Evaluating Text Classification: A Benchmark Study

Manon Reusens[a], Alexander Stevens[a], Jonathan Tonglet[a], Johannes De Smedt[a], Wouter Verbeke[a], Seppe vanden Broucke[a,b], Bart Baesens[a,c]

[a]*Research Centre for Information Systems Engineering (LIRIS), KU Leuven, Naamsestraat 69, Leuven, 3000, Belgium*
[b]*Department of Business Informatics and Operations Management, UGent, Tweekerkenstraat 2, Gent, 9000, Belgium*
[c]*School of Management, University of Southampton, 2 University Road, Highfield, Southampton, SO17 1BJ, UK*

Email addresses: manon.reusens@kuleuven.be (Manon Reusens), alexander.stevens@kuleuven.be (Alexander Stevens), jonathan.tonglet@student.kuleuven.be (Jonathan Tonglet), johannes.desmedt@kuleuven.be (Johannes De Smedt), wouter.verbeke@kuleuven.be (Wouter Verbeke), seppe.vandenbroucke@kuleuven.be (Seppe vanden Broucke), bart.baesens@kuleuven.be (Bart Baesens)

Abstract

This paper presents an impartial and extensive benchmark for text classification involving five different text classification tasks, 20 datasets, 11 different model architectures, and 42,800 algorithm runs. The five text classification tasks are fake news classification, topic detection, emotion detection, polarity detection, and sarcasm detection. While in practice, especially in Natural Language Processing (NLP), research tends to focus on the most sophisticated models, we hypothesize that this is not always necessary. Therefore, our main objective is to investigate whether the largest state-of-the-art (SOTA) models are always preferred, or in what cases simple methods can compete with complex models, i.e. for which dataset specifications and classification tasks. We assess the performance of different methods with varying complexity, ranging from simple statistical and machine learning methods to pretrained transformers like robustly optimized BERT (Bidirectional Encoder Representations from Transformers) pretraining approach (RoBERTa). This comprehensive benchmark is lacking in existing literature, with research mainly comparing similar types of methods. Furthermore, with increasing awareness of the ecological impacts of extensive computational resource usage, this comparison is both critical and timely. We find that overall, bidirectional long short-term memory (LSTM) networks are ranked as the best-performing method albeit not statistically significantly better than logistic regression and RoBERTa. Overall, we cannot conclude that simple methods perform worse although this depends mainly on the classification task. Concretely, we find that for fake news classification and topic detection, simple techniques are the best-ranked models and consequently, it is not necessary to train complicated neural network architectures for these classification tasks. Moreover, we also find a negative correlation between F1 performance and complexity for the smallest datasets (with dataset size less than 10,000). Finally, the different models' results are analyzed in depth to explain the model decisions, which is an increasing requirement in the field of text classification.

Keywords: Benchmark, Text Classification, RoBERTa, Bidirectional LSTM, Natural Language Processing, Machine Learning

1. Introduction

Over the past several years, the field of text classification has seen significant advancements that were largely driven by deep learning techniques (Kim, 2014; Wang et al., 2023; Galke et al., 2023; Aldunate et al., 2022). Moreover, the rise of transformer-based models has drastically transformed the Natural Language Processing (NLP) landscape. Methods like Bidirectional Encoder Representations Transformers (BERT) are now widely utilized as fine-tuned models for a range of applications and languages, providing state-of-the-art results for various text classification tasks (Devlin et al., 2018; Liu et al., 2019). However, such methods require substantially more computational resources compared to *simpler*[1] methods. Additionally, with growing concerns regarding the excessive use of computational resources (Ulmer et al., 2022; Hershcovich et al., 2022; Bannour et al., 2021) and their ecological footprint, it is imperative to consider computationally efficient methods, especially when they are not necessary for the task.

[1]We define simpler methods as methods that include fewer parameters and do not require as much computation power as is necessary for deep learning models such as training BERT-based models. Moreover, these simpler models have more difficulties with understanding the text in their context.

Despite notable progress in text classification, there is a shortage of comprehensive and objective overviews that evaluate models of varying sophistication over multiple applications in text classification and offer a critical view of performance. Generally, new methods are compared to existing models (Liu et al., 2021; Agrawal et al., 2022; Kaliyar et al., 2021; Hasan et al., 2021; Jin et al., 2020; Majeed et al., 2022) or the analysis is limited to a single application in text classification (Arslan et al., 2021; Agrawal et al., 2022; Kaliyar et al., 2021; Mandal et al., 2021; Parida et al., 2021; Rahman, 2020; Zhang and Zhang, 2020). This results in non-generalizable findings across different applications and methods. Recently, some studies were published that critically reflect upon recent progress in text classification (Wahba et al., 2023; Galke et al., 2023). However, these analyses focus on applying several models without explicit consideration of different classification tasks. We aim to bridge this gap by investigating for which classification tasks, if any, simpler models suffice and when more complex methods are required. Furthermore, often studies do not provide details regarding the chosen hyperparameters, do not publish the code used, and use proprietary datasets without publishing them, hindering validation and reproducibility (Mieskes et al., 2019). Finally, the analyses of the results usually do not explore similarities in errors made by models. Additionally, they often fail to conduct (robust) statistical tests to generalize the findings.

Therefore, this paper aims to address the aforementioned gaps in the literature through a comprehensive, impartial benchmark evaluation of various methods across multiple single-label text classification tasks and datasets. This allows us to determine for which dataset specifications and classification tasks, if any, the simpler methods can compete with more complex methods. This is valuable for the field, as the need for transparency (and interpretability) of the model decisions is an increasing requirement in the field of text classification. To facilitate the latter, our code is made publicly available on GitHub[2] to support researchers in reproducing and extending our results. Concretely, our contributions are summarized as follows:

- We offer an impartial and extensive benchmark for text classification including a broad range of methods of varying complexity and a variety of datasets, spanning different applications of text classification.
- We look specifically into the dataset and task specifications and extensively analyze the results for every task separately.
- We evaluate the performance-complexity trade-off and look into the changes in performance when opting for *simpler* models.
- We evaluate the performance-variance trade-off for the different experiments conducted.
- We provide solid statistical testing to arrive at valid general conclusions. Moreover, we compare the similarities in the errors made by the different models by breaking down misclassifications for the different text classification tasks.

In this paper, we start by giving an overview of related work. Next, we discuss the methodology used including the datasets, methods, hyperparameter tuning strategy, and performance measures. Subsequently, we discuss the obtained results both in terms of performance measures and statistical tests and analyze the hyperparameters of the best models. Finally, we conclude the paper with a general conclusion and discuss directions for further research.

[2] `https://github.com/manon-reusens/text-classification-benchmark`

2. Related Work

Text classification is a fundamental task in NLP that involves assigning labels to text documents that contain written text, including one-to-many sentences and/or words, to categorize them into predefined classes (Chen et al., 2021; Otter et al., 2020; Minaee et al., 2021). In this paper, we investigate the key categories in text classification, i.e. topic classification, emotion detection, polarity detection, sarcasm detection, and fake news classification. These categories are determined based on the most important topics present in the recent literature, as described in section 3.1. Using the reference search query ('text classification' AND 'benchmark') and keywords ('text classification' OR 'Benchmarking') for all papers from 2020 until 2022 in Scopus, we summarized relevant literature that was published in qualitative journals and/ or conferences, in table 1.

To assess reproducibility, we conducted a qualitative assessment of the reproducibility of the results per paper, resulting in a reproducibility score based on three key indicators: code availability, complete dataset accessibility, and transparency in hyperparameter selection for the trained models. Papers were categorized into three groups based on their reproducibility scores: low (zero or one indicator present), medium (two indicators present), and high (all three indicators present). Additionally, the table includes a column titled 'new method(s) introduced', which indicates whether the paper solely applied previously established techniques, or whether they also introduced a new technique. For systematic literature reviews of text classification studies, we refer the reader to da Costa et al. (2023),Minaee et al. (2021),Riduan et al. (2021), and Thangaraj and Sivakami (2018).

2.1. Fake news classification

With increasing amounts of misleading and fake information, **fake news detection** has become more important over the last few years. Fake news detection is the automatic detection of false news. It can also be further restricted to the identification of intentionally falsely published news, however, in this study, we retain the broader definition. Fake news and misinformation pose significant challenges to contemporary society, with social media platforms accelerating the spreading of misinformation (Zhou and Zafarani, 2020). Consequently, it is crucial for these platforms to label fake news correctly helping the mitigation of the spread of misinformation. However, that requires the initial detection of fake news. Different methods are used for this classification problem, ranging from machine learning and statistical methods to deep learning. Regarding the former, logistic regression (LR) and support vector machines (SVMs) are often used as simple methods (Mehta et al., 2021; Capuano et al., 2023) and random forest (RF) and extreme gradient boosting (XGB) techniques are often used as ensemble techniques (Capuano et al., 2023). Furthermore, Capuano et al. (2023) find that the overall most robust results for fake news detection are given, inter alia, by extreme gradient boosting. When comparing the performance of deep learning techniques and machine learning methods, different conclusions are found across studies. Wang (2017) finds that convolutional neural networks (CNNs) perform significantly better than simple machine learning models, while Mehta et al. (2021) report similar results between both. Moreover, Sharma and Garg (2021), Wang (2017), and Mehta et al. (2021) show that (bidirectional) LSTMs perform worse than the machine learning techniques. Among deep learning techniques, pretrained transformer models perform best (Capuano et al., 2023) and in some studies, they provide the best overall performance (Mehta et al., 2021; Khan et al., 2021; Kaliyar et al., 2021). There also exists a specialized BERT for Fake news detection (Kaliyar et al., 2021) that outperforms CNNs and LSTMs, according to their study. For a full and comprehensive overview of fake news detection literature, we refer to Capuano et al. (2023). However, as shown in table 1, we find that studies focusing on

