The Surprising Performance of Simple Baselines for Misinformation Detection

Kellin Pelrine[*]
kellin.pelrine@mila.quebec
SOCS, McGill University
Mila - Quebec AI Institute

Jacob Danovitch[*]
jacob.danovitch@mila.quebec
SOCS, McGill University
Mila - Quebec AI Institute

Reihaneh Rabbany
reihaneh.rabbany@mila.quebec
SOCS, McGill University
Mila - Quebec AI Institute

ABSTRACT

As social media becomes increasingly prominent in our day to day lives, it is increasingly important to detect informative content and prevent the spread of disinformation and unverified rumours. While many sophisticated and successful models have been proposed in the literature, they are often compared with older NLP baselines such as SVMs, CNNs, and LSTMs. In this paper, we examine the performance of a broad set of modern transformer-based language models and show that with basic fine-tuning, these models are competitive with and can even significantly outperform recently proposed state-of-the-art methods. We present our framework as a baseline for creating and evaluating new methods for misinformation detection. We further study a comprehensive set of benchmark datasets, and discuss potential data leakage and the need for careful design of the experiments and understanding of datasets to account for confounding variables. As an extreme case example, we show that classifying only based on the first three digits of tweet ids, which contain information on the date, gives state-of-the-art performance on a commonly used benchmark dataset for fake news detection –Twitter16. We provide a simple tool to detect this problem and suggest steps to mitigate it in future datasets.

CCS CONCEPTS

• **Computing methodologies → Artificial intelligence**; **Natural language processing**;

KEYWORDS

misinformation, social media, natural language processing, datasets, COVID-19

ACM Reference Format:
Kellin Pelrine, Jacob Danovitch, and Reihaneh Rabbany. 2021. The Surprising Performance of Simple Baselines for Misinformation Detection. In *The Web Conference '21.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3442381.3450111

1 INTRODUCTION

Social media is filled with both treasure troves and landmines of information. There are far too many interactions to mine them by hand. But failing to understand them can have dire consequences. Misinformation can challenge fair elections [54] and cost billions of dollars [73]. Misinformation spread in the COVID-19 "infodemic" [18, 69] can cost lives. At the same time, successful mining of useful information can enable contact tracing and other measures to save lives [46, 70], as well as other applications like explainable fact checking [20]. Motivated by the profound impact, there has been substantial research on detecting fake news, misinformation, and related topics (such as bot and troll detection) in the past few years [7, 10, 59, 63, 80]. A majority of these works are based on text/content classification [77, 81, 100, 101]. As real-world events such as the spread of COVID-19 misinformation show [18, 69], we are still far from addressing misinformation and more work is needed.

In this paper, we examine the use of modern pre-trained language models (LMs) for classifying content. Although language models are ubiquitous baselines in this domain, the ones used are often older models such as SVMs, CNNs, LSTMs, etc. [15, 23, 28, 78], instead of transformer-based models that have been dominant in NLP recently, such as the exemplar BERT model [24, 98]. Here, we report the performance of these recently proposed language models on common benchmark datasets for misinformation detection. More specifically, we consider small, medium and large models and show that most can achieve comparable or even better performance than state-of-the-art (SOTA) methods. Even the smallest 4 million parameter BERT-Tiny [88] can approach or beat state-of-the-art in some cases. These SOTA methods are usually much more complex and in many cases incorporate more information beyond the content, e.g. the reply thread. Although some of these more sophisticated methods are also designed to incorporate language models, it is not straightforward how to change their language module and they may or may not synergize well with the latest language models.

We further study commonly used benchmark datasets and report a significant issue with the Twitter15 and Twitter16 datasets [47, 51, 52], which can produce unrealistically high performance just by inferring the tweet date. This issue is also present to a lesser degree in FakeNewsNet [79]. We propose a simple test to identify it in other datasets and suggest how to mitigate it in data collection.

We also provide the exact splits to use our results as benchmarks, as well as other suggestions on proper evaluation techniques, and discuss potential future directions of work in this domain.

To summarize, the main contributions of this paper are threefold:

- We show that recent pretrained LMs are competitive with and sometimes even substantially better than SOTA models in common benchmarks for misinformation detection
- We discuss experiment design considerations, effect of different data splits, and further demonstrate potential data leakage issues on commonly used benchmark datasets

[*] Equal Contribution

Submitted to the Web Conference '21, Apr 2021,

ACM ISBN 978-1-4503-XXXX-X/18/06.
https://doi.org/10.1145/3442381.3450111

- We package our framework as a simple yet strong baseline for relevant tasks, and provide the tweet IDs to exactly reconstruct our splits, facilitating use of our reported results for benchmarks. Our framework is available on GitHub.

2 BACKGROUND AND RELATED WORK

There are two key literatures related to this work: (mis-)information classification on the application domain side, and natural language processing models on the methodological side. We discuss first the former and then the latter.

2.1 Detecting Misinformation

We can broadly divide misinformation detection algorithms into methods that analyze content and (social) context. Here, we focus on the methods that are content-based. For more complete review of the related works, please refer to the recent surveys [7, 10, 59, 77, 80, 101]. The content has been analyzed previously on multiple levels. At the word level, methods have used features including word counts, derived measures like TF-IDF, small combinations of words like bigrams, and word embeddings [5, 11, 31, 75]. At the sentence level, methods rely on features like syntax and complexity [5, 11, 31, 75]. Finally, there are features derived from the text as a whole, including general representations and more specialized ones such as topic or sentiment analysis [15, 21, 35, 36].

More specifically, we consider the following state-of-the-art algorithms as a representative set of the current methods:

- Cheng et al. [15] use a multitask (rumor detection, tracking, veracity classification, and stance classification) LSTM-based VAE [40], with the text as input.
- Han et al. [28] use a GNN [76] to encode the propagation network and features of the users in it (not including text).
- Huang et al. [30] build a heterogeneous tweet-word-user graph, linking the representations with an attention mechanism.
- Lu and Li [50] use a co-attention mechanism to combine text and propagation information.
- Shu et al. [78] use attention mechanisms to combine word- and sentence-level features from articles, encoded with a GRU, with a comment (tweet) encoder.
- Wang et al. [92] link text and object detections in images with entities and a knowledge graph constructed from them.
- Wu et al. [96] uses a combination of a decision tree model and co-attention to evaluate credibility of tweets using text, propagation, and user information.
- Wu et al. [97] build a propagation graph from tweets and replies. They embed each using Doc2Vec [45], and use a GRU (Gated Recurrent Unit) [17] and attention mechanism to pass information and pool the graph.

The most related work to ours is [32]. They highlight "cross-source" failure of existing misinformation detection methods, in terms of training on one dataset and testing on another, and also propose a new model designed to solve it. Our results support and expand on theirs by identifying "cross-topic" and "cross-domain" effects even within the same datasets, depending on how they are split, as well as differences between datasets. We take a different direction for modeling: rather than focusing on cross-source tasks and developing a specialized model, we look at a spectrum of tasks and ask how general language models perform in this domain.

2.2 Detecting Informative Content

Instead of detecting which tweets are spreading misinformation, there are also many contexts in which we want detect tweets that provide useful information. Recently, there has been a big focus in the research community on COVID-19 [8]. Mining information about disaster events is not new [1, 13, 34, 38, 43, 84–86]. But COVID-19 is unprecedented in its global scale and impact, and requires new methods and understanding.

