# Fighting Filterbubbles with Adversarial Training

Lukas Pfahler
lukas.pfahler@tu-dortmund.de
TU Dortmund University
Dortmund, Germany

Katharina Morik
katharina.morik@tu-dortmund.de
TU Dortmund University
Dortmund, Germany

## ABSTRACT

Recommender engines play a role in the emergence and reinforcement of filter bubbles. When these systems learn that a user prefers content from a particular site, the user will be less likely to be exposed to different sources or opinions and, ultimately, is more likely to develop extremist tendencies. We trace the roots of this phenomenon to the way the recommender engine represents news articles. The vectorial features modern systems extract from the plain text of news articles are already highly predictive of the associated news outlet. We propose a new training scheme based on adversarial machine learning to tackle this issue . Our preliminary experiments show that the features we can extract this way are significantly less predictive of the news outlet and thus offer the possibility to reduce the risk of manifestation of new filter bubbles.

## KEYWORDS

Recommender Systems, Adversarial Training, Deep Learning

**ACM Reference Format:**
Lukas Pfahler and Katharina Morik. 2020. Fighting Filterbubbles with Adversarial Training. In *Woodstock '18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY*. ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/1122445.1122456

## 1 INTRODUCTION

Recommender systems are a source of unwanted societal effects like filter bubbles or echo chambers, i.e. situations where users reinforce their own beliefs by only engaging with content that corresponds to their world views [4]. Typically, these automatic recommendations are based on the combination of inferred user preferences and properties of the content: The system learns which properties a user wants to see in the content he consumes. When working with texts, like news articles or social media posts, content properties are often represented by numerical features extracted with large neural network architectures. A prominent example of these neural architectures is BERT[1], which is for instance applied in Google search[1].

We investigate the space of feature vectors induced by BERT models in the domain of news articles and ask if the extraction of features is already a source of societal and cultural divides. We discover that the features are predictive of the news source and hence content recommended based on similarities is likely to come from the same source also, thereby possibly promoting the filter bubble effect. To address this issue, we propose to apply adversarial learning to compute features that are less predictive of their corresponding news source but are still informative of the semantics of the text.

We begin this extended abstract by briefly discussing the data set and the BERT model we use in this study. Then we describe our novel adversarial training procedure that trains our model, such that the vectorial features it extracts are less predictive of the news outlet. Finally, we test our new vectorial features in a small preliminary study. We discuss the results and outline the next steps towards a more thorough evaluation.

## 2 DATA SET

We use news articles from 7 news outlets[2] in English-speaking countries that were crawled from their respective homepages over the last three years[6]. We group the articles by their publication dates and use 90% of the days as training data for our embedding models. On the remaining 10%, we test and report our metrics.

## 3 BERT MODEL

For feature extraction, we rely on a DistilBERT model [5], a more resource-efficient variant of the original BERT deep network architecture. The model takes a sequence of tokens[3] $x_1, x_2, \dots x_l$ and, in a sequence of 6 transformer layers [7], produces an output sequence $z_1, z_2, \dots z_l$ where each $z_i \in \mathbb{R}^{768}$. At the training stage, the model is trained to solve masked language modeling task, where 15% of the input tokens are shadowed and we learn to predict the masked inputs $x_i$ based on the output vector $z_i$.

We use a model by HuggingFace [8] that is pretrained on the Toronto Book Corpus as well as the English Wikipedia . Then we fine-tune this pretrained model using paragraphs from our training data to adapt to the content and style of news articles. This fine-tuned model will act as the baseline in our study.

At inference time – following standard practice (e.g. [10]) – we compute embeddings of the respective news articles by computing the mean embedding $\bar{z}$ on the output sequence. The BERT model does not process full texts, but only sequences of up to length 512. To mitigate this, we process each paragraph individually, cropping excess tokens, and also average these paragraph vectors into a document vector. This way we reduce the full text, including headline, to a 768-dimensional vector that could be fed into a recommendation system.

[1] https://blog.google/products/search/search-language-understanding-bert/

[2] washingtontimes.com, washingtonpost.com, wsj.com, nytimes.com, foxnews.com, cnn.com, cbsnews.com, ft.com

[3] i.e. words or sub-words


*Woodstock '18, June 03–05, 2018, Woodstock, NY*

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

## 4 ADVERSARIAL TRAINING

We propose a machine learning model comprised of two parts, in accordance with the terminology commonly used in adversarial learning we call them generator and discriminator. The task of the generator model is to produce vectorial features of the tokenized text that contain little information about the news outlet. The discriminator tries to predict the news outlet of the text.

$$f_{\text{gen}}(x_1, \ldots, x_l) = \text{BERT}(x_1, \ldots, x_l; \Theta)$$

$$f_{\text{dis}}(z_1, \ldots, z_l) = V^T \sigma(l^{-1} \cdot \sum_{i=1}^{l} W^T z_i)$$

Both models act as adversaries: While the discriminator tries to classify feature vectors correctly by learning $V$ and $W$, the generator learns the BERT parameters $\Theta$ to produce features that allow reconstruction of masked tokens while not allowing the discriminator to correctly classify the output. This is captured in the different optimisation functions associated with the models. The generator tries to achieve a small loss on the masked-language model task while simultaneously forcing the discriminator to suffer a high classification loss:

$$L_{\text{gen}} = \ell_{\text{mask}}(f_{\text{gen}}) - \ell_{\text{classification}}(f_{\text{dis}} \circ f_{\text{gen}})).$$

On the other hand, the discriminator tries to correctly predict the news outlet by minimising its classification loss:

$$L_{\text{dis}} = \ell_{\text{classification}}(f_{\text{dis}} \circ f_{\text{gen}}).$$

We alternatingly minimize $L_{\text{gen}}$ with respect to $\Theta$ and $L_{\text{dis}}$ with respect to $V, W$. Following Goodfellow et al. [2], we perform 4 updates of the discriminator for one update of the generator. Our full training script can be found at http://tiny.cc/adversarial_training.

