Localization of Fake News Detection via Multitask Transfer Learning

Jan Christian Blaise Cruz, Julianne Agatha Tan, and **Charibeth Cheng**
Center for Language Technologies, College of Computer Studies
De La Salle University, Manila
{jan_christian_cruz, julianne_tan, charibeth.cheng}@dlsu.edu.ph

Abstract

The use of the internet as a fast medium of spreading fake news reinforces the need for computational tools that combat it. Techniques that train fake news classifiers exist, but they all assume an abundance of resources including large labeled datasets and expert-curated corpora, which low-resource languages may not have. In this paper, we show that Transfer Learning (TL) can be used to train robust fake news classifiers from little data, achieving 91% accuracy on a fake news dataset in the low-resourced Filipino language, reducing the error by 14% compared to established few-shot baselines. Furthermore, lifting ideas from multitask learning, we show that augmenting transformer-based transfer techniques with auxiliary language modeling losses improves their performance by adapting to stylometry. Using this, we improve TL performance by 4-6%, achieving an accuracy of 96% on our best model. We perform ablations that establish the causality of attention-based TL techniques to state-of-the-art results, as well as the model's capability to learn and predict via stylometry. Lastly, we show that our method generalizes well to different types of news articles, including political news, entertainment news, and opinion articles. The code and datasets are publicly available at https://github.com/jcblaisecruz02/Tagalog-fake-news.

1 Introduction

There is a growing interest in research revolving around automated fake news detection and fact checking as its need increases due to the dangerous speed fake news spreads on social media (Pérez-Rosas et al., 2018). With as much as 68% of adults in the United States regularly consuming news on social media[1], being able to distinguish fake from non-fake is a pressing need.

Numerous recent studies have tackled fake news detection with various techniques. The work of Bourgonje et al. (2017) identifies and verifies the stance of a headline with respect to its content as a first step in identifying potential fake news, achieving an accuracy of 89.59% on a publicly available article stance dataset. The work of Karimi et al. (2018) uses a deep learning approach and integrates multiple sources to assign a degree of "fakeness" to an article, beating representative baselines on a publicly-available fake news dataset.

More recent approaches also incorporate newer, novel methods to aid in detection. The work of Conforti et al. (2018) handles fake news detection as a specific case of *cross-level stance detection*. In addition, their work also uses the presence of an "inverted pyramid" structure as an indicator of real news, using a neural network to encode a given article's structure.

While these approaches are valid and robust, most, if not all, modern fake news detection techniques assume the existence of large, expertly-annotated corpora to train models from scratch. Both Bourgonje et al. (2017) and Conforti et al. (2018) use the Fake News Challenge[2] dataset, with 49,972 labeled stances for each headline-body pairs. Karimi et al. (2018), on the other hand, uses the LIAR dataset (Wang, 2017), which contains 12,836 labeled short statements as well as sources to support the labels.

This requirement for large datasets to effectively train fake news detection models from scratch makes it difficult to adapt these techniques into low-resource languages. Our work focuses on the use of Transfer Learning (TL) to evade this data scarcity problem.

[1] https://www.journalism.org/2018/09/10/news-use-across-social-media-platforms-2018/

[2] http://www.fakenewschallenge.org/

We make three contributions.

First, we construct the first fake news dataset in the low-resourced Filipino language, alleviating data scarcity for research in this domain.

Second, we show that TL techniques such as ULMFiT (Howard and Ruder, 2018), BERT (Devlin et al., 2018), and GPT-2 (Radford et al., 2018, 2019) perform better compared to few-shot techniques by a considerable margin.

Third, we show that auxiliary language modeling losses (Chronopoulou et al., 2019; Liu et al., 2019) allows transformers to adapt to the stylometry of downstream tasks, which produces more robust fake news classifiers.

2 Methods

We provide a baseline model as a comparison point, using a few-shot learning-based technique to benchmark transfer learning against methods designed with low resource settings in mind. After which, we show three TL techniques that we studied and adapted to the task of fake news detection.