Although the focus of our experiments is on misinformation, we include a dataset for detecting informative COVID-19 tweets - discussed in more detail in section 3.2 - to highlight the connections between these two areas and how language models can model both as content classification.

2.3 Pre-Trained Language Models

The invention of word embeddings was a fundamental breakthrough [55, 56] leading towards today's language models. By predicting words, one can go beyond one-hot and similar encodings and convert words into vectors that encode their meaning and are more useful for downstream tasks.

The first embedding models such as Word2Vec [55, 56], GloVe [66], and FastText [37] generate fixed vectors for each input word or token. This has the advantage of simplicity, but can struggle to capture words whose meanings vary depending on the context, and in turn struggle with downstream tasks that require this level of language understanding. Therefore, better methods to incorporate context became a key research topic. A followup direction was to use convolutional neural networks, ubiquitous in computer vision, on text [39]. These could effectively capture local context, but still missed incorporating longer range context. Similarly, RNNs, which could capture local sequences, had problems with vanishing gradients and could not capture longer range information [57].

LSTMs [29], which like RNNs operate on sequences but allow longer range dependencies, helped solve this issue. There are state of the art models for misinformation detection, including the ones discussed in the domain section above, which use various versions or improvements on LSTMs, such as the GRU by Cho et al. [17]. These also led to contextualized embeddings such as ELMo [68].

Another major breakthrough came with attention [4, 99] and then, based on attention, transformers [89]. These facilitate far more parallelization than LSTMs [57], which means models based on them can be trained on much larger and more general corpora. A key early model using these is BERT [24] (discussed further in the following section), and there has subsequently been numerous different models proposed based on transformers [98]. These vary from small tweaks to existing models such as pretraining on domain-specific data [60, 62], to significantly different models such as GPT (and then GPT-2 and GPT-3) [9, 71, 72]. Virtually all of these models can provide contextualized word/sentence/document embeddings for downstream tasks. These models are easily accessible through packages such as [93], and with hundreds of versions to choose from, as Xia et al. [98] suggests the hardest question may be which one to use.

Although there are many excellent options, a large number of recent papers in the misinformation domain still benchmark their methods against older models [2, 15, 20, 23, 28, 78, 95]. While there are still contexts in which these models do well and there is nothing wrong with making these comparisons, our results show that this can sometimes result in weak benchmarks when the more modern language models are omitted. Among other results, we give suggestions on how to incorporate these new models into evaluation in this domain. Please note that although training these newest models from scratch can require a prohibitive amount of computational resources, pretrained weights are readily available and are commonly used instead.

3 METHODOLOGY

Our baselines simply consist of a set of language models and benchmark datasets. Here we first explain the language models we consider in our baseline framework in Section 3.1, then we describe the benchmark datasets in Section 3.2. For the Funnel Transformer and ALBERT, we employ a mean pooling strategy over the final layer of output. For ELMo we use an LSTM pooler. For all others, we use the [CLS] token embedding from the final layer of output. For all models, we allow all parameters of the language model to be fine-tuned, use a single fully-connected layer on the pooled embeddings for classification, train using cross-entropy loss, and use AdamW [49] with a slanted triangular learning rate scheduler for optimization.

3.1 Language Models

We evaluate a variety of language models. Where available, we use the implementations and pre-trained weights provided by the Huggingface Transformers library [93], and train using PyTorch and PyTorch-Lightning [25, 64]. We include the following language models (with the exact version[1] in parentheses):

BERT (bert-base-uncased) [24].
Bidirectional Encoder Representations from Transformers, or BERT for short, is a large pre-trained bidirectional transformer. BERT was pre-trained using two objectives. The first, masked language modelling, required BERT to predict a masked token from the input. The second, next sentence prediction, required predicting whether two sentences appeared consecutively in the training corpus. To complete the latter task, BERT prepends the input text with a special [CLS] token, and inserts a special [SEP] token between the first and second sentence as well as at the end of the second sentence. The [CLS] token is commonly used as a document-level representation for classification tasks.

BERT-Tiny (bert_uncased_L-2_H-128_A-2) [88].
BERT-Tiny is the smallest of several pre-trained language models that follow the same pre-training procedure as BERT, as well as a knowledge distillation fine-tuning procedure. These smaller models achieve strong performance on downstream tasks with significantly fewer parameters.

RoBERTa (roberta-large) [48].
RoBERTa follows a similar architecture to BERT but removes the next-sentence prediction pre-training objective while making the masking procedure dynamic by regenerating the mask for each example every time and leverage larger batch sizes and increased training iterations to improve performance. They find that BERT underfits its training data.

ALBERT (albert-large-v2) [44].
ALBERT adds an additional pre-training objective while incorporating two methods that reduce the number of parameters in the model, factorizing the embedding layer and tying weights across hidden layers.

BERTweet (bertweet-base) [62].
BERTweet follows an identical training procedure to RoBERTa, and is pre-trained on 850 million tweets.

COVID-Twitter-BERT (covid-twitter-bert-v2) [60].
COVID-Twitter-BERT (CT-BERT) follows an identical training procedure to BERT, and the latest version is pre-trained on 1.2 billion training examples generated from 97 million tweets.

DeCLUTR (declutr-base) [26].
DeCLUTR is a transformer-based language model that proposes a contrastive, self-supervised method for learning general purpose sentence embeddings. The model is trained with a masked language modelling objective as well as with contrastive loss using both easy and hard negatives.

Funnel Transformer (xlarge-base) [22].
The Funnel Transformer improves the efficiency of bidirectional transformer models by applying a pooling operation after each layer, akin to convolutional neural networks, to reduce the length of the input.

ELMo (Original[2]) [68].
ELMo is one of the first large-scale pre-trained language models. It is a character-based model and is the only model in this list that does not optimize for masked language modelling. Instead, it uses bidirectional LSTMs [29] to perform autoregressive language modelling in both the forward and backward directions.

3.2 Benchmark Datasets

We consider the comprehensive set of benchmark datasets available in the literature for misinformation (information) detection. In the following we briefly explain each dataset.

3.2.1 PHEME. The PHEME dataset [41, 102] contains 6425 tweets about 9 newsworthy events. The events are unrelated. The tweets were collected as the stories developed, with a journalist annotating them as rumour vs. non-rumour. They grouped them by story, and then marked whether the stories were true or false once it was confirmed, or as unverified if they could not be certain during the collection period. There are also other labels such as stance, which we do not consider here.

We split this dataset five different ways to compare with different state-of-the-art results that use the corresponding settings.

PHEME9 T/F which matches Wu and Rao [94].
First, we take only the tweets marked as true or false and do a 70-10-20 train-dev-test split.

PHEME5 R/NR which matches Wang et al. [92].
Next we take the 5 largest events (comprising 90% of the dataset) and use the rumour and non-rumour labels. We again split 70-10-20.

[1] As available at https://huggingface.co/models

[2] Available at https://allennlp.org/elmo

This setup matches both [92] (70-30 split with no dev set) and [16] (which notes 10% withheld for validation).

PHEME5 3-way which matches Cheng et al. [16].
Here, we take the 5 largest events and tweets labeled true, false, and unverified, splitting 70-10-20.