Reference	Fake News	Topic Detection	Polarity Detection	Emotion Detection	Sarcasm Detection	number of datasets	Simple Machine Learning methods	Ensemble techniques	Deep Learning	Transformers	Code available	Transparency regarding hyperparameters	New method(s) introduced	Reproducibility score
(Arslan et al., 2021)	✗	✓	✗	✗	✗	4	✗	✗	✗	✓	✗	✓	✗	low
(Mandal et al., 2021)	✗	✓	✗	✗	✗	1	✗	✗	✗	✓	✗	✓	✗	low
(Parida et al., 2021)	✗	✓	✗	✗	✗	1	✓	✗	✗	✗	✗	✓	✗	medium
(Zhang and Zhang, 2020)	✗	✓	✓	✗	✗	5	✗	✗	✓	✓	✗	✓	✓	medium
(Wang and Fan, 2020)	✗	✓	✓	✗	✗	6	✗	✗	✓	✗	✗	✓	✓	medium
(Kim et al., 2020b)	✗	✓	✓	✗	✗	7	✗	✗	✓	✗	✓	✓	✓	high
(Li et al., 2020b)	✗	✓	✓	✗	✗	4	✗	✗	✓	✗	✗	✓	✓	medium
(Rahman, 2020)	✗	✓	✗	✗	✗	1	✓	✓	✓	✗	✗	✓	✗	low
(Liu et al., 2021)	✗	✓	✓	✗	✗	3	✗	✗	✓	✓	✗ [1]	✓	✓	medium
(Majeed et al., 2022)	✗	✗	✗	✓	✗	1	✓	✓	✓	✗	✗	✓	✓	low
(Hasan et al., 2021)	✗	✗	✗	✓	✗	1	✗	✗	✓	✓	✗	✓	✓	medium
(Jin et al., 2020)	✗	✗	✗	✓	✗	1	✓	✗	✓	✗	✗ [1]	✓	✓	low
(Mohammed and Kora, 2022)	✓	✗	✓	✗	✓	6	✓	✓	✓	✗	✗	✓	✗	medium
(Yousef et al., 2020)	✗	✗	✓	✗	✗	1	✓	✗	✓	✗	✗	✗	✓	low
(Palomino and Ochoa-Luna, 2020)	✗	✗	✓	✗	✗	2	✗	✗	✓	✗	✓	✓	✓	high
(Liu et al., 2020a)	✗	✓	✓	✗	✗	4	✓	✗	✗	✗	✗	✓	✗	medium
(Yue et al., 2020)	✗	✗	✓	✗	✗	5	✗	✗	✓	✗	✗	✗ [2]	✓	low
(Kim et al., 2020a)	✗	✗	✓	✗	✗	3	✓	✓	✓	✗	✗	✗	✗	low
(Lê et al., 2020)	✗	✗	✓	✗	✗	2	✗	✗	✗	✓	✗	✓	✓	medium
(Sutoyo et al., 2022)	✗	✗	✓	✗	✗	1	✓	✓	✗	✗	✗	✗	✗	low [3]
(Qureshi et al., 2022)	✗	✗	✓	✗	✗	1	✓	✗	✓	✗	✗	✗	✗	low
(He et al., 2020)	✗	✗	✗	✗	✓	4	✗	✗	✓	✗	✓	✓	✓	medium
(Choudhary et al., 2021)	✓	✗	✗	✗	✗	4	✗	✗	✓	✗	✗	✓	✓	medium
(Jindal et al., 2020)	✓	✗	✗	✗	✗	2	✗	✗	✓	✓	✗	✓	✓	medium
(Mehta et al., 2021)	✓	✗	✗	✗	✗	2	✗	✗	✗	✓	✗	✓	✓	medium
(Sharma and Garg, 2021)	✓	✗	✗	✗	✗	3	✓	✓	✓	✗	✗	✓	✗	medium
(Wang, 2017)	✓	✗	✗	✗	✗	1	✓	✗	✓	✗	✗	✓	✗	medium
(Kaliyar et al., 2021)	✓	✗	✗	✗	✗	1	✗	✗	✓	✓	✗	✓	✓	medium
(Worsham and Kalita, 2018)	✗	✓	✗	✗	✗	1	✓	✓	✓	✗	✓	✓	✗	high
(Escalante et al., 2016)	✗	✓	✗	✗	✗	4	✓	✗	✗	✗	✗	✗	✗	low
(Kim, 2014)	✗	✗	✓	✗	✗	7	✗	✗	✓	✗	✗	✓	✗	medium
(Pang et al., 2002)	✗	✗	✓	✗	✗	1	✓	✗	✗	✗	✗	✗	✗	low
(Kang et al., 2018)	✗	✗	✓	✗	✗	4	✓	✗	✓	✗	✗	✗	✓	low
(Liu et al., 2019)	✗	✗	✓	✗	✗	9	✗	✗	✗	✓	✓	✓	✓	high
(Wang et al., 2023)	✗	✓	✓	✗	✗	5	✗	✗	✓	✓	✓	✓	✓	high
(Sun et al., 2023)	✗	✓	✓	✗	✗	5	✗	✗	✓	✓	✓	✓	✓	high
(Wahba et al., 2023)	✗	✓	✗	✗	✗	3	✓	✗	✗	✓	✗	✗	✗	low
(Galke et al., 2023)	✗	✓	✓	✗	✗	5 [4]	✓	✗	✓	✓	✓	✓	✗	high
Our study	✓	✓	✓	✓	✓	20	✓	✓	✓	✓	✓	✓	✗	high

[1] Github is offline.
[2] Not that clear about for example what dropout values were tuned.
[3] Link to dataset not available anymore.
[4] Also 7 multilabel datasets were used.

Table 1: Relevant literature on text classification.

this text classification category, tend to only focus on this category without including others. Moreover, limited analysis across different models of different complexity is given. As shown in the table, code is often not available in these papers, resulting in a medium reproducibility score.

2.2. Topic classification

Topic classification is defined as automatically categorizing a text document into a predefined topic. We include classifying both texts from news articles and other texts into topics. This classification task helps people locate content that aligns with their interests more easily. For instance, in news categorization, the automatic detection of topics of news articles enables people with a preference in sports articles to find the articles of their preference more easily (Minaee et al., 2021). Machine learning methods that are often used for this classification task are LR, SVMs, RFs, XGB, and naive Bayes (NB) (Rahman, 2020; Parida et al., 2021; Worsham and Kalita, 2018). However, NB only performs at par or slightly worse compared to other machine learning models such as SVMs and LR across different applications in text classification including topic detection (Rahman, 2020). This method, however, is shown to perform well in early text classification (Escalante et al., 2016), that is when the category of a text document has to be known when only partial information about the text is provided. Moreover, Worsham and Kalita (2018) conclude that boosting mechanisms outperformed deep learning models such as CNNs and LSTMs. In addition to these two models, also bidirectional LSTMs and pretrained language models such as BERT, robustly optimized BERT pretraining approach (RoBERTa), and XLNet are used for this classification task (Minaee et al., 2021; Kim et al., 2020b; Li et al., 2020b; Wang and Fan, 2020; Worsham and Kalita, 2018). In several studies, it is shown that RoBERTa outperforms other large pretrained language models such as BERT and XLNet (Arslan et al., 2021; Mandal et al., 2021) and Sun et al. (2023) also include generative models for text classification in their analysis. Wahba et al. (2023) argue that for certain text classification tasks, linear models can provide similar results to the large, expensive transformer models with the additional benefit of being cheaper, comparable, interpretable, and reproducible. This is an interesting finding, however as the study was too limited (only three datasets and one text classification category are included), this finding should be tested on more datasets to come to generalizable conclusions over different text categories, a gap which we address in this work. The limited combination of different text classification categories is also shown in table 1. If different categories are combined, this is mostly limited to the combination of topic detection and polarity detection. Moreover also regarding method comparisons, studies often do not compare methods of many complexities. Galke et al. (2023) has compared several, however, they only focus on text classification and polarity detection and provide a limited number of datasets, which makes generalizing the findings harder.

2.3. Sentiment analysis

Sentiment analysis is an application of text classification where people's sentiment is automatically detected from written text. This research area comprises multiple applications such as emotion, polarity, and sarcasm detection. While these tasks are often grouped under sentiment analysis, we consider them as separate classification tasks due to their distinct and intrinsic characteristics. **Emotion detection** involves identifying emotion from written text, making it a multi-class task. Automatic emotion detection from text is beneficial for several areas including business, psychology, and human-robot interactions. Social media platforms provide the perfect medium for people to express their emotions in written text. Analyzing such data can result in psychological and business insights. Additionally, this analysis can

contribute to improving interactions between humans and robots, making them more empathetic and effective (Alswaidan and Menai, 2020). On the other hand, **polarity detection**, closely related to emotion detection, is a binary classification task where text is categorized in a positive or negative sentiment. Similar to the emotion detection task, also for polarity detection, businesses can gain valuable insights by automatically analyzing customer satisfaction and brand perception (Soleymani et al., 2017). Additionally, this analysis can also be used to obtain official statistics regarding the overall sentiment of a population (Boom and Reusens, 2023). Finally, **sarcasm detection** aims to identify sarcastic documents from non-sarcastic ones. This task holds significant value in sentiment analysis, as sarcastic texts have a negative underlying sentiment behind an ostensibly positive facade (Joshi et al., 2017). For businesses analyzing their product reviews, this distinction between sarcastic and non-sarcastic is crucial in preventing the misinterpretation of feedback from customers.

For these classification tasks, many simple machine learning methods such as SVMs, NB, ensemble methods such as RF and extreme gradient boosting are used for sentiment analysis (Sutoyo et al., 2022; Pang et al., 2002), similar to the methods used for fake news classification and topic classification. While hidden Markov models were historically also used for sentiment analysis (Kang et al., 2018), they are less prevalent in recent systematic literature reviews such as (Riduan et al., 2021; Thangaraj and Sivakami, 2018). Similarly to topic classification, NB shows similar or slightly worse performance than other machine learning models such as SVMs and logistic regression across different applications in sentiment analysis such as polarity prediction (Qureshi et al., 2022; Sutoyo et al., 2022; Mohammed and Kora, 2022), and emotion detection (Jin et al., 2020). Moreover, regarding deep learning architectures, Mohammed and Kora (2022); Jin et al. (2020) show that a bidirectional LSTM outperforms a CNN. Note that in some cases, these models are surpassed by machine learning models (Qureshi et al., 2022). In several studies, it was shown that RoBERTa often outperforms other large pretrained language models such as BERT and XLNet (Liu et al., 2019). Also for these categories, studies use limited amounts of datasets, which makes the generalization of results difficult. Furthermore, studies also often focus only on one or at most two text classification categories at the same time. Recently, we see that the reproducibility of the different studies has improved, however, still, not all studies are publishing their code and/or datasets and being transparent about the hyperparameters.

To summarize, most studies do not include different classification tasks and methods of varying complexity. Furthermore, they entail only a limited amount of datasets and provide limited reproducibility by not making their datasets and/or code publicly available and often not being transparent about the chosen hyperparameters. We address all these gaps in this work.

2.4. Other classification tasks

In our study, we concentrated on the most popular categories prevalent in the literature, outlined in Table 2. However, it is noteworthy to mention that other text classification tasks have also gained importance because of the rise of internet platforms. One example is *hate speech detection*. Given the rise of hate speech, correlated with the growth of these platforms, where people can express their opinions freely and sometimes also anonymously, the automatic detection of such text is another important research area. While our study does not elaborate further on this task, we refer to (Fortuna and Nunes, 2018) for a comprehensive overview of this interesting research field.

In addition, including multiple modalities is an important direction to further improve the detection of misinformation in the field of fake news detection (Comito et al., 2023). Moreover, also in sentiment analysis tasks, including facial and vocal displays, can provide deeper insights

into an individual's sentiment(Soleymani et al., 2017). Similarly, hate speech detection can benefit from using multimodal approaches, considering that often hate speech goes beyond textual content, also including visual elements, e.g. memes, and videos (Chhabra and Vishwakarma, 2023).

3. Methodology

3.1. Data

We gathered the most prominent topics in text classification for our benchmark study using the search query ('text classification' AND 'benchmark') and keywords ('text classification' OR Benchmarking') for all papers between 2020 and 2022 in Scopus. From these articles, we selected the ones that fit within the scope of our paper and were published in reputed journals or conferences. Next, we indicated the five most occurring categories. A list of the different topics that were identified together, including references, is shown in Table 2. Note that we discarded the papers where multilabel classification was considered and that we thus solely focus on single-label predictions.

Text classification task	**References**
NLU	(Naseem et al., 2022)
Topic classification	(Arslan et al., 2021; Mandal et al., 2021; Parida et al., 2021; Zhang and Zhang, 2020; Wang and Fan, 2020; Kim et al., 2020b; Li et al., 2020b; Rahman, 2020; Liu et al., 2021, 2020a)
Emotion detection	(Majeed et al., 2022; Hasan et al., 2021; Jin et al., 2020)
Polarity detection	(Mohammed and Kora, 2022; Wang and Fan, 2020; Yousef et al., 2020; Palomino and Ochoa-Luna, 2020; Liu et al., 2020a; Kim et al., 2020b; Yue et al., 2020; Kim et al., 2020a; Lê et al., 2020; Li et al., 2020b; Liu et al., 2021; Sutoyo et al., 2022; Qureshi et al., 2022)
Sarcasm detection	(Mohammed and Kora, 2022; He et al., 2020)
Fake news classification	(Choudhary et al., 2021; Mohammed and Kora, 2022; Jindal et al., 2020)
Spam detection	(Liu et al., 2020b)
Adverse drug reaction	(Yousef et al., 2020; Li et al., 2020c)
Subjectivity detection	(Yue et al., 2020)
Implied pornography	(He et al., 2020)

Table 2: Overview categories of text classification in literature.