2.1 Baseline

We use a siamese neural network, shown to perform state-of-the-art few-shot learning (Koch et al., 2015), as our baseline model.

A siamese network is composed of weight-tied twin networks that accept distinct inputs, joined by an energy function, which computes a distance metric between the representations given by both twins. The network could then be trained to differentiate between classes in order to perform classification (Koch et al., 2015).

We modify the original to account for sequential data, with each twin composed of an embedding layer, a Long-Short Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) layer, and a feed-forward layer with Rectified Linear Unit (ReLU) activations.

Each twin embeds and computes representations for a pair of sequences, with the prediction vector p computed as:

$$\mathrm{p} = \sigma(W_{\text{out}}|o_1 - o_2| + b_{\text{out}}) \qquad (1)$$

where o_i denotes the output representation of each siamese twin i , W_{out} and b_{out} denote the weight matrix and bias of the output layer, and σ denotes the sigmoid activation function.

2.2 ULMFiT

ULMFiT (Howard and Ruder, 2018) was introduced as a TL method for Natural Language Processing (NLP) that works akin to ImageNet (Russakovsky et al., 2015) pretraining in Computer Vision.

It uses an AWD-LSTM (Merity et al., 2017) pretrained on a language modeling objective as a base model, which is then finetuned to a downstream task in two steps.

First, the language model is finetuned to the text of the target task to adapt to the task syntactically. Second, a classification layer is appended to the model and is finetuned to the classification task conservatively. During finetuning, multiple different techniques are introduced to prevent catastrophic forgetting.

ULMFiT delivers state-of-the-art performance for text classification, and is notable for being able to set comparable scores with as little as 1000 samples of data, making it attractive for use in low-resource settings (Howard and Ruder, 2018).

2.3 BERT

BERT is a Transformer-based (Vaswani et al., 2017) language model designed to pretrain "deep bidirectional representations" that can be finetuned to different tasks, with state-of-the-art results achieved in multiple language understanding benchmarks (Devlin et al., 2018).

As with all Transformers, it draws power from a mechanism called "Attention" (Luong et al., 2015), which allows the model to compute weighted importance for each token in a sequence, effectively pinpointing context reference (Vaswani et al., 2017). Precisely, we compute attention on a set of queries packed as a matrix Q on key and value matrices K and V, respectively, as:

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V \qquad (2)$$

where d_k is the dimensions of the key matrix K. Attention allows the Transformer to refer to multiple positions in a sequence for context at any given time regardless of distance, which is an advantage over Recurrent Neural Networks (RNN).

BERT's advantage over ULMFiT is its bidirectionality, leveraging both left and right context using a pretraining method called "Masked Language Modeling." In addition, BERT also benefits from being *deep*, allowing it to capture more context and information. BERT-Base, the smallest BERT model, has 12 layers (768 units in each hidden layer) and 12 attention heads for a total of 110M parameters. Its larger sibling, BERT-Large, has 24 layers (1024 units in each hidden layer) and 16 attention heads for a total of 340M parameters.

2.4 GPT-2

The GPT-2 (Radford et al., 2019) technique builds up from the original GPT (Radford et al., 2018). Its main contribution is the way it is trained. With an improved architecture, it learns to do multiple tasks by just training on vanilla language modeling.

Architecture-wise, it is a Transformer-based model similar to BERT, with a few differences. It uses two feed-forward layers per transformer "block," in addition to using "delayed residuals" which allows the model to choose which transformed representations to output.

GPT-2 is notable for being *extremely deep*, with 1.5B parameters, 10x more than the original GPT architecture. This gives it more flexibility in learning tasks unsupervised from language modeling, especially when trained on a very large unlabeled corpus.

2.5 Multitask Finetuning

BERT and GPT-2 both lack an explicit "language model finetuning step," which gives ULMFiT an advantage where it learns to adapt to the stylometry and linguistic features of the text used by its target task. Motivated by this, we propose to augment Transformer-based TL techniques with a language model finetuning step.