PHEME9 4-way which matches Wu et al. [97].
For this, we subsample 800, 400, 600, 500 tweets with at least 3 replies, that are non-rumor, false, true, and unverified, respectively. They are split 80-10-10.

PHEME5 Lc which matches Cheng et al. [16].
Finally, we also examine an event-based split. Starting with the 5 largest events, we train on four and test on the largest (tweets related to Charlie Hebdo terror attack), with the 3-way true, false, and unverified labels.

3.2.2 ***FakeNewsNet.***

The FakeNewsNet dataset [79, 81, 82] contains articles fact-checked by PolitiFact[3] or GossipCop[4], and related tweets. The labels are "real" and "fake." We retrieve 467 thousand tweets in the **PolitiFact** dataset, and 1.25 million in **GossipCop**. We split this dataset by first splitting the articles, then assigning tweets to each split based on the article they are associated with. We divide it 75-10-15, corresponding to the splits in [28, 77, 83]. We remove two events from Politifact that are labeled both real and fake, politifact14920 and politifact14940. Work on this dataset such as [28, 77] has focused on classifying the articles, and using the tweets and related user information as supplemental information to improve performance. Because our pipeline was set up for tweet data rather than articles, we apply a simple way of classifying the articles through the tweets, by classifying the tweets and converting the predictions to an article prediction based on majority vote of the corresponding tweets' labels. Note that the works we compare with use the tweets, so this is not using an additional type of data that they have excluded. Following [28, 77], we evaluate only on articles with at least 3 corresponding tweets.

3.2.3 ***Twitter15** and **Twitter16**.*

These datasets [47, 51, 52] contain tweets related to stories from fact-checking websites and random tweets. There are 1490 and 818 tweets respectively, labeled true, false, unverified, or non-rumor. We reserve 10% for the dev set, then split the remainder 75:25, matching [30].

3.2.4 ***Twitter15 T/F** and **Twitter16 T/F**.*

These are Twitter15 and Twitter16 with the true and false examples only. This gives 742 and 412 examples respectively. We split 70-10-20, giving the same training set size as [50], which split 70-30 with no dev set.

3.2.5 ***WNUT-2020***.

This dataset was created for a shared task in WNUT-2020 on classifying tweets as "informative" or "uninformative", in providing information about "recovered, suspected, confirmed and death cases as well as location or travel history of the cases" [63]. Many of the models submitted in the competition were based on pre-trained transformer-based language models [3, 12, 33, 33, 53, 61, 61, 67, 87, 91], including the first place model which was an ensemble of CT-BERT and RoBERTa [42]. A 70-10-20 split was provided by the task organizers.

3.2.6 ***CoAID***.

The CoAID dataset [19] is a misinformation dataset related to COVID-19. This dataset is new and consists of tweets, articles, and claims. The authors provide baselines on the articles. To the best of our knowledge, as of this writing, no existing work has performed classification on the tweets.

We conducted experiments here aimed to establish a baseline for future works. However, we found that most splits result in virtually all models obtaining near-perfect (above 95) F1 score. We examined in particular the "news" tweets. Many tweets about the same news item have similar or even duplicated text, so we tried splitting similar to FakeNewsNet, first splitting the news articles and then assigning tweets to each set based on their corresponding article. We also tried subsampling a small train set to make the problem harder. This did result in one set of results with lower F1, but we found it was unstable and did not replicate consistently when we redid the splitting and training process. Thus, we do not report results on this dataset here. However, it contains a significant amount of useful data on an important problem, so We encourage future work to determine challenging splits and help others take full advantage of this resource.

3.3 Implementation Details

Each model is trained on an RTX8000 GPU using mixed precision with a batch size of 32. and learning rate of $1\mathrm{e}-5$. We do not perform any hyperparameter tuning. We train for 50 epochs on all datasets except Politifact and Gossipcop. There we train for 2 epochs, because they are much larger than the rest.

We run each model to completion exactly 5 times, and report mean and standard deviation, with two exceptions. First, datasets with a fixed split (PHEME5 Lc and WNUT-2020), which are run once. Second, a bug caused a small number of FakeNewsNet results to be lost, and due to time constraints we were unable to fully repeat the experiments. Thus we give the results of two runs each of Funnel on PolitiFact and CT-BERT on GossipCop, one of Elmo on GossipCop, and omit Funnel on Gossipcop.

Complete implementation details, hyperparameter configurations, and tweet IDs for each split of each dataset can be found on GitHub [5].

3.4 Experimental Results

3.4.1 *Performance Evaluation.*

We roughly categorize the algorithms by size. The first set are "large" models with around 400 million parameters. The second set are "medium" ones with around 100 million. The third set are "small" ones under 20 million. Order in the tables is alphabetical within category. The exact parameter counts are shown in table 1.

Results are shown in tables 2 and 3. The first row presents the algorithm that achieves, to the best of our knowledge, state-of-the-art (SOTA) performance on the given dataset and split.

We discuss these results in section 4.

3.4.2 *Potential Data Leakage.*

During our analysis, we observed a trend in the Twitter15 and Twitter16 dataset. It appears that time is a highly discriminative factor in separating the classes. This creates the possibility of data leakage by means of several

[3] https://www.politifact.com/

[4] https://www.gossipcop.com/

[5] https://github.com/ComplexData-MILA/misinfo-baselines

Table 1: Number of Parameters

CT-BERT	Funnel	RoBERTa	BERT	BERTweet	DeCLUTR	ELMo	ALBERT	BERT-Tiny
340M	468M	355M	110M	135M	125M	94M	17M	4M

Table 2: PHEME Results (Macro F1 score)

	PHEME9 T/F	PHEME5 R/NR	PHEME5 3-way	PHEME9 4-way	PHEME5 Lc	Average Rank
SOTA	82.5 [96]	87.6 [16, 92]	66.7 [16]	75.3 [97]	**51.3** [16]	5.6
CT-BERT	92.0 ± 0.9	89.0 ± 0.8	84.6 ± 1.5	79.0 ± 2.6	27.9	3.4
Funnel	86.7 ± 3.2	87.3 ± 0.6	79.4 ± 3.7	71.4 ± 3.3	28.7	6.4
RoBERTa	**93.2 ± 0.9**	**89.4 ± 0.3**	**87.7 ± 1.9**	**82.5 ± 3.3**	29.0	**2.0**
BERT	89.9 ± 1.1	87.2 ± 0.4	81.2 ± 1.4	76.8 ± 2.7	24.2	5.8
BERTweet	89.8 ± 0.6	87.3 ± 0.6	81.8 ± 0.9	76.6 ± 4.1	29.0	5.0
DeCLUTR	90.2 ± 0.8	88.3 ± 0.4	83.7 ± 2.1	77.8 ± 3.5	30.2	3.2
ELMo	81.7 ± 2.4	84.2 ± 0.8	65.8 ± 1.8	64.3 ± 4.0	30.3	9.4
ALBERT	85.3 ± 2.9	84.2 ± 2.7	71.1 ± 2.2	65.7 ± 3.1	29.4	7.2
BERT-Tiny	81.6 ± 2.0	84.7 ± 0.8	67.3 ± 2.0	61.0 ± 2.5	36.5	7.6

Table 3: Other Dataset Results (Macro F1 score)