For each of these classification tasks, we conducted an empirical analysis to find the four English datasets that are most often downloaded and cited in the existing literature. Deriving conclusions from only one dataset cannot be generalized, whereas, for two datasets, conclusions cannot be made from contradictory results. Furthermore, as Table 1 shows an average of 3.2 datasets per study, we opted to include four datasets per category in our study. Moreover, we focus on English datasets, as for different languages different models might be preferential. Table 3 offers an overview of the selected datasets per classification task. These datasets are also often occurring in the studies listed in Table 2. In Appendix 6.1, a description for each of the different datasets is included.

For the Gossipcop dataset of the FakeNewsNet Repository, we found many missing values. Therefore, we did our analysis solely using the titles which are present in the dataset. For the CoAID dataset, we decided to solely focus on news articles, and therefore, discarded the

Classification Task		Dataset	Classes	Size dataset	Source
	Fake News Classification	FakeNewsNet Repository: Gossipcop	2	22,140	(Shu et al., 2018)
		CoAID	2	2,162	(Cui and Lee, 2020)
		LIAR	6	12,836	(Wang, 2017)
		McIntire	2	4,594	McIntire [1]
	Topic Classification	20News	20	18,846	20News [2]
		AGNews	4	127,600	(Zhang et al., 2015)
		Web of Science Dataset (WOS)	7	11,967	(Kowsari et al., 2017)
		BBC	5	2,225	(Greene and Cunningham, 2006)
Sentiment Analysis	Emotion Detection	TweetEval Emotion Detection	4	5,052	(Barbieri et al., 2020)
		CARER Emotion	6	20,000	(Saravia et al., 2018)
		DailyDialog Act Corpus- Silicone	7	102,979	(Chapuis et al., 2020)
		MELD	7	13,708	(Poria et al., 2018)
	Polarity Detection	IMDb	2	50,000	(Maas et al., 2011)
		The Stanford Sentiment Treebank (SST2)	2	68,221	(Socher et al., 2013)
		Movie Review	2	10,662	(Pang and Lee, 2005)
		Customer Reviews (CR)	2	3,770	(Ding et al., 2008)
	Sarcasm Detection	iSarcasm - English	2	4,868	(Abu Farha et al., 2022)
		SemEval task 3	2	4,601	(Armendariz et al., 2020)
		Sarcasm News Headlines (SNH)	2	55,328	(Misra and Arora, 2019; Misra and Grover, 2021)
		Sarcasm v2: General (GENsarc)	2	6,520	(Oraby et al., 2016)

[1] `https://github.com/GeorgeMcIntire/fake_real_news_dataset`
[2] `http://qwone.com/~jason/20Newsgroups/`

Table 3: Overview of dataset specifications.

claims data since we opt for a consistent approach including only news articles. For the sarcasm detection datasets, we aim for a consistent experimental setup. Therefore, we only considered the binary classification task of the SemEval dataset and left the quaternary classification task out of the analysis.

3.2. Preprocessing

Given the lack of a widely accepted standard for text preprocessing, we adopt the techniques employed by Kratzwald et al. (2018): tokenization, removal of Unicode, punctuation, repeated letters, digits, and stop word removal. Additionally, for datasets containing tweets, we also removed emojis and URLs. Finally, we applied lemmatization – instead of stemming in (Kratzwald et al., 2018). Both preprocessing methods reduce the words to a root word. Lemmatization, however, also makes sure that the resulting word is an existing word (Reusens et al., 2022). Therefore, to preserve existing words when utilizing pretrained embeddings, we applied lemmatization as was done in (Alaparthi and Mishra, 2021).

Because of the ability of deep learning methods to handle raw text as input (Kraus et al., 2020; LeCun et al., 2015), we refrain from applying any preprocessing for these approaches. While we acknowledge that different preprocessing can influence the results obtained with different models, we conjecture that applying the same preprocessing for deep learning and machine learning methods may hinder the full potential of the more advanced deep learning methods. On the contrary, this would most likely be to their disadvantage. To ensure a fair comparison, we deliberately chose different preprocessing for the deep learning approaches to fully leverage their potential.

3.3. Vector Representations

The vector representations of the input text for the traditional machine learning models are gathered using TF-IDF and FastText.

TF-IDF stands for Term Frequency Inverse Document Frequency. This counting method is more sophisticated than Bag of Words (BoW). Unlike BoW, it does not only take into account the different term frequencies, but also the inverse document frequencies. We implemented this technique using (Pedregosa et al., 2011).

FastText (FT) offers pretrained word embeddings and is introduced by Mikolov et al. (2018). Pretrained word embeddings are vector representations of words that also include syntactic and semantic word relationships (Mikolov et al., 2013b). This method can handle unseen and rare words, contrary to other pretrained word embeddings such as word2Vec (Mikolov et al., 2013b,a) and GloVe (Pennington et al., 2014).

3.4. Methods

The methods considered range from the more simple machine learning methods to the current state-of-the-art. In several studies, different machine learning methods show similar performance (Qureshi et al., 2022; Sutoyo et al., 2022; Mohammed and Kora, 2022), hence we decided to leave out NB and a simple decision tree. Moreover, we leave stacked methods as a topic to explore in future research.

Logistic regression (LR) is a binary classification algorithm that estimates the relationship between the dependent and one or more independent variables with the use of a maximum likelihood function. The method predicts the log-odds of an instance belonging to a class and consequently converts this into a probability via the logistic function. In the multi-class scenario, the one versus rest-scheme is used, implemented by (Pedregosa et al., 2011).

Support vector machine (SVM) is a machine learning algorithm used for classification and regression. It estimates a hyperplane in the feature space using a large margin idea to separate the data into different classes. In the multi-class case, these models are trained using the one versus rest-scheme. We implemented the SVM using (Pedregosa et al., 2011).

Random forest (RF) is an ensemble technique of decision trees and is often used for both classification and regression. The method averages the predicted outcomes of the individual decision trees, each generated from a bootstrapped training dataset. The implementation was done using (Pedregosa et al., 2011)

Extreme gradient boosting (XGB) is a machine learning algorithm used for classification and regression. It is an ensemble learning method that combines multiple decision trees sequentially by which each newly added classifier corrects the mistakes of previously trained classifiers. We implemented the algorithm using (Chen and Guestrin, 2016).

Bidirectional long short-term memory network (BiLSTM) is a type of Recurrent Neural Network (RNN) that is designed to avoid long-term dependency problems with BackPropagation Through Time (BPTT) to overcome the vanishing or exploding gradient problem (Hochreiter and Schmidhuber, 1997). It contains two separate LSTM layers, where one LSTM layer reads the text in the forward direction and the other in the backward direction. This allows the network to gather dependencies in the data in both directions. This method was implemented using (Abadi et al., 2015). Moreover, we used FastText for the embedding layer of the model, as shown by Kim (2014) to improve the performance.

Convolutional neural network (CNN) is a deep feed-forward artificial neural network consisting of a series of convolutional and pooling layers followed by fully connected layers and a softmax output. The convolutional layer learns a number of filters or kernels by preserving spacial topology. The following pooling layer reduces the spatial size of the data and helps to control overfitting. This architecture is often used for image analysis, audio recognition, and natural language processing. The implementation of the CNN using (Abadi et al., 2015) is based on the implementation by Kim (2014) and includes an embedding layer based on pretrained FastText embeddings, as shown by Kim (2014) to improve the performance.

Robustly optimized BERT pretraining approach (RoBERTa) is a state-of-the-art transformer-based language model introduced by Liu et al. (2019) and is an improved version of the Bidirectional Encoder Representations from Transformers (BERT) model (Devlin et al., 2018). In literature, RoBERTa often outperforms other pretrained language models (Arslan et al., 2021; Liu et al., 2019; Mandal et al., 2021). RoBERTa is pretrained on a large corpus of text using dynamic masking to learn contextual relationships between words in sentences. The model can be finetuned for several different tasks such as text classification. We finetuned RoBERTa using (Wolf et al., 2020).

3.5. Hyperparameter Tuning

This section offers an overview of the hyperparameter tuning strategy. We start by explaining the implemented search strategy followed by the hyperparameters that are tuned. We used Weights and Biases (Biewald, 2020) to retain an organized overview of the different experiments, including hyperparameter tuning.

3.5.1. Search Strategy

In the literature, various search strategies for hyperparameter tuning have been explored. Bergstra and Bengio (2012) highlight the advantages of random search over a full grid search, while Shahriari et al. (2015) note that random search allows for the exploitation of the full dimensionality without knowing which dimensions are most important. Alternatively, Bayesian search is also often used (Snoek et al., 2012). This search strategy chooses new hyperparameter settings for the next iteration based on the performance of the previous hyperparameter setting and shows great potential as it optimizes the hyperparameter setting efficiently (Snoek et al., 2012). Although random search can effectively exploit the full dimensionality of a hyperparameter search, it faces challenges in handling high dimensionality (Shahriari et al., 2015). To overcome this limitation, we propose the following strategy for hyperparameter tuning.

We follow a coarse-finetuning strategy similar to Li et al. (2021); Huang et al. (2021) combined with Bayesian hyperparameter tuning for the neural networks. We start with common hyperparameter settings. Around these settings, we generate an interval and search for the best hyperparameter setting per dataset and per classifier. Through this strategy, we can reduce the dimensionality. Furthermore, to lower the probability of encountering local maxima, we run each Bayesian search for 10 different random seeds. As the machine learning methods suffer less from the high dimensionality problem, we also applied Bayesian hyperparameter tuning for these methods. Additionally, we combine Bayesian search with hyperband for the neural networks, as it is shown by Falkner et al. (2017) to perform best. This bandit-based strategy evaluates whether training should be stopped at predefined iteration counts or brackets. Around five brackets should be selected per run, with a minimum of three brackets. Therefore, we set the minimum number of iterations at one (Li et al., 2017). The hyperparameter tuning

of RoBERTa was done using the suggested hyperparameter settings of the original paper, combined with Bayesian search. For each random seed, we ran i iterations, with i the minimum of the number of combinations of the different hyperparameter settings and 30 (Yogatama et al., 2015).

3.5.2. Hyperparameters

The hyperparameter settings per model are shown in Table 4. As previously explained, the hyperparameters for the neural networks were chosen based on a coarse-finetuning approach. Therefore, in Table 4 we refer to Appendix 6.2 for an overview of commonly used hyperparameter settings. Note that the hyperparameter C for LR and SVM stands for the L2 regularization parameter, which is inversely proportional to the regularization strength. Moreover, for bidirectional LSTMs and convolutional neural networks, exponential decay was set to the learning rate with patience 20 until 0.00000001 (Sachan et al., 2019).

Method	**Hyperparameter**	**Value**	**Reference**
LR	Class weights	balanced	(Ghosh, 2022)
	C	{0.001, 0.01, 0.1, 1, 10, 100, 1000}	(Galli et al., 2022)
SVM	Class weights	balanced	(Moreo et al., 2021)
	Kernel	linear	(Moreo et al., 2021)
	C	{0.001, 0.01, 0.1, 1, 10, 100, 1000}	(Moreo et al., 2021)
RF	Number of Estimators	[1,200]	(Wu et al., 2019b)
	Maximum of Features	[1, 20]	(Wu et al., 2019b)
XGB	Learning Rate	{0.0001, 0.001, 0.01, 0.1}	(Lai et al., 2019)
	Maximum Depth	[3,7]	(Lai et al., 2019)
	Gamma	[1,10]	(Lai et al., 2019)
	Number of Estimators	{10, 100, 1000, 10000}	(Lai et al., 2019)
	Colsample by Tree	[0.1, 1]	(Lai et al., 2019)
BiLSTM	Optimizer	Adam	Appendix 6.2
	Learning Rate	{0.0001, 0.001, 0.01}	Appendix 6.2
	Hidden Layer Size	{128, 256, 512 }	Appendix 6.2
	Batch Size	{64, 128, 256, 512, 1024, 2048}	Appendix 6.2
	Epochs	100	Appendix 6.2
	Hidden Layers	1	Appendix 6.2
	Drop-out	0.5	Appendix 6.2
CNN	Optimizer	Adam	Appendix 6.2
	Learning Rate	{0.0001, 0.001, 0.01}	Appendix 6.2
	Filters	{128, 256, 512 }	Appendix 6.2
	Batch Size	{64, 128, 256, 512, 1024, 2048}	Appendix 6.2
	Epochs	100	Appendix 6.2
	Drop-out	0.5	Appendix 6.2
RoBERTa	Batch size	{16, 32}	(Liu et al., 2019)
	Learning Rate	{0.00001, 0.00002, 0.00003}	(Liu et al., 2019)
	Epochs	10	(Liu et al., 2019)

Table 4: Hyperparameter table.