Motivated by recent advancements in multitask learning, we finetune the model to the stylometry of the target task *at the same time* as we finetune the classifier, instead of setting it as a separate step. This produces two losses to be optimized *together* during training, and ensures that no task (stylometric adaptation or classification) will be prioritized over the other. This concept has been proposed and explored to improve the performance of transfer learning in multiple language tasks (Chronopoulou et al., 2019; Liu et al., 2019).

We show that this method improves performance on both BERT and GPT-2, given that it learns to adapt to the idiosyncracies of its target task in a similar way that ULMFiT also does.

3 Experimental Setup

3.1 Fake News Dataset

We work with a dataset composed of 3,206 news articles, each labeled *real* or *fake*, with a perfect 50/50 split between 1,603 real and fake articles, respectively. *Fake* articles were sourced from online sites that were tagged as *fake news sites* by the non-profit independent media fact-checking organization Verafiles[3] and the National Union of Journalists in the Philippines (NUJP)[4]. *Real* articles were sourced from mainstream news websites in the Philippines, including Pilipino Star Ngayon[5], Abante[6], and Bandera[7].

For preprocessing, we only perform tokenization on our dataset, specifically "Byte-Pair Encoding" (BPE) (Cherry et al., 2018). BPE is a form of fixed-vocabulary *subword tokenization* that considers *subword units* as the most primitive form of entity (i.e. a *token*) instead of canonical words (i.e. "I am walking today" → "I am walk ##ing to ##day"). BPE is useful as it allows our model to represent out-of-vocabulary (OOV) words unlike standard tokenization. In addition, it helps language models in learning morphologically-rich languages as it now treats morphemes as primary enitites instead of canonical word tokens.

For training/finetuning the classifiers, we use a 70%-30% train-test split of the dataset.

3.2 Pretraining Corpora

To pretrain BERT and GPT-2 language models, as well as an AWD-LSTM language model for use in ULMFiT, a large unlabeled training corpora is needed. For this purpose, we construct a corpus of 172,815 articles from Tagalog Wikipedia[8] which we call *WikiText-TL-39* (Cruz and Cheng, 2019). We form training-validation-test splits of 70%-15%-15% from this corpora.

Preprocessing is similar to the fake news dataset, with the corpus only being lightly preprocessed and tokenized using Byte-Pair Encoding.

Corpus statistics for the pretraining corpora are shown on table 1.

[3] verafiles.org

[4] https://nujp.org/

[5] https://www.philstar.com/pilipino-star-ngayon

[6] https://www.abante.com.ph

[7] https://bandera.inquirer.net/balita

[8] https://tl.wikipedia.org/wiki/Unang_Pahina

Split	Documents	Tokens	Unique Tokens	Num. of Lines
Training	120,975	39,267,089	279,153	1,403,147
Validation	25,919	8,356,898	164,159	304,006
Testing	25,921	8,333,288	175,999	298,974
OOV Tokens	28,469 (0.1020%)			

Table 1: Statistics for the WikiText-TL-39 Dataset.

3.3 Siamese Network Training

We train a siamese recurrent neural network as our baseline. For each twin, we use 300 dimensions for the embedding layer and a hidden size of 512 for all hidden state vectors.

To optimize the network, we use a regularized cross-entropy objective of the following form:

$$\begin{aligned}\mathcal{L}(x_1, x_2) = \mathrm{y}(x_1, x_2) \log \mathrm{p}(x_1, x_2) + \\ (1 - \mathrm{y}(x_1, x_2)) \log(1 - \mathrm{p}(x_1, x_2)) + \lambda |w|^2\end{aligned} \quad (3)$$

where $\mathrm{y}(x_1, x_2) = 1$ when x_1 and x_2 are from the same class and 0 otherwise. We use the Adam optimizer (Kingma and Ba, 2014) with an initial learning rate of 1e-4 to train the network for a maximum of 500 epochs.