	PolitiFact	GossipCop	Twitter15	Twitter16	Twitter15 T/F	Twitter16 T/F	WNUT-2020	Average Rank
SOTA	**92.8** [77]	85.0 [28]	**91.0** [30]	**92.4** [30]	82.5 [50]	75.9 [50]	**91.0** [43, 58]	4.4
CT-BERT	86.0 ± 3.2	90.6 ± 0.3[a]	83.5 ± 2.8	83.9 ± 0.9	93.8 ± 1.6	94.0 ± 3.5	90.6	2.8
Funnel	89.1 ± 2.5[a]	–[a]	66.9 ± 3.0	69.6 ± 2.9	83.2 ± 3.8	90.8 ± 2.2	88.5	–
RoBERTa	86.7 ± 1.2	**92.8 ± 0.5**	81.8 ± 1.5	84.8 ± 1.9	**94.4 ± 0.8**	**95.7 ± 2.8**	90.5	**2.3**
BERT	81.8 ± 3.0	89.8 ± 0.4	77.5 ± 3.3	78.2 ± 4.1	89.7 ± 1.6	91.6 ± 4.5	88.5	5.3
BERTweet	88.5 ± 1.2	92.6 ± 0.6	76.7 ± 2.9	77.7 ± 2.7	86.7 ± 1.8	92.0 ± 3.7	88.8	4.4
DeCLUTR	36.6 ± 1.4[b]	43.3 ± 0.4[b]	80.4 ± 2.6	80.5 ± 1.7	91.7 ± 1.5	94.5 ± 2.5	89.1	5.1
ELMo	83.1 ± 1.6	90.9[a]	53.7 ± 2.7	55.5 ± 4.9	74.4 ± 3.5	83.3 ± 5.0	82.4	8.0
ALBERT	80.1 ± 2.9	88.2 ± 0.9	63.4 ± 4.0	68.0 ± 3.5	83.3 ± 1.8	88.9 ± 4.3	86.8	7.7
BERT-tiny	85.3 ± 2.8	86.5 ± 0.6	54.6 ± 3.4	48.8 ± 3.9	77.8 ± 4.8	77.8 ± 4.9	79.9	8.9

[a] Due to a bug, some of the runs for this model and dataset were lost. See section 3.3 for details.
[b] Although in most cases our untuned hyperparameters work well, it appears they are not appropriate for DeCLUTR on FakeNewsNet.

potential confounding variables. To illustrate, we demonstrate that competitive performance can be achieved by using only the first 2 or 3 digits of the tweet ID. Due to the way unique IDs are generated for tweets, these digits reveal information about the time the tweet was posted.[6] We train a random forest on these digits, setting a high depth (25, such that increasing it further does not change validation performance) and otherwise using the default `scikit-learn` [65] parameters.

The class labels are balanced, so a random baseline will have 25% F1. This is roughly what one would intuitively expect from a model using only IDs as features, because the time, particularly at this approximate scale, should have little correlation if any with whether a tweet is true or false, and would not be useful on its own for classification in applied settings. Unfortunately, as shown in table 6, this is not the case. The IDs perform moderately well on Twitter15, and approach state-of-the-art on Twitter16. The non-rumor class on Twitter15 is also trivial to detect from the date.

Of course, classification using tweet IDs alone is uninteresting in isolation. However, this paints a broader picture of potential confounding variables within these datasets (and others collected in the same manner). Tweet IDs are not the only things that evolve over time; the content posted to Twitter itself changes drastically over time, as language evolves and topics enter and exit public discourse. For example, consider two particularly time-sensitive words: "Clinton" and "Trump", the surnames of the 2016 US presidential candidates. The presence of these words, like the tweet IDs, is revealing about the time at which a tweet was posted (and therefore the potential label distribution), as public discourse largely focused on their campaigns throughout 2016. In table 6, we see that the presence of those words (uncased) alone allow us to rule out "true," and in the case of Twitter15 also "false." This is obviously very unrealistic, and can cause a classifier to rely too much on particular keywords and fail to generalize to unseen data in applied settings.

After discovering this problem with Twitter15 and Twitter16, we next examined the other datasets. Results are shown in table 4. Here **Top Model** refers to the best model reported in this paper, either

[6] https://github.com/client9/snowflake2time

Table 4: Evaluating tweet ID classification (Macro F1 score)

Twitter15	False	True	Unverified	Non-rumor	Macro Avg.
SOTA [30]	92.9	90.5	85.4	95.3	91.0
2-digit RF	62.4	65.6	61.1	99.4	72.1
3-digit RF	73.0	69.5	79.7	98.2	80.1
Twitter16	False	True	Unverified	Non-rumor	Macro Avg.
SOTA [30]	91.3	94.7	89.9	93.5	92.4
2-digit RF	83.5	87.6	82.1	90.7	86.0
3-digit RF	90.7	95.3	84.4	92.9	90.8

Table 5: Label counts of tweets containing "Clinton" and "Trump"

Twitter15	Clinton	Trump	**Twitter16**	Clinton	Trump
True	0	0	True	0	0
False	0	0	False	17	18
Unverified	22	30	Unverified	17	39
Non-rumor	6	14	Non-rumor	8	6

the SOTA model (indicated by a citation) or one of our language models. **PHEME9 All** is comprised of all PHEME tweets with the full 4-way labels. We did not compare results on this split, so we do not list a top model entry in that column. **Random** refers to a stratified random classifier, i.e. outputting a class randomly with probability according to the training label distribution.

The issue is not as pronounced on these datasets as on Twitter15 and Twitter16, but is still present, especially in PolitiFact and GossipCop. CoAID and PHEME are not too bad; classifying the IDs gives better than random performance, but the performance is still bad. WNUT-2020 is split particularly well in time, as classifying the IDs is no better than random.

To examine the data from another perspective, we can look at words by label in Twitter16. In figure 1, we show an example visualization of the label "false" versus the rest. We can see that "steve" is extremely correlated with a label of "false." This is due to a false tweet about Steve Jobs being adopted that is exactly duplicated 12 times, and nearly duplicated an additional 5. In the real world, it is unlikely that "Steve" guarantees a false label, but in this data it does. We can also see that there seem to be some false tweets about mass shootings - again, unlikely to be so strongly discriminative in the real world.

4 DISCUSSION

These datasets fall on a spectrum of required generalization. On one end, there are "in-topic" datasets such as WNUT-2020 and the four PHEME splits not including PHEME5 Lc. These correspond to a real-world problem where one wants to detect misinformation or other content on a known topic. For example, a current task is detecting 5G COVID-19 conspiracy tweets [6]. On the other end, there are "cross-domain" datasets, such as PHEME5 Lc, where although the task is shared, the content of the test examples is otherwise unrelated to the train examples. A real-world application of this is detecting misinformation about a new event. In the middle, there are datasets whose examples share related domains, but have somewhat different topics, such as PolitiFact.

The language models examined here excel at "in-topic" classification, often beating state-of-the-art methods by large margins in the results here. On the other hand, they perform poorly at cross-domain tasks. For example, they produce very mediocre performance on PHEME Lc, likely due to overfitting to the topics of the training set and failing to generalize to a different topic. State-of-the-art models in this setting, such as [16], use other information beyond content to avoid this failure. Meanwhile, in between the two extreme types of datasets, we see more variable results. For example, LMs beat state-of-the-art on GossipCop but not PolitiFact.

