By adding our labelled unsupervised data to the training data, we hope to improve the performance of the supervised model.

3.3. Evaluation

For evaluation we used ROC-AUC and Precision/Recall/F1Score. A high precision means that when a comment is labelled toxic it is actually toxic, while a high recall means that we have identified a high number of all toxic comments. Using these metrics, we propose that a high precision with decent recall is ideal as it will lead to the removal of a moderate amount of actual toxic comments without removing too many non-toxic comments that are classified as toxic, as this could also lead to a decrease in participation. ROC-AUC was used as the accuracy metric for the test data with a high score indicating that were a high number of true positive labels and a high number of false positive labels. Due to the imbalance of the training data, ROC-AUC was not as useful as Precision/Recall/F1 for our evaluation.

4. Results

4.1. Models

4.1.1 Baseline

Zero R. As seen in tables 1 and 2, Zero R achieved a ROC AUC of 0.5 which serves as a good baseline for comparison. Precision/F1 were both 0 as they we were interested in toxic comments and all test instances were assigned as non-toxic due to the majority distribution.

4.1.2 TFIDF

Naïve Bayes. When using Multinomial NB with TFIDF features precision was high but recall low as seen in table 1, indicating that the classifier was

accurate in predicting toxicity but only on a small subset of comments. In comparison, using Gaussian NB with embedding features performed better regarding F1 score as seen in table 2.

Logistic Regression. When using LR, the evaluations of the features were similar with the embedding being slightly better. It also performed better in comparison to NB and as such we decided to use LR in the semi-supervised model.

4.1.3 Unsupervised

K-means. We ran K-means multiple times with different random states and selected the state with the highest evaluations as seen in tables 1 and 2. TDIDF was more balanced in F1/precision/recall whilst embedding had a high recall but low precision.

T-SNE and DBSCAN. Unfortunately, we were unable to adequately complete this section due to memory constraints, however we were able to generate some a T-SNE plot with perplexity 50 of the first 10,000 TDIDF unlabeled features that looked interesting as seen in figure 2.

Model	ROC AUC	F1	Precision	Recall
Zero-R	0.50	0	0	0
NB	0.76	0.04	0.74	0.02
LR	0.78	0.31	0.64	0.20
K-Means	-	0.34	0.33	0.33
T-SNE + DBSCAN	-	-	-	-

Table 1. Model evaluations using TFIDF features

Model	ROC AUC	F1	Precision	Recall
Zero-R	0.50	0	0	0
NB	0.75	0.45	0.34	0.69
LR	0.82	0.37	0.65	0.26
K-Means	-	0.36	0.22	0.88
T-SNE + DBSCAN	-	-	-	-

Table 2. Model evaluations using embedding features

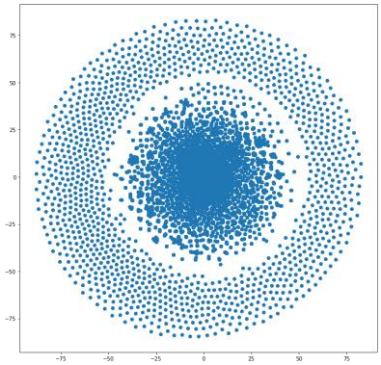

Figure 2. T-SNE plot of unlabeled TDIDF features

4.2. Semi-supervised

Unfortunately, we ran out of time and were unable to complete the necessary steps to obtain the results for semi-supervised learning using additional data obtained from K-Means and combining it with the original dataset to be used in training the logistic regression classifier.