3.6. Evaluation Methods

3.6.1. Performance Metrics

To ensure that the obtained results are robust, reliable and generalizable, we only report results that are averaged over ten different experimental runs and initialized with a different

seed. This additional measure on top of the extensive hyperparameter search and optimization is necessary to eliminate random effects. To evaluate the performance of the different techniques, we calculate the accuracy, macro F1-measure, macro precision, and macro recall for all the best-performing models per sweep. For the binary classification tasks, we also calculate the area under the ROC-curve (AUC) score and the area under the precision-recall curve (AUCPR) score. We then average the performance metrics over the ten runs per classifier and dataset to obtain one average per method-dataset combination. For all of these experiments, we also provide the standard deviation over the different random seeds.

3.6.2. Statistical testing

To account for sampling noise in comparing the performance of the various methods across the experiments and to arrive at statistically valid conclusions, we rigorously apply statistical tests using Autorank (Herbold, 2020). First, we perform pairwise tests across the different classifiers using the Wilcoxon signed-rank test with both a 90% and 95% confidence level (Chen et al., 2021; Pattanayak et al., 2021). This statistical test is non-parametric and therefore a valid alternative to the paired t-test, as it makes fewer assumptions and is stronger than the sign test (Demšar, 2006).

To mitigate the risk of incorrectly rejecting null hypotheses during multiple pair-wise tests, we adopt a statistical testing procedure as proposed by Demšar (2006) to compare multiple classifiers. In the initial step, we employ the Friedman test, the non-parametric equivalence of the ANOVA test, to check whether there is a statistically significant difference across the classifiers. If a significant difference is detected, we proceed with the post-hoc Nemenyi test to evaluate the significance of the difference between the best-performing method and the other methods (Nemenyi, 1963; Pattanayak et al., 2021). This two-step approach ensures a robust and reliable assessment of the performance of the classifiers.

4. Results

In this section, we first give an overview of the results of the different methods per dataset. Next, we continue with the statistical tests. This subsection is threefold: we perform statistical tests over all datasets and per performance measure, statistical tests per text classification application, and statistical tests on the results classified in dataset specifications. Finally, we analyze the hyperparameters of the best models.

4.1. Experiments

Table 5 displays the average performance and standard deviation in terms of the F1-measure for all datasets and methods calculated over the ten different random seed initializations. The other performance metrics are shown in Appendix 6.3. Overall, we see a wide variation in the difficulty of the different predictive tasks, with the results on the LIAR, Silicone, and MELD datasets being the worst. Furthermore, the best-performing model differs across the different datasets. However, some patterns can be detected. In the next paragraphs, we discuss the performance of the different datasets per classification task. In addition, when comparing our results to existing leaderboards for similar datasets, we find similar results.

First, TF-IDF seems to work well for fake news classification, although the results reveal variability in the best-performing method per dataset. Notably, LR TF-IDF scores best on recall for two out of four best datasets, and RF TF-IDF scores best for the AUC and AUCPR scores for two out of four datasets. For the Gossipcop dataset, LR TF-IDF performs best, however, many non-significant differences with other methods are observed with a confidence

level of 95%. Concerning CoAID, SVM TF-IDF performs significantly better than all other methods, except for LR TF-IDF, RF FT, and RF TF-IDF with a confidence level of 95%. Regarding LIAR, no significant differences across the performance of the different models are found, except for LR TF-IDF with a confidence level of 90%. Finally, the best-performing method for the McIntire dataset is RoBERTa, significantly outperforming all other models except RF TF-IDF with a confidence of 95%.

Similarly to fake news classification, we find different best-performing methods per dataset for topic detection, except for the datasets WOS and BBC where RoBERTa is the best-performing method. LR TF-IDF and BiLSTM are the best-performing methods for 20News and AGNews, respectively. With a confidence level of 90%, we conclude that these best-performing methods significantly outperform all other models trained on the same dataset.

For emotion detection, RoBERTa performs well on the TweetEval and CARER datasets, significantly outperforming all other methods with a confidence level of 90% except RF TF-IDF on the CARER dataset. Moreover, RoBERTa is, also on the MELD dataset, the best-performing method, significantly outperforming all other methods except XGB FT and XGB TF-IDF. For silicone, however, the biggest dataset in the emotion detection category, RoBERTa performs worst out of all methods in terms of F1-measure, precision, and recall, despite obtaining a decent accuracy score. Upon closer inspection, it is found that all 10 trained RoBERTa models predict the same label for over 90% of the cases. As the dataset is imbalanced, this results in a good accuracy score but bad other performance metrics. Moreover, RF TF-IDF significantly outperforms RoBERTa and XGB FT with a confidence level of 90%.

For polarity detection, RoBERTa outperforms all other methods for three out of four datasets in terms of all performance metrics except for AUC and AUCPR where the BiLSTM outperforms RoBERTa. Moreover, the null hypothesis that the performance of BiLSTM and RoBERTa are similar could not be rejected at a 95% confidence level for these three polarity detection tasks. Additionally, the null hypothesis could also not be rejected for CNN, XGB TF-IDF, SVM TF-IDF, and LR TF-IDF on the IMDb dataset, while all other methods perform significantly worse than RoBERTa at a 95% confidence level. For the SST2 and Movie Review datasets, all methods except BiLSTM show significantly lower performance than RoBERTa, with a confidence level of 95%. For the CR dataset, however, RoBERTa offers bad results in terms of all performance metrics except accuracy, because of the dataset imbalance.

		Fake News Classification				Topic Modeling			
model	**vec_rep**	**Gossipcop**	**CoAID**	**LIAR**	**McIntire**	**20News**	**AGNews**	**WOS**	**BBC**
LR	**FT**	0.71 ± 0.01	0.90 ± 0.02	***0.23 ± 0.01***	0.82 ± 0.01	*0.62 ± 0.00*	*0.89 ± 0.00*	0.84 ± 0.00	*0.97 ± 0.00*
	TF-IDF	**0.77 ± 0.01**	***0.95 ± 0.02***	*0.22 ± 0.00*	0.83 ± 0.01	**0.66 ± 0.00**	*0.91 ± 0.00*	*0.89 ± 0.00*	*0.97 ± 0.01*
SVM	**FT**	0.72 ± 0.01	0.90 ± 0.03	**0.23 ± 0.01**	0.82 ± 0.01	*0.60 ± 0.01*	*0.89 ± 0.00*	*0.84 ± 0.03*	*0.96 ± 0.01*
	TF-IDF	0.76 ± 0.01	**0.95 ± 0.01**	***0.22 ± 0.02***	0.82 ± 0.02	*0.64 ± 0.01*	*0.91 ± 0.01*	*0.89 ± 0.01*	*0.97 ± 0.00*
RF	**FT**	0.65 ± 0.01	***0.94 ± 0.01***	***0.20 ± 0.01***	0.80 ± 0.01	*0.54 ± 0.00*	*0.88 ± 0.00*	0.79 ± 0.01	*0.94 ± 0.01*
	TF-IDF	***0.74 ± 0.01***	***0.94 ± 0.01***	***0.22 ± 0.01***	*0.83 ± 0.01*	*0.61 ± 0.00*	*0.91 ± 0.01*	*0.82 ± 0.01*	*0.93 ± 0.01*
XGB	**FT**	***0.73 ± 0.01***	0.95 ± 0.01	***0.22 ± 0.01***	0.81 ± 0.01	*0.57 ± 0.03*	*0.87 ± 0.04*	*0.81 ± 0.04*	*0.94 ± 0.00*
	TF-IDF	***0.74 ± 0.01***	0.89 ± 0.02	***0.22 ± 0.01***	0.80 ± 0.01	*0.59 ± 0.01*	*0.87 ± 0.10*	*0.91 ± 0.02*	*0.94 ± 0.01*
BiLSTM		***0.76 ± 0.01***	0.91 ± 0.01	***0.23 ± 0.01***	0.82 ± 0.01	*0.61 ± 0.01*	**0.93 ± 0.00**	*0.91 ± 0.01*	*0.94 ± 0.01*
CNN		***0.75 ± 0.01***	0.90 ± 0.01	***0.20 ± 0.01***	0.81 ± 0.01	*0.55 ± 0.01*	*0.91 ± 0.00*	*0.88 ± 0.01*	*0.94 ± 0.01*
RoBERTa		***0.76 ± 0.01***	0.94 ± 0.00	***0.14 ± 0.01***	**0.90 ± 0.01**	*0.59 ± 0.01*	*0.89 ± 0.01*	**0.92 ± 0.01**	**0.97 ± 0.00**

		Sentiment Analysis							
		Emotion Detection				Polarity Detection			
model	**vec_rep**	**TweetEval**	**CARER**	**Silicone**	**MELD**	**IMDb**	**SST2**	**Movie Review**	**CR**
LR	**FT**	*0.65 ± 0.01*	***0.61 ± 0.00***	*0.23 ± 0.00*	*0.28 ± 0.01*	0.85 ± 0.00	0.80 ± 0.00	0.76 ± 0.01	0.76 ± 0.01
	TF-IDF	*0.62 ± 0.02*	*0.85 ± 0.01*	***0.30 ± 0.01***	*0.27 ± 0.00*	*0.87 ± 0.01*	0.79 ± 0.02	0.74 ± 0.01	0.74 ± 0.01
SVM	**FT**	*0.64 ± 0.02*	*0.62 ± 0.00*	***0.31 ± 0.00***	*0.29 ± 0.00*	0.85 ± 0.00	0.81 ± 0.00	0.76 ± 0.01	0.76 ± 0.01
	TF-IDF	*0.59 ± 0.01*	*0.83 ± 0.02*	***0.38 ± 0.01***	*0.27 ± 0.01*	*0.86 ± 0.02*	0.78 ± 0.01	0.73 ± 0.01	0.74 ± 0.03
RF	**FT**	*0.52 ± 0.02*	*0.34 ± 0.01*	***0.37 ± 0.01***	*0.21 ± 0.01*	0.81 ± 0.00	0.77 ± 0.01	0.75 ± 0.01	0.71 ± 0.02
	TF-IDF	*0.53 ± 0.02*	***0.81 ± 0.01***	**0.39 ± 0.01**	*0.25 ± 0.01*	0.83 ± 0.01	0.79 ± 0.01	0.74 ± 0.01	0.69 ± 0.02
XGB	**FT**	*0.64 ± 0.01*	*0.58 ± 0.02*	*0.28 ± 0.11*	***0.25 ± 0.02***	0.85 ± 0.00	0.79 ± 0.02	0.76 ± 0.01	0.76 ± 0.01
	TF-IDF	*0.55 ± 0.03*	*0.81 ± 0.13*	***0.27 ± 0.02***	***0.21 ± 0.01***	*0.87 ± 0.00*	0.78 ± 0.01	0.70 ± 0.01	0.69 ± 0.02
BiLSTM		*0.66 ± 0.01*	*0.85 ± 0.08*	***0.34 ± 0.02***	*0.29 ± 0.01*	***0.90 ± 0.00***	***0.84 ± 0.01***	*0.79 ± 0.01*	**0.78 ± 0.01**
CNN		*0.61 ± 0.03*	*0.81 ± 0.01*	***0.26 ± 0.01***	*0.26 ± 0.01*	***0.89 ± 0.00***	0.81 ± 0.01	0.76 ± 0.01	0.75 ± 0.01
RoBERTa		**0.76 ± 0.01**	**0.87 ± 0.00**	*0.19 ± 0.01*	**0.40 ± 0.01**	**0.93 ± 0.00**	**0.89 ± 0.02**	**0.87 ± 0.01**	0.56 ± 0.03