Real-world problems can fall anywhere along the entire spectrum, and even the best results on these datasets - whether ours or others' - still fail on a significant fraction of detections. So better methods are needed on all counts. The results here suggest that understanding where one's task falls on this spectrum is very helpful. When it is further towards the "in-topic" side, recent language models may be very effective, at minimum providing strong benchmarks, and potentially a foundation on which to build more sophisticated models.

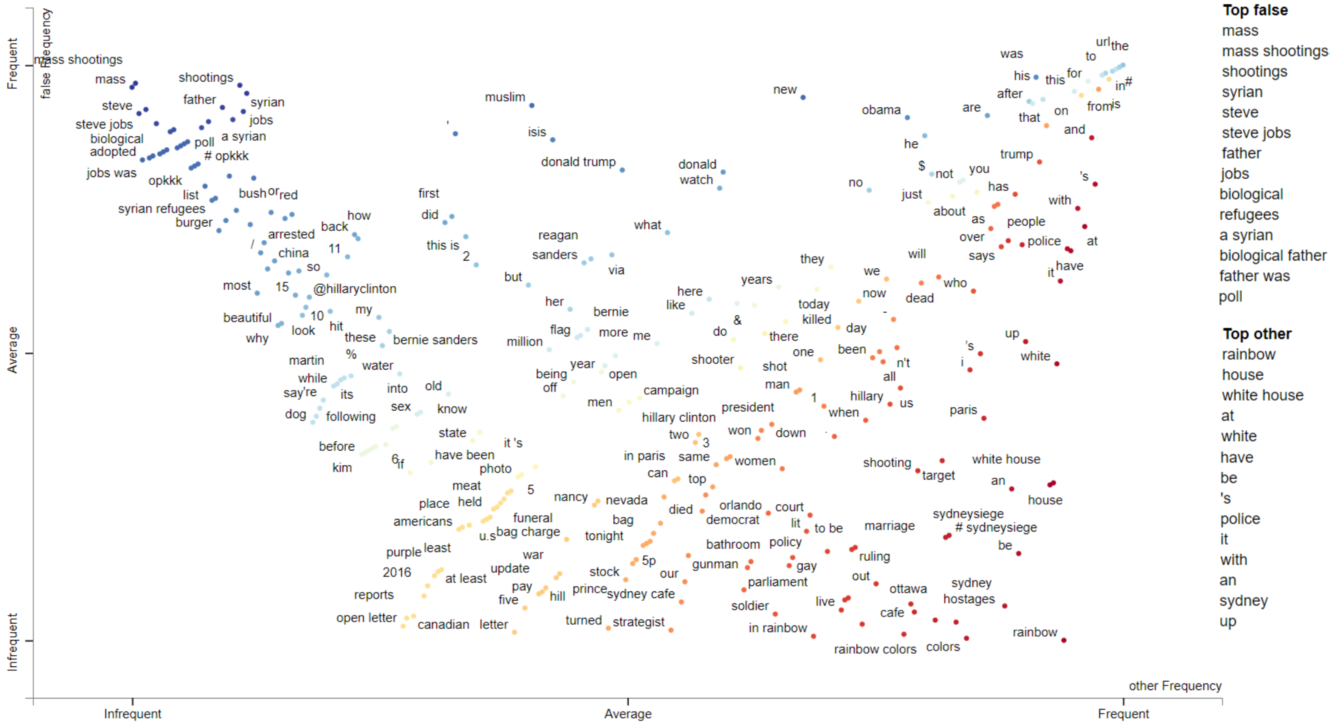
Comparing new domain-specific models with older language models may complicate interpretability of the results. The obvious, definitive solution is to always compare with the latest language models. However, due to time and computation constraints, this may not always be feasible - for example, even the largest models we report here are not the largest models available. There are three alternatives. One is to match splits and rely on other papers that run the latest LMs. This is may not be viable for new data, but we encourage work on standard datasets to provide solid benchmarks, and matching splits with those benchmarks whenever possible to facilitate comparison. As noted in the introduction, we provide IDs for our exact splits so that our results can be used in this way.[7] Another alternative is to build models which can incorporate the

[7] https://github.com/ComplexData-MILA/misinfo-baselines

Table 6: Evaluating tweet ID classification (F1 score)

	PolitiFact	GossipCop	PHEME9 All	CoAID	WNUT-2020
Top Model	92.8 [77]	91.0	-	82.2	91.6
Random	50.1	49.9	26.8	35.1	51.4
2-digit RF	77.3	65.8	25.6	49.5	34.6
3-digit RF	76.2	66.2	43.5	46.2	51.4

Figure 1: False vs. Other Classes



latest language model, and provide evidence that it improves results compared to language models alone (and does so better in some way than existing work). These may be particularly valuable, as they are to some degree future-proofed, or at least future-adaptive, since they can be updated with newer language models. Finally, researchers can explore different directions that are orthogonal to content, and give evidence that they provide information that cannot be extracted from the content by language models. There is already a great deal of research that can fall along these lines, such as [14, 32, 74]. But it is not always clear to what degree other features are orthogonal to content, and explicit analysis in future work could be useful.

Our results on classifying tweet IDs show that depending on the collection strategy, the tweet date may be surprisingly informative for the labels. This means the distribution of data may not be a good match for real-world tasks. A simple random forest on the first few digits of the tweet ID can be used to detect if this issue is present in the data. We suggest this test be applied to future datasets to ensure a reasonable level of temporal randomization, or to find and report that it is not random and thus to facilitate designing algorithms and interpreting results in light of that. Similar techniques can be applied to data from other social networks.

It is also possible to time-randomize data after the primary collection. In this domain, the fake examples are typically the ones that are hard to find, as in most contexts the majority of tweets are real [27]. We can time-randomize without retrieving new fake tweets if we can retrieve additional real tweets (and tweets from the other classes, when working with more than real/fake) from approximately the same time as the fake ones. Then we can replace the original real tweets, yielding both fake and real tweets that are hard to distinguish using time. This may be facilitated by the new Twitter Academic API,[8] which gives more historical access than the normal API. Since Twitter16 is commonly used [30, 35, 90], has severe time-determinacy, and does not have too many tweets to deal with, future work to improve this dataset in particular could

[8] https://developer.twitter.com/en/solutions/academic-research

be worthwhile, though the keyword issues with fake tweets might be hard to fix.

5 CONCLUSION

Overall, our findings here highlight the need to combine both development of better algorithms and data science. To solve critical real-world problems like misinformation, we need better understandings of how different models and datasets compare and interact. Otherwise we risk creating sophisticated models that are beaten by brute force approaches - applying the biggest LM one can run - or models that work well on standard datasets but poorly in practice, e.g. by learning to predict the date or keywords like "Steve." The SOTA models we compare with in this work have more to offer than pure performance (for example, explainability), and there are other tasks like early detection that we have not examined here. But we suggest future work can benefit from adjusting and further considering how we frame these problems. This will help not only build higher metrics but also real-world solutions.

Besides considerations for improving current lines of research, this paper suggests a two other promising directions for future work:

- Benchmarking standard misinformation datasets and publishing exact splits. We see, for example on PHEME, that there are many ways researchers have split the data, and benchmarks are lacking. This leads to difficulty for the research community in comparing results. Better benchmarks and the ability to compare exact splits would help mitigate this issue.
- New, thoroughly benchmarked and validated datasets. Although Twitter15 and Twitter16 have flaws highlighted in this paper, they also have excellent propagation information, which has likely motivated many of the approaches that leverage that to use those datasets. Improved datasets should lead to improved real-world results.