## 5 PRELIMINARY EXPERIMENTS

We have to investigate two hypothesis: a) The features obtained with adversarial training contain less information about the media outlet than the features from the plain model and b) the features from adversarial training are still useful to recommend related articles. In order to measure these effects we define quantitative measures. To measure how well the features predict the news outlet, we test three different machine learning models on the classification task and report estimates of their performance.
We report the prediction accuracy of

(1) 1-nearest-neighbor classification (leave-one-out validation)
(2) logistic regression (10-fold stratified cross-validation)
(3) random forest classification (10-fold stratified cross-validation)

The hyperparameters of the classifiers were optimized in a grid search. Note that for our test data, blindly guessing the news outlet would yield an accuracy of 13.50%. To measure of the adversarial features still capture semantic relatedness, we use two different similarity metrics for texts and check their scores for all nearest neighbor pairs under Cosine similarity of our feature vectors.

(1) We use pretrained Glove word embeddings and compute the Cosine similarity of the resulting document embeddings using the spacy library [3].

**Table 1: Metrics of our BERT features. The adversarial training makes the outlet less predictable by all three classifiers. Simultaneously, the similarity of nearest neighbors judged by NERs and Glove-Vectors does not suffer.**

| | Only Fine-Tuning | Adversarial Training | Relative Change |
|---|---|---|---|
| Nearest Neighbor | 25.78% | 20.03% | -22% |
| Logistic Regression | 26.90% | 20.82% | -22% |
| Random Forest | 38.54% | 28.16% | -27% |
| Glove Similarity | 0.9635 | 0.9644 | ±0% |
| Named Entities | 0.5384 | 0.5684 | +5% |

(2) We extract the named entities in the texts also using spacy [3], build one-hot-encoded entity vectors for each text and compute their Cosine similarity.

We report our findings in Table 1. We see that the baseline BERT model produces feature vectors that are very predictive of the news outlet: Using a random forest we were able to correctly predict it in 38% of the test cases. However, by using the adversarial training, we can reduce the classification performance by more than 20% for all classification models. Interestingly, the quality of the features for judging the semantic similarity does not seem to suffer. Neither of our quality measures decreases when using adversarial training.

## 6 DISCUSSION AND OUTLOOK

We were able to show that features extracted from news articles using modern deep neural networks are highly predictive of the associated news source. With our adversarial training approach we were able to significantly reduce this predictability. However, we are still far away from guessing. But this can probably not be achieved without sacrificing the usefulness of the features. For instance, an article on finance is more likely to appear in the Financial Times than other news outlets. In order to hide this fact from the recommendation engine, we have to hide substantial parts of the content, rendering the features pretty much useless.

A caveat of our study are the many hyperparameters in the BERT model, the training process, but also the metrics in our evaluation. We tried to set them according to known best-practices, but right now our results hold only for one particular configuration. Consequently, a more thorough, larger investigation is still required.

Recently, the Microsoft news team started a challenge on news recommendation [9], in the future we want to investigate our method for the intended use-case of recommendation more directly using their dataset. This way we can actually learn a recommendation engine with our feature vectors and see how the performance of the full system is affected by our training approach and if we can increase the diversity with respect to the news outlets in our recommendations.

# REFERENCES


[1] Jacob Devlin, Ming-Wei Chang, Lee Kenton, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of NAACL-HLT 2019*. Association for Computational Linguistics, 4171–4186.

[2] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in neural information processing systems*. 2672–2680.

[3] Matthew Honnibal and Ines Montani. 2017. {spaCy 2}: Natural language understanding with {B}loom embeddings, convolutional neural networks and incremental parsing. (2017).

[4] E Pariser. 2011. *The Filter Bubble: What The Internet Is Hiding From You.* Penguin Books Limited. https://books.google.de/books?id=kQs-_i9WavcC

[5] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. (2019), 2–6. http://arxiv.org/abs/1910.01108

[6] Erich Schubert, Andreas Spitz, and Michael Gertz. 2018. Exploring significant interactions in live news. *CEUR Workshop Proceedings* 2079 (2018), 39–44. http://ceur-ws.org/Vol-2079/paper9.pdf

[7] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. *Advances in Neural Information Processing Systems* (2017).

[8] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. HuggingFace's Transformers: State-of-the-art Natural Language Processing. *ArXiv* abs/1910.0 (2019).

[9] Fangzhao Wu, Ying Qiao, Jiun-Hung Chen, Chuhan Wu, Tao Qi, Jianxun Lian, Danyang Liu, Xing Xie, Jianfeng Gao, Winnie Wu, Ming Zhou, and Microsoft Research. 2020. MIND: A Large-scale Dataset for News Recommendation. *Acl* (2020), 3597–3606. https://www.kaggle.com/gspmoreira/news-portal-user-

[10] Han Xiao. 2018. bert-as-service. \url{https://github.com/hanxiao/bert-as-service}.