		Sentiment Analysis			
		Sarcasm Detection			
model	**vec_rep**	**iSarcasm**	**SemEval A**	**SNH**	**GENsarc**
LR	**FT**	0.46 ± 0.01	0.62 ± 0.012	0.76 ± 0.00	0.72 ± 0.01
	TF-IDF	*0.55 ± 0.01*	**0.64 ± 0.01**	0.91 ± 0.04	0.71 ± 0.01
SVM	**FT**	0.47 ± 0.01	0.61 ± 0.01	0.76 ± 0.00	0.72 ± 0.01
	TF-IDF	0.54 ± 0.03	0.63 ± 0.01	0.90 ± 0.03	0.70 ± 0.02
RF	**FT**	0.48 ± 0.01	0.60 ± 0.01	0.95 ± 0.00	0.71 ± 0.01
	TF-IDF	0.49 ± 0.02	0.62 ± 0.01	**0.96 ± 0.00**	0.70 ± 0.01
XGB	**FT**	0.49 ± 0.02	0.60 ± 0.01	0.94 ± 0.03	0.72 ± 0.01
	TF-IDF	**0.55 ± 0.02**	0.61 ± 0.01	0.84 ± 0.02	0.68 ± 0.01
BiLSTM		0.55 ± 0.02	0.63 ± 0.01	0.93 ± 0.01	*0.73 ± 0.01*
CNN		0.47 ± 0.00	0.62 ± 0.03	0.93 ± 0.01	0.72 ± 0.01
RoBERTa		0.46 ± 0	0.44 ± 0.02	0.90 ± 0.00	**0.79 ± 0.01**

Table 5: Results F1 performance. The best performances are underlined and put in bold face. We did Wilcoxon signed-rank test to check pairwise differences between the results. The italic results do not significantly differ from the best result with a confidence interval of 95%. The bold numbers do not differ with a 90% confidence interval.

Finally, for sarcasm detection, it is noteworthy that the performance of RoBERTa is significantly worse than the best-performing method for all datasets, except GENsarc, where it is the best-performing model. When analyzing the predictions made by RoBERTa for the iSarcasm and SemEvalA datasets, it is observed that all trained models almost always predict the majority class. For the iSarcasm dataset, we conclude that XGB TF-IDF significantly outperforms all other methods except LR TF-IDF and that LR TF-IDF performs significantly better than all other methods on the SemEvalA dataset with a confidence level of 95%. Finally, RF TF-IDF significantly outperforms all methods with a confidence level of 95%.

4.2. Statistical Tests

4.2.1. General Rankings

Table 6 shows the main ranking and a ranking per performance metric of the different methods across all datasets based on the post-hoc Nemenyi test after rejecting the null hypothesis of the non-parametric Friedman test. Two general rankings are given: one where all performance metrics were combined (General) and one were all performance metrics except AUC and AUCPR are included (General v2). As the AUC and AUCPR performance metrics are only provided for the binary classification tasks, the results are disproportionally influenced by these metrics for the general ranking. We underlined the best ranking and put them in bold face. Moreover, we indicated the rankings that do not significantly differ from this best-ranked method in bold, that is where the difference between the mean rank is not greater than the critical distance of the Nemenyi test. These tests are all conducted with a confidence level of 95%.

		General	General v2	ACC	F1	Precision	Recall	AUC	AUCPR
LR	**FT**	6.55	6.56	7.6	**6.4**	7.2	**5.05**	**6.82**	**3.73**
	TF-IDF	**4.65**	**4.83**	**5.7**	**4.2**	**5.75**	**3.65**	**4.27**	**6.18**
SVM	**FT**	6.42	6.44	7.35	**6**	**6.95**	**5.45**	**6.55**	**6.18**
	TF-IDF	5.74	5.59	**6.2**	**4.9**	**6.5**	**4.75**	**6.18**	**6.36**
RF	**FT**	8.25	8.29	8.2	8.6	7.4	8.95	8.27	7.91
	TF-IDF	6.08	6.23	**6.05**	**6.55**	**5.1**	7.2	**5.55**	**5.55**
XGB	**FT**	6.55	6.9	6.45	7.15	**6.3**	7.7	**5.55**	**5**
	TF-IDF	7.45	7.4	7	7.95	**6.45**	8.2	7.73	7.55
BiLSTM		**3.31**	**3.56**	**3.2**	**3.2**	**3.6**	**4.25**	**2.45**	**2.36**
CNN		5.96	6.08	**5.4**	**6.3**	**6.25**	**6.35**	**5.73**	**5.36**
RoBERTa		5.05	**4.14**	**2.85**	**4.75**	**4.5**	**4.45**	**6.91**	9.82
Critical Distance		1.5	1.69	3.38	3.38	3.38	3.38	4.56	4.56

Table 6: Overview rankings across different performance metrics. The best ranking is underlined and put in bold face. The bold results do not differ from the best result with a 95% confidence interval.

As shown in Table 6, when taking into account all performance metrics including the AUC and AUCPR, BiLSTM significantly outperforms all other methods except LR TF-IDF over all different experiments. The next in ranking are RoBERTa, SVM TF-IDF, CNN, and RF TF-IDF. Note that for these traditional machine learning methods, the FT preprocessing performs worse than the TF-IDF, and in case of the LR and RF this is a significant difference. Moreover, we can conclude that both XGB methods perform significantly worse than LR TF-IDF and BiLSTM. However, as indicated previously, some datasets are overrepresented in this ranking as AUC and AUCPR are only present for 11 out of 20 datasets. Hence, we also look into the general ranking without these two performance measures. The best-ranked method is again the BiLSTM, however, there is no significant difference between this method, RoBERTa, and LR TF-IDF. These three methods perform thus, overall best. It is a surprising finding that LR TF-IDF shows overall similar performance to sophisticated methods such as RoBERTa and BiLSTM, meaning that simple techniques can still compete with these advanced methods. In conclusion, we see that, in general, when looking at all different classification tasks, the context

of the text seems to matter. The only exception to this finding is the LR TF-IDF which shows a non-significant difference in performance compared to BiLSTM.

4.2.2. Rankings per Category

		Fake News	Topic	Emotion	Polarity	Sarcasm
LR	**FT**	8	**5.69**	**7**	**4.58**	7.46
	TF-IDF	**3.55**	**2.75**	**6.06**	6.08	**4.54**
SVM	**FT**	7.18	6.69	**6.5**	5	6.92
	TF-IDF	**4.14**	**3**	**6.38**	8.21	6.13
RF	**FT**	7.86	9.75	7.88	9	7.08
	TF-IDF	**3.64**	6.88	**5.63**	8.5	**5.67**
XGB	**FT**	7.5	9.88	**6.19**	6.04	**5.79**
	TF-IDF	**6.19**	**6.5**	**6.81**	8.75	7.13
BiLSTM		**4.55**	**4.69**	**3.38**	**1.54**	**3**
CNN		7.86	**6.44**	**6.75**	**4**	**5.33**
RoBERTa		**5.95**	**3.69**	**3.44**	**4.29**	6.96
Critical Distance		3.22	3.77	3.77	3.08	3.08

Table 7: Overview rankings across different categories. The best ranking is underlined and put in bold face. The bold results do not differ from the best result with a 95% confidence interval.

When looking further into the ranking per classification task shown in Table 7, we see that for all categories, BiLSTM is the best-ranked method, except for fake news detection and topic detection, where LR TF-IDF respectively is ranked best. However, we should note that as there were fewer observations, the critical distance grows and thus fewer conclusions can be made based on these tests, and further research is required. In the following, we present conclusions per classification task, backed by theoretical explanations and existing literature.

Fake News Detection

Regarding the fake news detection category, we can conclude that there is a significant difference between LR TF-IDF on the one hand and all FT machine learning methods and CNN on the other hand. Moreover, for all machine learning methods, we find that TF-IDF is preferred over FT for this classification task. This means that in combination with easy methods, the presence of certain words benefits the prediction of fake news more than the semantic meaning of the words. This might be due to the datasets, where certain words might be more related to fake news than to real news. Contrarily to this finding, (Gravanis et al., 2019) proposes that the semantic meaning of words is, in fact, important, and Gravanis et al. (2019) and Capuano et al. (2023) suggest that boosting methods combined with pretrained embeddings performs well for fake news detection. Nevertheless, as the difference in ranking is non-significant according to our tests, except for LR and RF, further research is required to come to a generalizable conclusion. When comparing our findings to other existing benchmarks in fake news detection, we see that Khan et al. (2021) finds that RoBERTa performs best out of all classification methods. However, note that different hyperparameters were set in their experiments. The conclusions on the performance of deep learning methods compared with machine learning methods differ widely across studies with some concluding that CNN performs significantly better (Wang, 2017) or (Bi)LSTMs that perform worse than these simple techniques (Sharma and Garg, 2021; Wang, 2017). This last conclusion shows the necessity of a good hyperparameter tuning approach for these methods.

Topic detection

Concerning topic detection, we find that LR TF-IDF is the best-performing method, followed

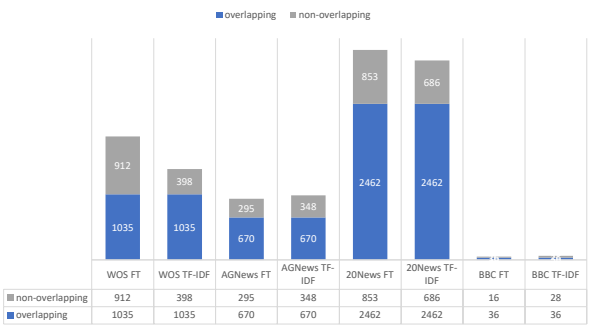
by SVM TF-IDF and only then we find RoBERTa and BiLSTM. This suggests that for this classification task, reading sentences in a way to capture words in their context might not be necessary. Simple methods like LR or SVM also perform well. Furthermore, note that for all these traditional machine learning methods, both simple and ensemble methods, the performance of FT lags behind that of TF-IDF. Moreover, a statistical difference is found between the ranking of LR TF-IDF and SVM FT, RF FT, and XGB FT with a confidence level of 95%. This is intuitive, as for detecting a topic, it is not necessary to understand the full semantic meaning of the sentence as long as the essential related words are retained. Also in existing literature, TF-IDF is found to perform well for topic detection (Rahman, 2020). However, in their study, the model with word2Vec embeddings performs slightly better. Figure 1 shows the breakdown of the misclassifications of the different machine learning methods between FT and TF-IDF. We included all the different mistakes per method and dataset over the ten random seeds and display the overlapping mistakes between the FT and TF-IDF models per dataset in blue and the non-overlapping mistakes in grey. When looking at the breakdown of misclassifications, we see that the SVM and LR act similarly with similar amounts of overlapping misclassifications and the classifiers with FT make slightly more non-overlapping mistakes than TF-IDF except for the AGNews and BBC datasets. For the ensemble methods, however, we see on the one hand for the RF models a high amount of overlapping misclassifications and similar amounts of non-overlapping mistakes among the FT and TF-IDF models. On the other hand, we find that XGB TF-IDF makes fewer different mistakes than XGB FT on the WOS, 20News, and BBC datasets, while for AGNews, XGB TF-IDF makes four times more different mistakes compared to XGB FT. This is also shown in Table 5, as the variation in performance over the ten random seeds was three times higher than for XGB FT. For WOS, however, we see that the high number of different misclassifications is not due to the variation in the performance of the different models as the standard deviation is relatively low, but because of the worse overall performance of XGB FT. Finally, our results also show that BiLSTM outperforms CNN, however it is a non-significant difference. Nevertheless, similar conclusions are found in (Rahman, 2020; Zhang and Zhang, 2020; Gutiérrez-Batista et al., 2019).