Finally, we hope to integrate these models with work on other features and data types, and produce thoroughly evaluated models that can combine the best of both recent language models and misinformation detection domain algorithms.

REFERENCES

[1] Piush Aggarwal. 2019. Classification approaches to identify informative tweets. In *Proceedings of the Student Research Workshop Associated with RANLP 2019*. 7–15.

[2] Oluwaseun Ajao, Deepayan Bhowmik, and Shahrzad Zargari. 2019. Sentiment aware fake news detection on online social networks. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2507–2511.

[3] Yandrapati Prakash Babu and R. Eswari. 2020. CIA_NITT at WNUT-2020 Task 2: Classification of COVID-19 Tweets Using Pre-trained Language Models. *ArXiv* abs/2009.05782 (2020).

[4] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473* (2014).

[5] Adrien Benamira, Benjamin Devillers, Etienne Lesot, Ayush K Ray, Manal Saadi, and Fragkiskos D Malliaros. 2019. Semi-supervised learning and graph neural networks for fake news detection. In *2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. IEEE, 568–569.

[6] MediaEval Multimedia Benchmark. 2020. FakeNews: Corona virus and 5G conspiracy. https://multimediaeval.github.io/editions/2020/tasks/fakenews/. Accessed: 2020-10-20.

[7] Alessandro Bondielli and Francesco Marcelloni. 2019. A survey on fake news and rumour detection techniques. *Information Sciences* 497 (2019), 38–55.

[8] Jeffrey Brainard. 2020. Scientists are drowning in COVID-19 papers. Can new tools keep them afloat. *Science* (2020).

[9] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165* (2020).

[10] Juan Cao, Junbo Guo, Xirong Li, Zhiwei Jin, Han Guo, and Jintao Li. 2018. Automatic Rumor Detection on Microblogs: A Survey. *CoRR* abs/1807.03505 (2018). arXiv:1807.03505 http://arxiv.org/abs/1807.03505

[11] Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. 2011. Information credibility on twitter. In *Proceedings of the 20th international conference on World wide web*. 675–684.

[12] Kumud Chauhan. 2020. NEU at WNUT-2020 Task 2: Data Augmentation To Tell BERT That Death Is Not Necessarily Informative. *ArXiv* abs/2009.08590 (2020).

[13] Sijing Chen, Jin Mao, Gang Li, Chao Ma, and Yujie Cao. 2020. Uncovering sentiment and retweet patterns of disaster-related tweets from a spatiotemporal perspective–A case study of Hurricane Harvey. *Telematics and Informatics* 47 (2020), 101326.

[14] Zhouhan Chen and Juliana Freire. 2020. Proactive Discovery of Fake News Domains from Real-Time Social Media Feeds. In *Companion of The 2020 Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, Amal El Fallah Seghrouchni, Gita Sukthankar, Tie-Yan Liu, and Maarten van Steen (Eds.). ACM / IW3C2, 584–592. https://doi.org/10.1145/3366424.3385772

[15] Mingxi Cheng, Shahin Nazarian, and Paul Bogdan. 2020. VRoC: Variational Autoencoder-aided Multi-task Rumor Classifier Based on Text. In *WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, Yennun Huang, Irwin King, Tie-Yan Liu, and Maarten van Steen (Eds.). ACM / IW3C2, 2892–2898. https://doi.org/10.1145/3366423.3380054

[16] Mingxi Cheng, Shahin Nazarian, and Paul Bogdan. 2020. VRoC: Variational Autoencoder-aided Multi-task Rumor Classifier Based on Text. In *Proceedings of The Web Conference 2020*. 2892–2898.

[17] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078* (2014).

[18] Matteo Cinelli, Walter Quattrociocchi, Alessandro Galeazzi, Carlo Michele Valensise, Emanuele Brugnoli, Ana Lucia Schmidt, Paola Zola, Fabiana Zollo, and Antonio Scala. 2020. The covid-19 social media infodemic. *arXiv preprint arXiv:2003.05004* (2020).

[19] Limeng Cui and Dongwon Lee. 2020. CoAID: COVID-19 Healthcare Misinformation Dataset. arXiv:2006.00885 [cs.SI]

[20] Limeng Cui, Haeseung Seo, Maryam Tabar, Fenglong Ma, Suhang Wang, and Dongwon Lee. 2020. DETERRENT: Knowledge Guided Graph Attention Network for Detecting Healthcare Misinformation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 492–502.

[21] Limeng Cui, Suhang Wang, and Dongwon Lee. 2019. SAME: sentiment-aware multi-modal embedding for detecting fake news. In *ASONAM '19: International Conference on Advances in Social Networks Analysis and Mining, Vancouver, British Columbia, Canada, 27-30 August, 2019*, Francesca Spezzano, Wei Chen, and Xiaokui Xiao (Eds.). ACM, 41–48. https://doi.org/10.1145/3341161.3342894

[22] Zihang Dai, Guokun Lai, Yiming Yang, and Quoc V. Le. 2020. Funnel-Transformer: Filtering out Sequential Redundancy for Efficient Language Processing. arXiv:2006.03236 [cs.LG]

[23] Ronald Denaux and Jose Manuel Gomez-Perez. 2020. Linked Credibility Reviews for Explainable Misinformation Detection. *arXiv preprint arXiv:2008.12742* (2020).

[24] 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 (2018). arXiv:1810.04805 http://arxiv.org/abs/1810.04805

[25] WA Falcon. 2019. PyTorch Lightning. *GitHub. Note: https://github.com/PyTorchLightning/pytorch-lightning* 3 (2019).

[26] John M Giorgi, Osvald Nitski, Gary D. Bader, and Bo Wang. 2020. DeCLUTR: Deep Contrastive Learning for Unsupervised Textual Representations. *ArXiv* abs/2006.03659 (2020).

[27] Nir Grinberg, Kenneth Joseph, Lisa Friedland, Briony Swire-Thompson, and David Lazer. 2019. Fake news on Twitter during the 2016 U.S. presidential election. *Science* 363, 6425 (2019), 374–378. https://doi.org/10.1126/science.aau2706 arXiv:https://science.sciencemag.org/content/363/6425/374.full.pdf

[28] Yi Han, Shanika Karunasekera, and Christopher Leckie. 2020. Graph Neural Networks with Continual Learning for Fake News Detection from Social Media. *arXiv preprint arXiv:2007.03316* (2020).

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

[30] Qi Huang, Junshuai Yu, Jia Wu, and Bin Wang. 2020. Heterogeneous Graph Attention Networks for Early Detection of Rumors on Twitter. *arXiv preprint arXiv:2006.05866* (2020).

[31] Yen-Hao Huang, Ting-Wei Liu, Ssu-Rui Lee, Fernando Henrique Calderon Alvarado, and Yi-Shin Chen. 2020. Conquering Cross-source Failure for News Credibility: Learning Generalizable Representations beyond Content Embedding. In *WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, Yennun Huang, Irwin King, Tie-Yan Liu, and Maarten van Steen (Eds.). ACM / IW3C2, 774–784. https://doi.org/10.1145/3366423.3380158

[32] Yen-Hao Huang, Ting-Wei Liu, Ssu-Rui Lee, Fernando Henrique Calderon Alvarado, and Yi-Shin Chen. 2020. Conquering Cross-source Failure for News Credibility: Learning Generalizable Representations beyond Content Embedding. In *Proceedings of The Web Conference 2020*. 774–784.