3.4 Transfer Pretraining

We pretrain a cased BERT-Base model using our prepared unlabeled text corpora using Google's provided pretraining scripts[9]. For the masked language model pretraining objective, we use a 0.15 probability of a word being masked. We also set the maximum number of masked language model predictions to 20, and a maximum sequence length of 512. For training, we use a learning rate of 1e-4 and a batch size of 256. We train the model for 1,000,000 steps with 10,000 steps of learning rate warmup for 157 hours on a Google Cloud Tensor processing Unit (TPU) v3-8.

For GPT-2, we pretrain a GPT-2 Transformer model on our prepared text corpora using language modeling as its sole pretraining task, according to the specifications of (Radford et al., 2019). We use an embedding dimension of 410, a hidden dimension of 2100, and a maximum sequence length of 256. We use 10 attention heads per multihead attention block, with 16 blocks composing the encoder of the transformer. We use dropout on all linear layers to a probability of 0.1. We initialize all parameters to a standard deviation of 0.02. For training, we use a learning rate of 2.5e-4, and a batch size of 32, much smaller than BERT considering the large size of the model. We train the model for 200 epochs with 1,000 steps of learning rate warmup using the Adam optimizer. The model was pretrained for 178 hours on a machine with one NVIDIA Tesla V100 GPU.

For ULMFiT, we pretrain a 3-layer AWD-LSTM model with an embedding size of 400 and a hidden size of 1150. We set the dropout values for the embedding, the RNN input, the hidden-to-hidden transition, and the RNN output to (0.1, 0.3, 0.3, 0.4) respectively. We use a weight dropout of 0.5 on the LSTMs recurrent weight matrices. The model was trained for 30 epochs with a learning rate of 1e-3, a batch size of 128, and a weight decay of 0.1. We use the Adam optimizer and use slanted triangular learning rate schedules (Howard and Ruder, 2018). We train the model on a machine with one NVIDIA Tesla V100 GPU for a total of 11 hours.

For each pretraining scheme, we checkpoint models every epoch to preserve a copy of the weights such that we may restore them once the model starts overfitting. This is done as an extra regularization technique.

3.5 Finetuning

We finetune our models to the target fake news classification task using the pretrained weights with an appended classification layer or *head*.

For BERT, we append a classification head composed of a single linear layer followed by a softmax transformation to the transformer model. We then finetune our BERT-Base model on the fake news classification task for 3 epochs, using a batch size of 32, and a learning rate of 2e-5.

For GPT-2, our classification head is first comprised of a layer normalization transform, followed by a linear layer, then a softmax transform. We finetune the pretrained GPT-2 transformer for 3 epochs, using a batch size of 32, and a learning rate of 3e-5.

For ULMFiT, we perform language model fine-

[9] https://github.com/google-research/bert

tuning on the fake news dataset (appending no extra classification heads yet) for a total of 10 epochs, using a learning rate of 1e-2, a batch size of 80, and weight decay of 0.3. For the final ULMFiT finetuning stage, we append a compound classification head (linear → batch normalization → ReLU → linear → batch normalization → softmax). We then finetune for 5 epochs, gradually unfreezing layers from the last to the first until all layers are unfrozen on the fourth epoch. We use a learning rate of 1e-2 and set Adam's α and β parameters to 0.8 and 0.7, respectively.

To show the efficacy of Multitask Finetuning, we augment BERT and GPT-2 to use this finetuning setup with their classification heads. We finetune both models to the target task for 3 epochs, using a batch size of 32, and a learning rate of 3e-5. For optimization, we use Adam with a warmup steps of 10% the number of steps, comprising 3 epochs.

3.6 Generalizability Across Domains

To study the generalizability of the model to different news domains, we test our models against test cases not found in the training dataset. We mainly focus on three domains: political news, opinion articles, and entertainment/gossip articles. Articles used for testing are sourced from the same websites that the training dataset was taken from.