Emotion detection

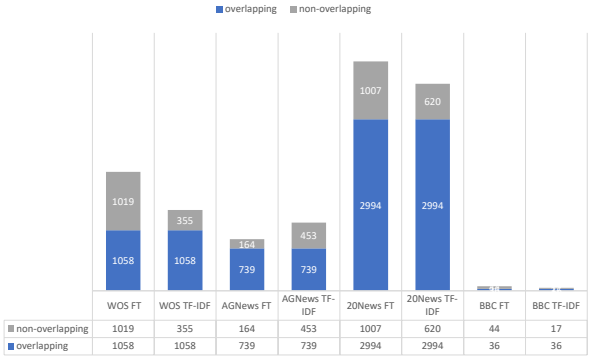
Next, the BiLSTM method shows the best ranking for emotion detection. However, no significant differences can be found between this method and the other methods at a 95% confidence level except RF FT. RoBERTa is ranked second best and only then, RF TF-IDF and LR TF-IDF can be found. This suggests that context is important to correctly predict emotion expressed in text. In the existing literature, it is shown that deep learning methods outperform simpler methods (Jin et al., 2020; Majeed et al., 2022). Furthermore, BERT-based models often outperform BiLSTM (Hasan et al., 2021). However, this is not reflected in the ranking. When we look back at the different results in Table 5, this is explained by RoBERTa's bad performance on the silicone dataset. As stated before, RoBERTa mostly predicts the majority class. Therefore, it is outperformed by all other methods on all performance metrics except accuracy and this makes the method perform worse in the overall ranking for this text classification category. Nevertheless, the ranking is still close to the ranking of BiLSTM, suggesting its good performance on this classification task.

Polarity detection

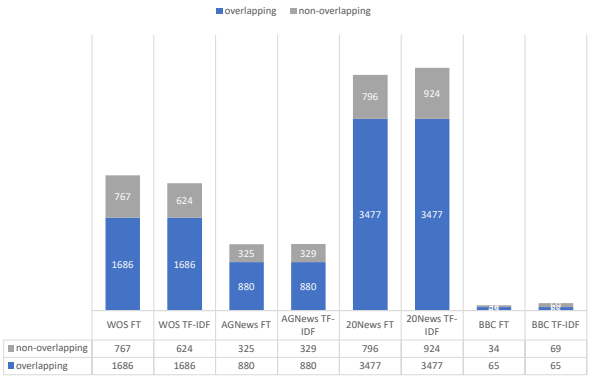
For polarity detection, the BiLSTM again ranks best. No significant difference can be found between this method and CNN, RoBERTa, and LR FT. Note that here, the semantic meaning of words is shown to be more important, as for all machine learning methods except RF, the method using FT outperforms the one using TF-IDF. Moreover, all three deep learning methods, including CNN, show a good performance. This suggests that context is important on top of the semantic meaning of individual words. Furthermore, as shown, CNN performs



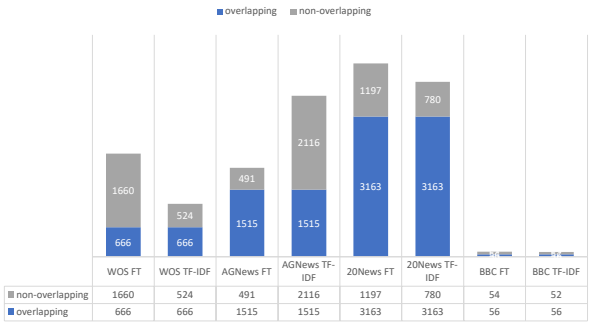
(a) Breakdown misclassifications LR



(b) Breakdown misclassifications SVM



(c) Breakdown misclassifications RF



(d) Breakdown misclassifications XGB

Figure 1: Breakdown misclassifications machine learning models FT vs TF-IDF for the three topic detection datasets.

second-best on this classification task, which is the best ranking for all classification tasks. A possible explanation for this high ranking is that three out of the four polarity datasets contain more than 10,000 sentences and deep learning outperforms machine learning methods when the dataset size grows as we show in Section 4.2.3. In literature, we also see that both methods perform similarly for polarity classification tasks, sometimes with CNN performing best (Yan et al., 2022; Lei et al., 2018), and sometimes BiLSTM (Wu et al., 2019a; Yu et al., 2017).

Sarcasm detection

For the sarcasm datasets, BiLSTM performs best and significantly better than all methods except LR TF-IDF, RF TF-IDF, XGB FT, and CNN. Note that only on this classification task, RoBERTa is significantly outperformed by the best-performing method with a confidence level of 95%. At first sight, this looks counterintuitive as understanding the full context of a text as is done using RoBERTa is necessary. Moreover, as RoBERTa is a pretrained language model, this would provide the model already with a basic understanding of words. When looking more closely at the results, for both iSarcasm and SemEValA, we find that RoBERTa predicts non-sarcastic most of the time. This is explained by an imbalanced dataset such as iSarcasm. However, as SemEvalA is not imbalanced, but a small dataset, it suggests that the fact that RoBERTa is pretrained on large amounts of mostly non-sarcastic texts is disadvantageous for the models' performance, especially when finetuning the model using a small dataset. Furthermore, we see that also for SNH RoBERTa is not among the best-performing methods. However, the good performance on the GENsarc dataset, shows that RoBERTa can possibly perform very well on sarcasm detection tasks. As also in literature it is found that transformer-based models perform well (Kayalvizhi et al., 2019), we suggest to further research this and include more datasets. Similar to our findings, existing literature does find that among the machine learning

methods RF (Charalampakis et al., 2016) and LR (Khatri and Pranav, 2020; Razali et al., 2021) perform well. Moreover, note that TF-IDF is ranked better than FT for all methods.

In conclusion, it is noteworthy that, across all tasks except polarity detection, TF-IDF consistently performs better than FT. Although the deep learning models were also trained using an embedding layer based on FT, this embedding layer was further optimized together with the other layers of the models, which might explain the additional advantage that was absent in the machine learning models by fine-tuning it on the particular dataset. Moreover, note that these models also comprise context. The FT models also generate sentence embeddings, so it is also context-aware, but to a smaller extent than the deep learning models.

4.2.3. Rankings per Dataset Specifications

We conduct statistical tests to find patterns in the best-performing methods regarding the dataset specifications, more specifically the size of the dataset and the number of target labels. The lower the value for a dataset-classifier combination, the better the average ranking across this combination. These rankings are shown in Table 8. Firstly, we evaluate models trained on a similar number of target labels and rank them accordingly. For both the binary and multi-class datasets, we see that BiLSTM performs best. However, no significant difference is observed between this method and LR TF-IDF for both groups. For binary datasets, no significant difference is found between BiLSTM and CNN, while for the multi-class datasets, SVM TF-IDF, RF TF-IDF, and RoBERTa could also not be distinguished as significantly different from BiLSTM. However, we want to emphasize that this split is also closely related to the classification task, as emotion detection and topic detection are always multi-class datasets.

Secondly, we split the datasets based on the dataset size. As the critical distance grows, we find fewer significant differences. However, it is interesting that for the group with the smallest dataset size, RoBERTa is the best-ranked method followed by the LR TF-IDF. As RoBERTa is a pretrained model, it does not need as much training to perform well as other deep learning methods such as CNN and BiLSTM require. Moreover, for these small datasets, the probability of overfitting grows, as the number of features might be larger than the number of topics, LR is better suited for such problems than for example SVM (Thangaraj and Sivakami, 2018). For the two other dataset sizes, we see that BiLSTM outperforms the other methods. Moreover, for the middle-sized datasets, we find a significant difference between BiLSTM on the one hand and CNN, RF FT, and XGB FT on the other hand. Furthermore, our finding that the higher the size of the dataset, the worse the performance of the traditional machine learning models, is also concluded in (Zhang et al., 2015). Additionally, note that TF-IDF would come at a higher computational cost as the dataset size grows (Aka Uymaz and Kumova Metin, 2022)

4.3. Overall Performance Trade-offs

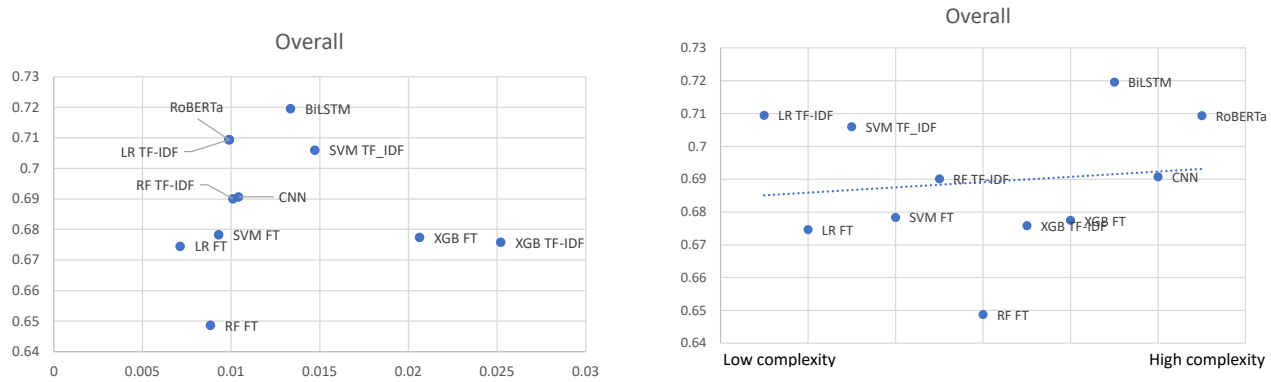
In addition to evaluating the models on their performance rankings, it is essential to also consider other aspects, such as the trade-off between performance and variance as well as the trade-off between performance and complexity. These are discussed in detail in the next paragraphs and are depicted in Figure 2.

Performance-Variance trade-off. This trade-off is illustrated by plotting the average F1 performance against the standard deviation of the performance across the different methods. We find similar variability in F1 results for all methods, except for BiLSTM and SVM TF-IDF showing on average a higher variance, and both XGB methods displaying the most significant variability overall. Consequently, when requiring stable results, these last two models are less appropriate.

Performance-Complexity trade-off. To map the complexity of the different models, we rank them based on the performance-complexity trade-off as outlined in Arrieta et al. (2020).

		Binary	Multiclass	$<$ 10k	$<$ 50k	$\geq$ 50k
LR	**FT**	6.73	**6.22**	**6.16**	**5.75**	7.92
	TF-IDF	**4.58**	**4.78**	**4.34**	**5.54**	**4.75**
SVM	**FT**	6.44	6.39	**5.75**	6.25	7.54
	TF-IDF	6.23	**4.83**	**5.13**	**5.71**	6.08
RF	**FT**	8.08	8.56	7.97	8.96	8.04
	TF-IDF	6.06	**6.11**	7.22	6.25	**4.92**
XGB	**FT**	5.85	7.83	**6.06**	7.17	7.75
	TF-IDF	8.05	6.36	8.31	**5.96**	7.63
BiLSTM		**3.03**	**3.83**	**4.19**	**3.17**	**2.79**
CNN		5.5	6.81	**6.59**	7.21	**4.25**
RoBERTa		5.47	**4.28**	**4.03**	**4.04**	**4.38**
Critical Distance		1.86	2.52	2.67	3.08	3.08

Table 8: Overview rankings across different dataset specifications. The best ranking is underlined and put in bold face. The bold results do not differ from the best result with a 95% confidence interval.



(a) Overall Performance-Variance trade-off

(b) Overall Performance-Complexity trade-off

Figure 2: Overview overall trade-offs

This ranking considers factors such as the models' trainable parameters, the complexity, and the computational time and resources required for training. Considering the significant differences in computational demands and trainable parameters across the methods we employed, we preferred this ranking rather than attempting to quantify the exact number of trainable parameters per method. This decision is especially justified for tree-based methods like random forests, where the number of trainable parameters is also influenced by the depth of the individual trees.