[33] Tin Van Huynh, L. Nguyen, and Son T. Luu. 2020. BANANA at WNUT-2020 Task 2: Identifying COVID-19 Information on Twitter by Combining Deep Learning and Transfer Learning Models. *ArXiv* abs/2009.02671 (2020).

[34] Muhammad Imran, Prasenjit Mitra, and Carlos Castillo. 2016. Twitter as a Lifeline: Human-annotated Twitter Corpora for NLP of Crisis-related Messages. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)* (Portoroz, Slovenia, 23-28). European Language Resources Association (ELRA), Paris, France.

[35] Mohammad Raihanul Islam, Sathappan Muthiah, and Naren Ramakrishnan. 2019. RumorSleuth: joint detection of rumor veracity and user stance. In *ASONAM '19: International Conference on Advances in Social Networks Analysis and Mining, Vancouver, British Columbia, Canada, 27-30 August, 2019*, Francesca Spezzano, Wei Chen, and Xiaokui Xiao (Eds.). ACM, 131–136. https://doi.org/10.1145/3341161.3342916

[36] Jun Ito, Jing Song, Hiroyuki Toda, Yoshimasa Koike, and Satoshi Oyama. 2015. Assessment of tweet credibility with LDA features. In *Proceedings of the 24th International Conference on World Wide Web*. 953–958.

[37] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759* (2016).

[38] Nayomi Kankanamge, Tan Yigitcanlar, Ashantha Goonetilleke, and Md Kamruzzaman. 2020. Determining disaster severity through social media analysis: Testing the methodology with South East Queensland Flood tweets. *International journal of disaster risk reduction* 42 (2020), 101360.

[39] Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882* (2014).

[40] Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).

[41] Elena Kochkina, Maria Liakata, and Arkaitz Zubiaga. 2018. All-in-one: Multi-task learning for rumour verification. *arXiv preprint arXiv:1806.03713* (2018).

[42] Priyanshu Kumar and Aadarsh Singh. 2020. NutCracker at WNUT-2020 Task 2: Robustly Identifying Informative COVID-19 Tweets using Ensembling and Adversarial Training. *arXiv preprint arXiv:2010.04335* (2020).

[43] Shamanth Kumar, Geoffrey Barbier, Mohammad Ali Abbasi, and Huan Liu. 2011. Tweettracker: An analysis tool for humanitarian and disaster relief. *ICwSM* 11 (2011), 78–82.

[44] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. *ArXiv* abs/1909.11942 (2020).

[45] Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *International conference on machine learning*. 1188–1196.

[46] L. Li, Q. Zhang, X. Wang, J. Zhang, T. Wang, T. Gao, W. Duan, K. K. Tsoi, and F. Wang. 2020. Characterizing the Propagation of Situational Information in Social Media During COVID-19 Epidemic: A Case Study on Weibo. *IEEE Transactions on Computational Social Systems* 7, 2 (2020), 556–562.

[47] Xiaomo Liu, Armineh Nourbakhsh, Quanzhi Li, Rui Fang, and Sameena Shah. 2015. Real-time rumor debunking on twitter. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*. 1867–1870.

[48] Y. Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, M. Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *ArXiv* abs/1907.11692 (2019).

[49] Ilya Loshchilov and Frank Hutter. 2017. Decoupled Weight Decay Regularization. arXiv:1711.05101 [cs.LG]

[50] Yi-Ju Lu and Cheng-Te Li. 2020. GCAN: Graph-aware co-attention networks for explainable fake news detection on social media. *arXiv preprint arXiv:2004.11648* (2020).

[51] Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, Bernard J Jansen, Kam-Fai Wong, and Meeyoung Cha. 2016. Detecting rumors from microblogs with recurrent neural networks. (2016).

[52] Jing Ma, Wei Gao, and Kam-Fai Wong. 2017. Detect rumors in microblog posts using propagation structure via kernel learning. Association for Computational Linguistics.

[53] Nickil Maveli. 2020. EdinburghNLP at WNUT-2020 Task 2: Leveraging Transformers with Generalized Augmentation for Identifying Informativeness in COVID-19 Tweets. *ArXiv* abs/2009.06375 (2020).

[54] Priyanka Meel and Dinesh Kumar Vishwakarma. 2020. Fake news, rumor, information pollution in social media and web: A contemporary survey of state-of-the-arts, challenges and opportunities. *Expert Systems with Applications* 153 (2020), 112986. https://doi.org/10.1016/j.eswa.2019.112986

[55] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[56] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.

[57] Shervin Minaee, Nal Kalchbrenner, Erik Cambria, Narjes Nikzad, Meysam Chenaghlu, and Jianfeng Gao. 2020. Deep learning based text classification: A comprehensive review. *arXiv preprint arXiv:2004.03705* (2020).

[58] Anders Giovanni Møller, Rob Van Der Goot, and Barbara Plank. 2020. NLP North at WNUT-2020 Task 2: Pre-training versus Ensembling for Detection of Informative COVID-19 English Tweets. In *Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)*. 331–336.

[59] Valeryia Mosinzova, Benjamin Fabian, Tatiana Ermakova, and Annika Baumann. 2019. Fake News, Conspiracies and Myth Debunking in Social Media-A Literature Survey Across Disciplines. *Conspiracies and Myth Debunking in Social Media-A Literature Survey Across Disciplines (February 3, 2019)* (2019).

[60] Martin Müller, Marcel Salathé, and Per E Kummervold. 2020. COVID-Twitter-BERT: A Natural Language Processing Model to Analyse COVID-19 Content on Twitter. *arXiv preprint* arXiv:2005.07503 (2020).

[61] A. Nguyen. 2020. TATL at W-NUT 2020 Task 2: A Transformer-based Baseline System for Identification of Informative COVID-19 English Tweets. *ArXiv* abs/2008.12854 (2020).

[62] Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. 2020. BERTweet: A pre-trained language model for English Tweets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*.

[63] Dat Quoc Nguyen, Thanh Vu, Afshin Rahimi, Mai Hoang Dao, Linh The Nguyen, and Long Doan. 2020. WNUT-2020 Task 2: Identification of Informative COVID-19 English Tweets. In *Proceedings of the 6th Workshop on Noisy User-generated Text*.

[64] Adam Paszke, S. Gross, Soumith Chintala, G. Chanan, E. Yang, Zachary Devito, Zeming Lin, Alban Desmaison, L. Antiga, and A. Lerer. 2017. Automatic differentiation in PyTorch.

[65] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[66] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. In *Empirical Methods in Natural Language Processing (EMNLP)*. 1532–1543. http://www.aclweb.org/anthology/D14-1162

[67] Calum Perrio and Harish Tayyar Madabushi. 2020. CXP949 at WNUT-2020 Task 2: Extracting Informative COVID-19 Tweets – RoBERTa Ensembles and The Continued Relevance of Handcrafted Features.

[68] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proc. of NAACL*.