5. Discussion

When comparing the features, the difference in the two NB models was most likely due to the sparsity of the TFIDF feature vector which contained a large number of zeroes, whilst in comparison the embedding feature vector was denser. There may have been some words in the TFIDF features that were highly indicative of toxicity and hence the high precision, but for the rest of the words the classifier may have been unable to adequately distinguish toxicity due to the sparsity of the distances which is seen in the extremely low recall. The high recall of the K-means classifier with low precision for the embedding features indicated that model was assigning a larger number of toxic labels in comparison to the other models, and this may have been due to the 2 clusters hyperparameter specified. As such, for K-means it would have also been better to use a heuristic such as the elbow method to determine the optimal number of clusters for K-means and assigning labels based on the proportion of toxic and non-toxic comments found in the training set instead of just selecting 2 clusters. This approach may have improved the evaluations for K-means. In the Kaggle competition where the use of comments in the existing training set, translating it to other languages and back to English to be used as additional training data increased the performance of the

classifiers, this was most likely due to the extra training data with correct labels. Since we did not obtain evidence, we hypothesize that the semi-supervised learning method we proposed would not have increased the performance of the classifier and instead would have decreased it since the extra training data did not have a high number of correct labels as seen by the evaluation results of K-Means. Additionally, we performed a learning curve of the LR classifier as seen in figure 3 which showed that the training score and cross-validation score converged quickly, indicating underfitting which meant that our model was too simple, and that additional training data would not lead to increases in performance. While the T-SNE plot using unlabeled TDIDF features may represent two distinct clusters, due to the loss in dimensionality and hence information when creating the T-SNE plot, clusters seen are most likely not to represent actual clusters in the original data and hence we hypothesize that the T-SNE + DBSCAN approach would also lead to a decrease in performance with the same reasoning for K-means aforementioned.

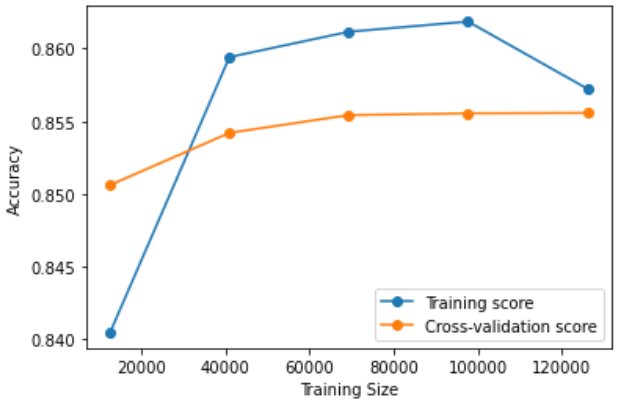

Figure 3. Learning curve for the LR classifier

6. Conclusions

We were able to implement two supervised and one unsupervised model which were all improvements to the baseline model. However, we were not able to adequately answer the research question due to limitations in the methods, research design, results and lack of time spent. Instead of only selecting the two best unsupervised and supervised for semi-supervised learning, it would have been best to test all combinations to get a better understanding of how the models interacted with one another. Further work would be to revisit our proposed topic and explore it properly, as well as explore more complex models such as deep learning/neural network approaches. We apologize for the lack of results and in-depth discussion of whether unlabeled data improves comment toxic classification.

References

[1] Jigsaw/Conversation AI. Jigsaw unintended bias in toxicity classification. https://www.kaggle.com/competitions/jigsaw-unintended-bias-in-toxicity-classification Accessed: July, 2022

[2] Pallam Ravi, Greeshma S Hari Narayana Batta, and Shaik Yaseen. Toxic comment classification. International Journal of Trend in Scientific Research and Development (IJTSRD), 2019.

[3] Sara Zaheri, Jeff Leath, and David Stroud. Toxic comment classification. SMU Data Science Review, 3(1):13, 2020.

[4] Hao Li, Weiquan Mao, Hanyuan Liu. Toxic Comment Detection and Classification. http://cs229.stanford.edu/proj2019spr/report/71.pdf Accessed: October, 2022

[5] Betty Van Aken, Julian Risch, Ralf Krestel, and Alexander Loser. Challenges for toxic comment ¨ classification: An in-depth error analysis. arXiv preprint arXiv:1809.07572, 2018

[6] Manav Kohli, Emily Kuehler, John Palowitch. Paying attention to toxic comments online. https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1184/reports/6856482.pdf Accessed: October, 2022

[7] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert networks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 11 2019.