Moreover, since various models require different computing resources -CPU vs GPU- comparing their training times is not straightforward. LR and SVM are shown to have the lowest complexity with their number of trainable parameters depending on the number of input features. RF and XGB are more complex due to their methodology of training multiple decision trees for data classification. This makes them less intuitive, with the number of trainable parameters depending on both the number of trees estimated and the depth of each tree. Next, deep learning approaches are found to be more complex than the previous aproaches.Table 9 shows a clear difference in complexity for an attention layer, convolutional layer, and recurrent layer. Given this table, BiLSTM is assessed to have lower complexity than CNNs. However, CNNs benefit from easier parallelization, which will significantly reduce the computational time required. Given that self-attention is an important component of RoBERTa, which scales

Layer type	Complexity
Recurrent	$O(nd^2)$
Convolutional	$O(knd^2)$
Self-attenion	$O(n^2d)$

Table 9: Different complexities per layer type, n is the sequence length, d is the representation dimension, k equals the kernel size. (Vaswani et al., 2017)

quadratically with the input size, this method is considered the most complex within our compared models. Furthermore, given that FT also stems from a deep learning approach, we rank FT embeddings as more complex than TF-IDF. Nevertheless, since this is only adopted in the preprocessing phase, we still assume that this does not impact the complexity ranking of other ML tasks.

Figure 2b shows this trade-off and illustrates an upward trend in F1 performance with higher complexity. This shows that overall, using these complex methods thus results in higher performance. However, it should be noted that our simplest method, LR TF-IDF, also shows a relatively high performance, despite being the least complex method. This finding is in line with our previous findings and shows that depending on the application, sometimes less complex methods can be beneficial.

4.4. Performance Trade-offs per Classification task

Next, we look into the different trade-offs per classification task: *fake news classification, topic modeling, emotion detection, polarity detection, and sarcasm detection.* These trade-offs are discussed in the following paragraphs.

Performance-Variance trade-off. Given the benefit of stable performance of a model trained for a text classification task, Figure 3 visualizes the trade-off between performance and variance in the different results. This figure shows the relation between averaged F1 performances and the standard deviations per classification task for every method. Consistent with our findings in Table 7, Figure 3a also indicates LR TF-IDF as the preferred method for fake news classification. It maintains an average standard deviation comparable to RoBERTa and RF FT, however with a higher F1 measure. Nevertheless, when the lowest variance is favored, BiLSTM can be a potential alternative, despite its lower performance.

The trade-off for topic modeling is shown in Figure 3b. LR FT presents the lowest average standard deviation, however, it also shows a mediocre performance, relative to the other models. LR TF-IDF, on the other hand, continues to show a well-balanced trade-off between performance and variance for this classification task. It is noteworthy that XGB shows high variability in its performance. This combined with its low performance, makes the method a less desirable choice for this classification task.

For emotion detection, we find a distinct cluster on the left of Figure 3c, with the best-performing model, RoBERTa, also showing a low average standard deviation for this task. This underscores the high potential of this method. Moreover, also LR TF-IDF shows a favorable balance between performance and variance. However, the trained XGB models display again a high variability in the performance across different models, similar to BiLSTM for this classification task. Contrary to this finding, for polarity detection, BiLSTM demonstrates low variance and is the best-performing model. RoBERTa while being ranked second in terms of average F1 performance, displays a higher variance than most of the other models. Finally, in Figure 3e, RoBERTa is noted for its relatively low variance but also exhibits a low performance. However, BiLSTM shows better performance while maintaining a medium variance compared

to the other methods. In this case, BiLSTM is distinctly preferred over LR TF-IDF among others, offering a higher performance combined with a lower variance.

Performance-Complexity trade-off. In Figure 4, the trade-off between complexity and F1 performance is illustrated: it shows average F1 scores per model ranked according to complexity. For fake news detection and topic modeling, we find a downward trend in the performance of the models with increasing complexity. Although this might seem counterintuitive, we have previously explained this phenomenon, noting that for these datasets, the identification of certain words seems to suffice to correctly classify the sentences. For the remaining three tasks, however, we do find a positive correlation between model complexity and performance. Especially for emotion and polarity detection, this trend is very pronounced, due to the necessity of a deeper understanding of the sentences within these tasks. For the sarcasm detection task, however, this increase is less pronounced, likely due to the low performance of RoBERTa on the imbalanced datasets.

4.5. Performance Trade-offs per Dataset Specifications

Similarly to the performance analysis, we continue the analysis focusing on the different dataset specifications: *binary, multi-class, <10k, <50k, ≥50k*. We provide the figures in Appendices 6.4 and 6.5. The main findings are described in the following paragraphs.

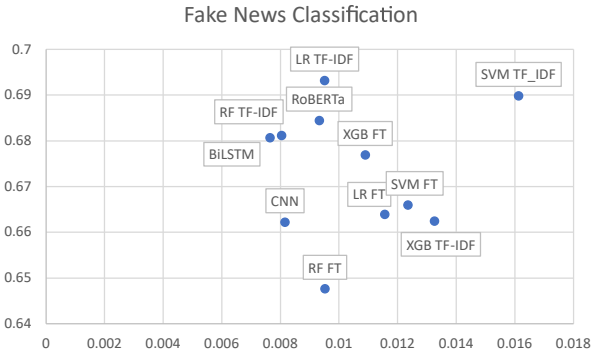
Performance-Variance trade-off. We find that for the binary datasets, BiLSTM shows low variance, while being the best-performing model. However, its variance increases for the multi-class datasets, where RoBERTa shows a lower variance and also high performance. Across all different dataset specifications, XGB displays a high variance in performance. Furthermore, for both the datasets with less than 10,000 and more than or equal to 50,000 examples, we find that BiLSTM again shows a relatively low variance, while obtaining high performance. In the category of fewer than 50,000 examples, we see this phenomenon occurring for RoBERTa and LR TF-IDF.

Performance-Complexity trade-off. Across almost all different dataset specifications, we observe an increasing trend, indicating that an increase in model complexity correlates with enhanced performance. However, an exception is shown for the category including the datasets with a dataset size lower than 10,000, which shows a downward trend. In this case, LR TF-IDF is preferred, likely due to the need for more examples to effectively train the more complex methods.

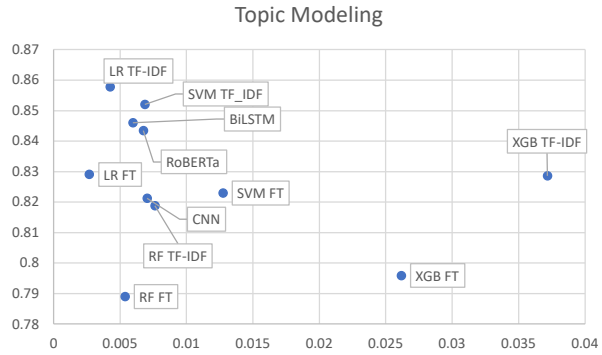
4.6. Summary of the findings

In Table 10 an overview of our general findings is provided. This table shows the overall best-performing method and the best-performing method per text classification task and dataset specification. Moreover, we also provide the best-performing machine learning model per task. Note that often the best-performing model does not significantly differ from the other models. We refer the reader to previous sections for the details.

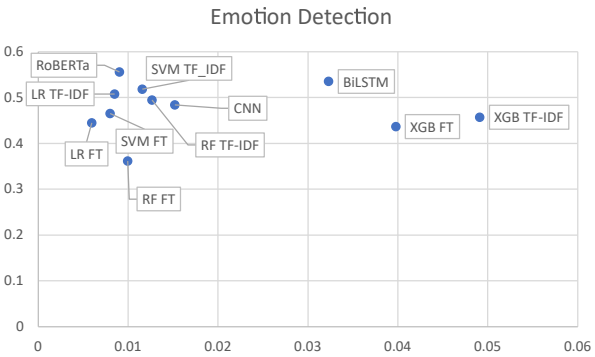
As shown, BiLSTM is most often the best-performing method, except for the topic classification and fake news detection datasets where LR TF-IDF is the best-performing model and the smallest datasets where RoBERTa is the best-performing model. In terms of best-performing machine learning models, we see that LR is always the best method except for emotion detection where RF TF-IDF is preferred. Moreover, LR FT is only preferred for a dataset size of 10,000 to 50,000 data points and polarity detection classification tasks. On top of the overview of the best-performing models, we have also added for the different classification tasks and different dataset specifications, the model with the lowest variance, as well as whether there is a positive or negative correlation between performance and complexity. The combination



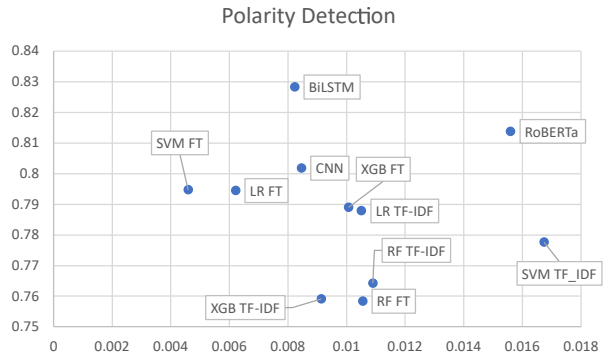
(a) Performance-Variance trade-off Fake News Classification



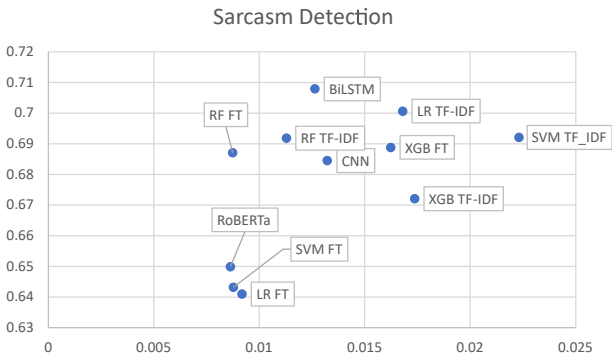
(b) Performance-Variance trade-off Topic Modeling



(c) Performance-Variance trade-off Emotion detection

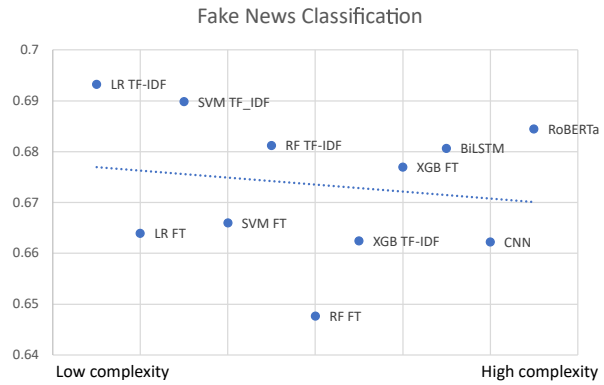


(d) Performance-Variance trade-off Polarity Detection

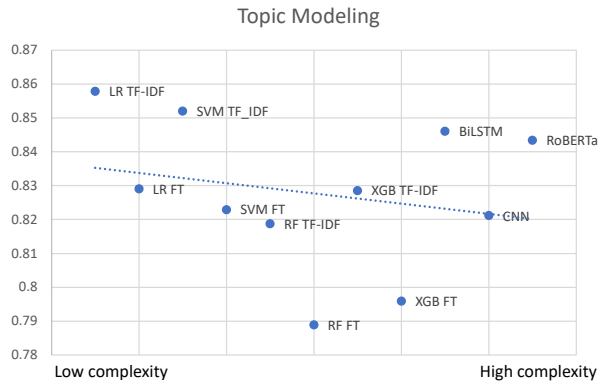


(e) Performance-Variance trade-off Sarcasm Detection

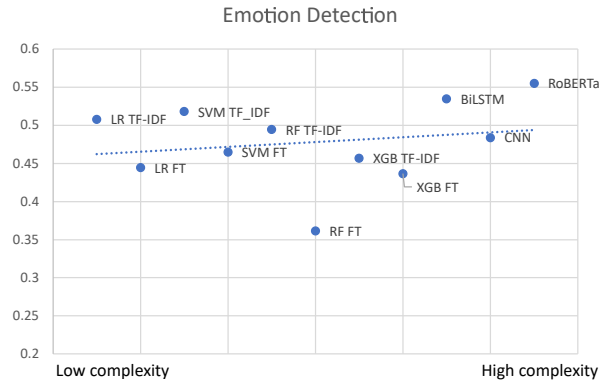
Figure 3: Overview Performance-Variance trade-off per classification task.