4 Results and Discussion

4.1 Classification Results

Our baseline model, the siamese recurrent network, achieved an accuracy of 77.42% on the test set of the fake news classification task.

The transfer learning methods gave comparable scores. BERT finetuned to a final 87.47% accuracy, a 10.05% improvement over the siamese network's performance. GPT-2 finetuned to a final accuracy of 90.99%, a 13.57% improvement from the baseline performance. ULMFiT finetuning gave a final accuracy of 91.59%, an improvement of 14.17% over the baseline Siamese Network.

We could see that TL techniques outperformed the siamese network baseline, which we hypothesize is due to the intact pretrained knowledge in the language models used to finetune the classifiers. The pretraining step aided the model in forming relationships between text, and thus, performed better at stylometric based tasks with little finetuning.

The model results are all summarized in table 2.

4.2 Language Model Finetuning Significance

One of the most surprising results is that BERT and GPT-2 performed worse than ULMFiT in the fake news classification task despite being deeper models capable of more complex relationships between data.

We hypothesize that ULMFiT achieved better accuracy because of its additional language model finetuning step. We provide evidence for this assumption with an additional experiment that shows a decrease in performance when the language model finetuning step is removed, droppping ULMFiT's accuracy to 78.11%, making it only perform marginally better than the baseline model. Results for this experiment are outlined in Table 3

In this finetuning stage, the model is said to "adapt to the idiosyncracies of the task it is solving" (Howard and Ruder, 2018). Given that our techniques rely on linguistic cues and features to make accurate predictions, having the model adapt to the stylometry or "writing style" of an article will therefore improve performance.

4.3 Multitask-based Finetuning

We used a multitask finetuning technique over the standard finetuning steps for BERT and GPT-2, motivated by the advantage that language model finetuning provides to ULMFiT, and found that it greatly improves the performance of our models.

BERT achieved a final accuracy of 91.20%, now marginally comparable to ULMFiT's full performance. GPT-2, on the other hand, finetuned to a final accuracy of 96.28%, a full 4.69% improvement over the performance of ULMFiT. This provides evidence towards our hypothesis that a language model finetuning step will allow transformer-based TL techniques to perform better, given their inherent advantage in modeling complexity over more shallow models such as the AWD-LSTM used by ULMFiT. Rersults for this experiment are outlined in Table 4.

5 Ablation Studies

Several ablation studies are performed to establish causation between the model architectures and the performance boosts in the study.

Model	Val. Accuracy	Loss	Val. Loss	Pretraining Time	Finetuning Time
Siamese Networks	77.42%	0.5601	0.5329	N/A	4m per epoch
BERT	87.47%	0.4655	0.4419	66 hours	2m per epoch
GPT-2	90.99%	0.2172	0.1826	78 hours	4m per epoch
ULMFiT	**91.59%**	**0.3750**	**0.1972**	**11 hours**	**2m per epoch**

Table 2: Final model results. Pretraining time refers to the number of hours the model took to finish the pretraining objective (masked-language modeling and next-sentence prediction for BERT, and language modeling for GPT-2 and ULMFiT (AWD-LSTM), respectively. Finetuning time refers to minutes per epoch. BERT and GPT-2 were finetuned for 3 epochs, while ULMFiT was finetuned for 5.

Model	Val. Accuracy
With LM Finetuning	91.59%
Without LM Finetuning	78.11%

Table 3: ULMFiT results with and without language model finetuning. Removing the language model finetuning step shows a significant drop in performance, giving evidence to the hypothesis that such a step improves the model by adapting to its stylometry.

Model	Accuracy	Loss	Val. Loss
ULMFiT	91.59	0.3750	0.1972
BERT	91.20%	0.3115	0.3023
GPT-2	**96.28%**	**0.2609**	**0.2197**

Table 4: ULMFiT compared to transfer learning techniques augmented with multitask finetuning. Including a language modeling finetuning task to the transformer-based transfer learning techniques improved their performance, with GPT-2 outperforming ULMFiT by 4.69%. "Val. Accuracy" in this table refers to validation accuracy at test time.