[69] Cristina M Pulido, Beatriz Villarejo-Carballido, Gisela Redondo-Sama, and Aitor Gómez. 2020. COVID-19 infodemic: More retweets for science-based information on coronavirus than for false information. *International Sociology* (2020), 0268580920914755.

[70] Lei Qin, Qiang Sun, Yidan Wang, Ke-Fei Wu, Mingchih Chen, Ben-Chang Shia, and Szu-Yuan Wu. 2020. Prediction of Number of Cases of 2019 Novel Coronavirus (COVID-19) Using Social Media Search Index. *International Journal of Environmental Research and Public Health* 17, 7 (Mar 2020), 2365. https://doi.org/10.3390/ijerph17072365

[71] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

[72] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language Models are Unsupervised Multitask Learners. (2019).

[73] Kenneth Rapoza. 2017. Can 'fake news' impact the stock market? *by Forbes* (2017).

[74] Nir Rosenfeld, Aron Szanto, and David C. Parkes. 2020. A Kernel of Truth: Determining Rumor Veracity on Twitter by Diffusion Pattern Alone. In *WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, Yennun Huang, Irwin King, Tie-Yan Liu, and Maarten van Steen (Eds.). ACM / IW3C2, 1018–1028. https://doi.org/10.1145/3366423.3380180

[75] Victoria L Rubin, Niall Conroy, Yimin Chen, and Sarah Cornwell. 2016. Fake news or truth? using satirical cues to detect potentially misleading news. In *Proceedings of the second workshop on computational approaches to deception detection*. 7–17.

[76] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. 2008. The graph neural network model. *IEEE Transactions*

on Neural Networks 20, 1 (2008), 61–80.

[77] Kai Shu, Limeng Cui, Suhang Wang, Dongwon Lee, and Huan Liu. 2019. defend: Explainable fake news detection. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 395–405.

[78] Kai Shu, Limeng Cui, Suhang Wang, Dongwon Lee, and Huan Liu. 2019. dEFEND: Explainable Fake News Detection. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, Ankur Teredesai, Vipin Kumar, Ying Li, Rómer Rosales, Evimaria Terzi, and George Karypis (Eds.). ACM, 395–405. https://doi.org/10.1145/3292500.3330935

[79] Kai Shu, Deepak Mahudeswaran, Suhang Wang, Dongwon Lee, and Huan Liu. 2018. FakeNewsNet: A Data Repository with News Content, Social Context and Dynamic Information for Studying Fake News on Social Media. *arXiv preprint arXiv:1809.01286* (2018).

[80] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake News Detection on Social Media: A Data Mining Perspective. *SIGKDD Explorations* 19, 1 (2017), 22–36. https://doi.org/10.1145/3137597.3137600

[81] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake news detection on social media: A data mining perspective. *ACM SIGKDD Explorations Newsletter* 19, 1 (2017), 22–36.

[82] Kai Shu, Suhang Wang, and Huan Liu. 2017. Exploiting Tri-Relationship for Fake News Detection. *arXiv preprint arXiv:1712.07709* (2017).

[83] Kai Shu, Guoqing Zheng, Yichuan Li, Subhabrata Mukherjee, Ahmed Hassan Awadallah, Scott Ruston, and Huan Liu. 2020. Leveraging Multi-Source Weak Social Supervision for Early Detection of Fake News. *arXiv preprint arXiv:2004.01732* (2020).

[84] Jyoti Prakash Singh, Yogesh K Dwivedi, Nripendra P Rana, Abhinav Kumar, and Kawaljeet Kaur Kapoor. 2019. Event classification and location prediction from tweets during disasters. *Annals of Operations Research* 283, 1 (2019), 737–757.

[85] Bruno Takahashi, Edson C Tandoc Jr, and Christine Carmichael. 2015. Communicating on Twitter during a disaster: An analysis of tweets during Typhoon Haiyan in the Philippines. *Computers in human behavior* 50 (2015), 392–398.

[86] Hien To, Sumeet Agrawal, Seon Ho Kim, and Cyrus Shahabi. 2017. On identifying disaster-related tweets: Matching-based or learning-based?. In *2017 IEEE Third International Conference on Multimedia Big Data (BigMM)*. IEEE, 330–337.

[87] Khiem Vinh Tran, Hao Phu Phan, Kiet Van Nguyen, and Ngan Luu-Thuy Nguyen. 2020. UIT-HSE at WNUT-2020 Task 2: Exploiting CT-BERT for Identifying COVID-19 Information on the Twitter Social Network. *arXiv preprint arXiv:2009.02935* (2020).

[88] Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Well-Read Students Learn Better: On the Importance of Pre-training Compact Models. *arXiv preprint arXiv:1908.08962v2* (2019).

[89] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. 5998–6008.

[90] Amir Pouran Ben Veyseh, My T Thai, Thien Huu Nguyen, and Dejing Dou. 2019. Rumor detection in social networks via deep contextual modeling. In *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. 113–120.

[91] Anshul Wadhawan. 2020. Phonemer at WNUT-2020 Task 2: Sequence Classification Using COVID Twitter BERT and Bagging Ensemble Technique based on Plurality Voting. *arXiv preprint arXiv:2010.00294* (2020).

[92] Youze Wang, Shengsheng Qian, Jun Hu, Quan Fang, and Changsheng Xu. 2020. Fake News Detection via Knowledge-driven Multimodal Graph Convolutional Networks. In *Proceedings of the 2020 International Conference on Multimedia Retrieval*. 540–547.

[93] 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/1910.03771 (2019).

[94] Lianwei Wu and Yuan Rao. 2020. Adaptive Interaction Fusion Networks for Fake News Detection. *arXiv preprint arXiv:2004.10009* (2020).

[95] Lianwei Wu, Yuan Rao, Haolin Jin, Ambreen Nazir, and Ling Sun. 2019. Different absorption from the same sharing: Sifted multi-task learning for fake news detection. *arXiv preprint arXiv:1909.01720* (2019).

[96] Lianwei Wu, Yuan Rao, Yongqiang Zhao, Hao Liang, and Ambreen Nazir. 2020. DTCA: Decision tree-based co-attention networks for explainable claim verification. *arXiv preprint arXiv:2004.13455* (2020).

[97] Zhiyuan Wu, Dechang Pi, Junfu Chen, Meng Xie, and Jianjun Cao. 2020. Rumor Detection Based On Propagation Graph Neural Network With Attention Mechanism. *Expert Systems with Applications* (2020), 113595.

[98] Patrick Xia, Shijie Wu, and Benjamin Van Durme. 2020. Which* BERT? A Survey Organizing Contextualized Encoders. *arXiv preprint arXiv:2010.00854* (2020).

[99] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*. 1480–1489.

[100] Xinyi Zhou and Reza Zafarani. 2018. Fake news: A survey of research, detection methods, and opportunities. *arXiv preprint arXiv:1812.00315* (2018).

[101] Xinyi Zhou, Reza Zafarani, Kai Shu, and Huan Liu. 2019. Fake news: Fundamental theories, detection strategies and challenges. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*. 836–837.

[102] Arkaitz Zubiaga, Maria Liakata, Rob Procter, Geraldine Wong Sak Hoi, and Peter Tolmie. 2016. Analysing how people orient to and spread rumours in social media by looking at conversational threads. *PloS one* 11, 3 (2016), e0150989.