5.1 Pretraining Effects

An ablation on pretraining was done to establish evidence that pretraining before finetuning accounts for a significant boost in performance over the baseline model. Using non-pretrained models, we finetune for the fake news classification task using the same settings as in the prior experiments.

In Table 5, it can be seen that generative pretraining via language modeling does account for a considerable amount of performance, constituting 44.32% of the overall performance (a boost of 42.67% in accuracy) in the multitasking setup, and constituting 43.93% of the overall performance (a boost of 39.97%) in the standard finetuning setup.

This provides evidence that the pretraining step is necessary in achieving state-of-the-art performance.

5.2 Attention Head Effects

An ablation study was done to establish causality between the multiheaded nature of the attention mechanisms and state-of-the-art performance. We posit that since the model can refer to multiple context points at once, it improves in performance.

For this experiment, we performed several pretraining-finetuning setups with varied numbers of attention heads using the multitask-based finetuning scheme. Using a pretrained GPT-2 model, attention heads were masked with zero-tensors to downsample the number of positions the model could attend to at one time.

As shown in Table 6, reducing the number of attention heads severely decreases multitasking performance. Using only one attention head, thereby attending to only one context position at once, degrades the performance to less than the performance of 10 heads using the standard finetuning scheme. This shows that more attention heads, thereby attending to multiple different contexts at once, is important to boosting performance to state-of-the-art results.

While increasing the number of attention heads improves performance, keeping on adding extra heads will not result to an equivalent boost as the performance plateaus after a number of heads.

As shown in Figure 1, the performance boost of the model plateaus after 10 attention heads, which was the default used in the study. While the performance of 16 heads is greater than 10, it is only a marginal improvement, and does not justify the added costs to training with more attention heads.

6 Stylometric Tests

To supplement our understanding of the features our models learn and establish empirical difference in their stylometries, we use two stylometric tests traditionally used for authorship attribution: Mendenhall's Characteristic Curves (Mendenhall,

Finetuning	Pretrained?	Accuracy	Val. Loss	Acc. Inc.	% of Perf.
Multitasking	No	53.61%	0.7217	-	-
	Yes	96.28%	0.2197	+42.67%	44.32%
Standard	No	51.02%	0.7024	-	-
	Yes	90.99%	0.1826	+39.97%	43.93%

Table 5: An ablation study on the effects of pretraining for multitasking-based and standard GPT-2 finetuning. Results show that pretraining greatly accounts for almost half of performance on both finetuning techniques. "Acc. Inc." refers to the boost in performance contributed by the pretraining step. "% of Perf." refers to the percentage of the total performance that the pretraining step contributes.

# of Heads	Accuracy	Val. Loss	Effect
1	89.44	0.2811	-6.84%
2	91.20%	0.2692	-5.08%
4	93.85%	0.2481	-2.43%
8	96.02%	0.2257	-0.26%
10	96.28%	0.2197	
16	96.32%	0.2190	+0.04

Table 6: An ablation study on the effect of multiple heads in the attention mechanisms. The results show that increasing the number of heads improves performance, though this plateaus at 10 attention heads. All ablations use the multitask-based finetuning method. "Effect" refers to the increase or decrease of accuracy as the heads are removed. Note that 10 heads is the default used throughout the study.

1887) and John Burrow's Delta Method (Burrows, 2002).

We provide a characteristic curve comparison to establish differences between real and fake news. For the rest of this section, we refer to the characteristic curves on Figure 2.

When looking at the y-axis, there is a big difference in word count. The fake news corpora has twice the amount of words as the real news corpora. This means that fake news articles are at average lengthier than real news articles. The only differences seen in the x-axis is the order of appearance of word lengths 6, 7, and 1. The characteristic curves also exhibit differences in trend. While the head and tail look similar, the body show different trends. When graphing the corpora by news category, the heads and tails look similar to the general real and fake news characteristic curve but the body exhibits a trend different from the general corpora. This difference in trend may be attributed to either a lack of text data to properly represent real and fake news or the existence of a stylistic difference between real and fake news.

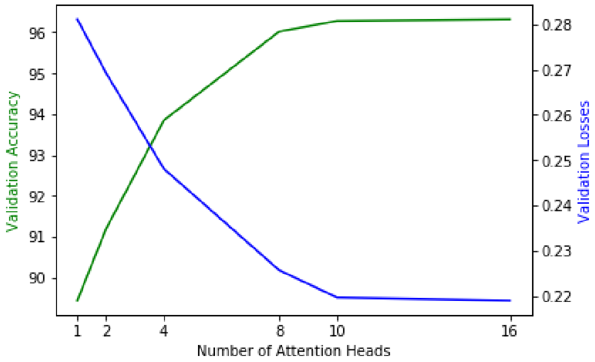

Figure 1: Ablation showing accuracy and loss curves with respect to attention heads.

We also use Burrows Delta method to see a numeric distance between text samples. Using the labeled news article corpora, we compare samples outside of the corpora towards real and fake news to see how similar they are in terms of vocabulary distance. The test produces smaller distance for the correct label, which further reaffirms our hypothesis that there is a stylistic difference between the labels. However, the difference in distance between real and fake news against the sample is not significantly large. For articles on politics, business, entertainment, and viral events, the test generates distances that are significant. Meanwhile news in the safety, sports, technology, infrastructure, educational, and health categories have negligible differences in distance. This suggests that some categories are written similarly despite veracity.

7 Further Discussions

7.1 Pretraining Tasks

All the TL techniques were pretrained with a language modeling-based task. While language modeling has been empirically proven as a good pretraining task, we surmise that other pretraining

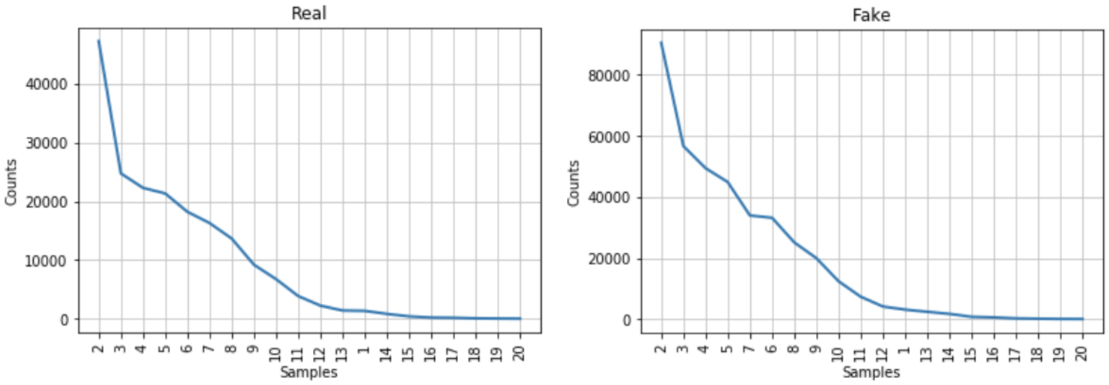

Figure 2: Comparison of the characteristic curves of fake news and real news.

tasks could replace or support it.

Since automatic fake news detection uses stylometric information (i.e. writing style, language cues), we predict that the task could benefit from pretraining objectives that also learn stylometric information such as authorship attribution.

7.2 Generalizability Across Domains

When testing on three different types of articles (Political News, Opinion, Entertainment/Gossip), we find that writing style is a prominent indicator for fake articles, supporting previous findings regarding writing style in fake news detection (Potthast et al., 2018).

Supported by our findings on the stylometric differences of fake and real news, we show that the model predicts a label based on the test article's stylometry. It produces correct labels when tested on real and fake news.

We provide further evidence that the models learn stylometry by testing on out-of-domain articles, particularly opinion and gossip articles. While these articles aren't necessarily real or fake, their stylometries are akin to real and fake articles respectively, and so are classified as such.

8 Conclusion

In this paper, we show that TL techniques can be used to train robust fake news classifiers in low-resource settings, with TL methods performing better than few-shot techniques, despite being a setting they are designed in mind with.

We also show the significance of language model finetuning for tasks that involve stylometric cues, with ULMFiT performing better than transformer-based techniques with deeper language model backbones. Motivated by this, we augment the methodology with a multitask learning-inspired finetuning technique that allowed transformer-based transfer learning techniques to adapt to the stylometry of a target task, much like ULMFiT, resulting in better performance.

For future work, we propose that more pretraining tasks be explored, particularly ones that learn stylometric information inherently (such as authorship attribution).

Acknowledgments

The authors would like to acknowledge the efforts of VeraFiles and the National Union of Journalists in the Philippines (NUJP) for their work covering and combating the spread of fake news.

We are partially supported by Google's Tensoflow Research Cloud (TFRC) program. Access to the TPU units provided by the program allowed the BERT models in this paper, as well as the countless experiments that brought it to fruition, possible.

References

Peter Bourgonje, Julian Moreno Schneider, and Georg Rehm. 2017. From clickbait to fake news detection: An approach based on detecting the stance of headlines to articles. In *Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism*, pages 84–89, Copenhagen, Denmark. Association for Computational Linguistics.

John Burrows. 2002. delta: a measure of stylistic difference and a guide to likely authorship. *Literary and linguistic computing*, 17(3):267–287.

Colin Cherry, George Foster, Ankur Bapna, Orhan Firat, and Wolfgang Macherey. 2018. Revisiting character-based neural machine translation with capacity and compression. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4295–4305, Brussels, Belgium. Association for Computational Linguistics.

Alexandra Chronopoulou, Christos Baziotis, and Alexandros Potamianos. 2019. An embarrassingly simple approach for transfer learning from pretrained language models. *arXiv preprint arXiv:1902.10547*.

Costanza Conforti, Mohammad Taher Pilehvar, and Nigel Collier. 2018. Towards automatic fake news detection: Cross-level stance detection in news articles. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 40–49, Brussels, Belgium. Association for Computational Linguistics.

Jan Christian Blaise Cruz and Charibeth Cheng. 2019. Evaluating language model finetuning techniques for low-resource languages. *arXiv preprint arXiv:1907.00409*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. Association for Computational Linguistics.

Hamid Karimi, Proteek Roy, Sari Saba-Sadiya, and Jiliang Tang. 2018. Multi-source multi-class fake news detection. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1546–1557, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. 2015. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-task deep neural networks for natural language understanding. *arXiv preprint arXiv:1901.11504*.

Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. *CoRR*, abs/1508.04025.

Thomas Corwin Mendenhall. 1887. The characteristic curves of composition. *Science*, 9(214):237–249.

Stephen Merity, Nitish Shirish Keskar, and Richard Socher. 2017. Regularizing and Optimizing LSTM Language Models. *arXiv preprint arXiv:1708.02182*.

Verónica Pérez-Rosas, Bennett Kleinberg, Alexandra Lefevre, and Rada Mihalcea. 2018. Automatic detection of fake news. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3391–3401, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Martin Potthast, Johannes Kiesel, Kevin Reinartz, Janek Bevendorff, and Benno Stein. 2018. A stylometric inquiry into hyperpartisan and fake news. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 231–240, Melbourne, Australia. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. *URL https://s3-us-west-2. amazonaws. com/openai-assets/research-covers/languageunsupervised/language understanding paper. pdf*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8).

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

William Yang Wang. 2017. ”liar, liar pants on fire”: A new benchmark dataset for fake news detection. *CoRR*, abs/1705.00648.