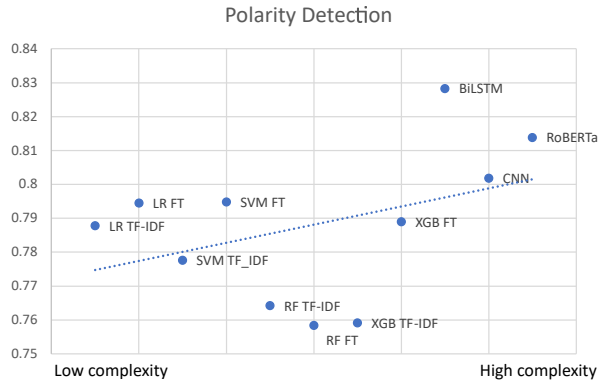
(a) Performance-Complexity trade-off Fake News Classification



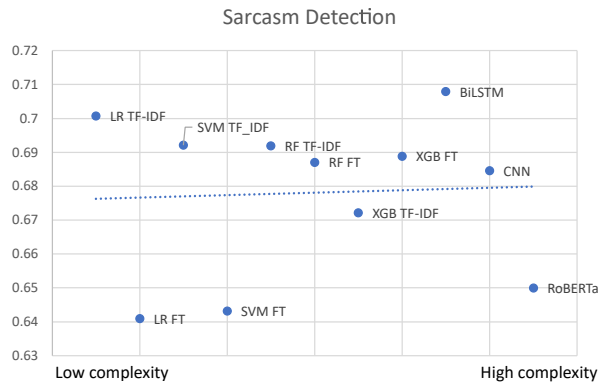
(b) Performance-Complexity trade-off Topic Modeling



(c) Performance-Complexity trade-off Emotion detection



(d) Performance-Complexity trade-off Polarity Detection



(e) Performance-Complexity trade-off Sarcasm Detection

Figure 4: Overview Performance-Complexity trade-off per classification task.

	Task	Best method	Best machine learning method	Lowest variance	Correlation Performance - Complexity
	Overall	BiLSTM	LR TF-IDF	LR FT	Positive
Text Classification Task	**Fake news**	LR TF-IDF	LR TF-IDF	BiLSTM	Negative
	Topic	LR TF-IDF	LR TF-IDF	LR FT	Negative
	Emotion	BiLSTM	RF TF-IDF	LR FT	Positive
	Polarity	BiLSTM	LR FT	SVM FT	Positive
	Sarcasm	BiLSTM	LR TF-IDF	RoBERTa	Positive
Dataset Specifications	**Binary**	BiLSTM	LR TF-IDF	SVM FT	Positive
	Multi-class	BiLSTM	LR TF-IDF	LR FT	Positive
	<10k	RoBERTa	LR TF-IDF	RoBERTa	Negative
	< 50k	BiLSTM	LR TF-IDF	LR TF-IDF	Positive
	≥ 50k	BiLSTM	LR TF-IDF	LR FT	Positive

Table 10: Summary findings.

of these different factors can help in the decision-making when choosing the most appropriate method to use for the specific application. More details are provided in the previous sections.

5. Conclusion

In this paper, we conduct an extensive, impartial benchmark on five different text classification tasks: fake news detection, topic classification, emotion detection, polarity detection, and sarcasm detection. After extensive hyperparameter tuning, we train models of varying complexity using twenty frequently used datasets over these five text classification tasks. Furthermore, we thoroughly analyze the results of these different models and compare them using statistical testing techniques. We used both theoretical explanations and existing literature to solidify our findings. Finally, we highlight the critical trade-offs between performance-variance and performance-complexity which are key factors in selecting the most suitable method. Given the growing concern regarding the ecological footprint of using extensive computational resources, this benchmark study offers valuable insights by giving guidance on when less computationally expensive methods are more appropriate than more complex methods.

We find that BiLSTM is the overall best-ranked method, and it significantly outperforms all other methods except LR TF-IDF, and RoBERTa with a confidence level of 95%. As no significant difference between these models is detected, overall, this indicates that the computational effort put into training deep learning models is not justified for text classification tasks. When looking into the overall F1 performance-complexity trade-off, we do see however a positive correlation. In terms of the performance-variance trade-off, we see similar variance for both LR TF-IDF and RoBERTa. Both models also show a lower variance than BiLSTM. However, the best-performing technique does depend on the text classification task at hand. For the fake news detection and topic detection tasks, LR TF-IDF is the best-ranked method, suggesting that identifying certain words is enough for these classification tasks. This finding is further backed by the decreasing trend in the performance-complexity trade-off for these classification tasks. In terms of the variance, we see that all models perform similarly for these two tasks, except SVM and XGB for fake news classification and the latter model for topic modeling. BiLSTM and RoBERTa are both performing very well for emotion detection datasets. RoBERTa is ranked second-best, because of its bad performance on one of the four datasets. The three deep learning methods are best suited for polarity detection datasets. Moreover, for this classification task, the semantic meaning of words is important as FT outperforms TF-IDF.

For sarcasm detection, we again find that BiLSTM works best, followed by LR TF-IDF and CNN. RoBERTa, however, does not work well for sarcasm detection. Nevertheless, all three classification tasks do show a positive correlation between performance and complexity. XGB shows again high variance compared to the other methods for emotion detection and sarcasm detection. For the latter classification task, we again see a high variance for SVM TF-IDF. Nevertheless, also the more complex methods show high variance in performance, e.g. BiLSTM for emotion detection and RoBERTa for polarity detection.

When looking into the dataset specifications, we find that BiLSTM and LR TF-IDF perform well for binary classifications, while multi-class classification problems are best solved using BiLSTM, RoBERTa, or LR TF-IDF. In terms of the dataset size, we see that RoBERTa is the best-performing method for the smallest datasets closely followed by LR TF-IDF. The larger the dataset, however, the worse the performance of the machine learning methods compared to the deep learning methods. For the middle-sized datasets, Bidirectional LSTMs and RoBERTa perform best and for the largest datasets, in addition to these two methods, CNNs also perform well. These findings also correspond to the ones in the performance-complexity trade-off, where all dataset specifications showed an increasing trend except for the datasets with less than 10,000 examples, where the less complex methods are more beneficial. Furthermore, for the different dataset specifications, we find that often the XGB methods and SVM TF-IDF show high variance compared to the other methods.

This paper emphasizes that the most sophisticated methods may not always be the optimal choice, especially when taking into account complexity or explainability. It also underscores the significance of tailoring the model selection based on the dataset specifications and the text classification task at hand. Our research lays the foundation of a useful tool to help decide what method is best suited for a specific text classification problem and dataset specification. This helps future research in choosing the right method for the task that needs to be solved. Further extending this study is necessary, however, to distinguish when the best-performing model depends on the dataset specification and when it depends on the classification task. Moreover, we did not look into feature selection methods in our research, which would prove to be interesting to include, especially for the larger datasets. Some applied methods are computationally expensive, especially BiLSTM which is the overall best-performing method. It would be interesting to focus on the effects of feature selection techniques on the performance of this method and compare them to other implemented methods. In practice, this would also provide helpful solutions for a lack of computing power. Next, more dimensions could be added to the performance evaluation, such as robustness, which is also an important criterion to take into account when evaluating the models' performance. These additional measures further provide interesting inputs for deciding upon the best-suited method for the classification problem, depending on the requirements of the task. Finally, we think it might be valuable to further look into the coverage of mistakes made by different models and see whether stacking different models might be interesting to further boost the performance.

Acknowledgements: The resources and services used in this work were provided by the VSC (Flemish Supercomputer Center), funded by the Research Foundation - Flanders (FWO) and the Flemish Government.

Funding: This research was funded by the Statistics Flanders research cooperation agreement on Data Science for Official Statistics.

6. Appendix

6.1. Extra information datasets

Classification Task		Dataset	Description
Sentiment Analysis	Fake News Classification	Gossipcop	This dataset includes titles of news articles that are either fake or true. This dataset was gathered using Gossipcop, a website for fact-checking entertainment stories combined from several sources.
		CoAID	This dataset includes articles about COVID-19 misinformation. The dataset is split into real and fake articles and includes news from websites and social platforms.
		LIAR	This dataset contains human-labeled statements from Politifact. The statements are rated by a Politifact editor in terms of truthfulness on a scale of 0 to 5.
		McIntire	This dataset contains fake news articles from another fake news dataset[1] including articles about the US elections of 2016 and real news articles gathered from All Sides
	Topic Classification	20News	This dataset contains several newsgroup posts split into 20 different categories.
		AGNews	This dataset contains articles from several different news sources. The articles are classified into the following categories: World, Sports, Business, and Sci/Tech
		WOS	This dataset contains data and metadata about published articles from Web Of Science.
		BBC	This dataset contains several different BBC articles that are categorized into 5 different categories: business, entertainment, politics, sport, tech
	Emotion Detection	TweetEval	This dataset contains 7 different tweet classification tasks in total. One of the tasks is the emotion detection task which classifies tweets into 4 different emotions: anger, joy, sadness, optimism.
		CARER	CARER contains tweets that are classified into 6 different categories: anger, fear, joy, love, sadness, and surprise.
		Silicone	This dataset contains several different tasks all meant for training, evaluating, and analyzing language understanding in models. One of the different tasks contained in this dataset is emotion detection. Classifying the data in 7 different classes: disgust, fear, happiness, no emotion, sadness, and surprise.
		MELD	This dataset contains utterances of dialogues from the series 'Friends'. Besides text data, the dataset also contains audio and visual data. For our experiments, we only used the textual data. The data is classified into 7 different categories: anger, disgust, sadness, joy, neutral, surprise, and fear.
	Polarity Detection	IMDb	This dataset contains movie reviews from IMDb and includes two labels: positive and negative.
		SST2	This dataset contains single sentences from movie reviews. The sentences are annotated by human annotators and include binary labels: positive and negative.
		Movie Review	This dataset contains positive and negative movie reviews.
		CR	This dataset contains customer reviews and every example includes into positive or negative labels.
	Sarcasm Detection	iSarcasm - English	This dataset includes text fragments that are labeled as either sarcastic or non-sarcastic. These labels are provided by the author of the texts and thus represent the intended meaning of the texts.
		SemEval A	This dataset contains tweets that are labelled into two categories: sarcastic or non-sarcastic
		SNH	This dataset contains news headlines collect from two news sources. One of these news sites: TheOnion makes sarcastic news headlines. The non-sarcastic news headlines come from HuffPost.
		GENsarc	This dataset is part of the Sarcasm Corpus V2 and is the largest dataset from the corpus containing sentences that are classified as generic sarcastic or non-sarcastic posts.

[1] https://www.kaggle.com/datasets/mrisdal/fake-news

Table 11: Overview of dataset specifications.

6.2. Hyperparameter settings BiLSTM, CNN

In Table 12 and Table 13, an overview is provided of several studies using CNNs and BiLSTM including their hyperparameter tuning settings. When a certain value is not clearly reported in a study, we indicate this with a '?'.

Reference	**Method**	**Optimizer**	**Learning Rate**	**Drop-out**	**Batch size**	**Hidden Layers**	**Hidden Layer Size**
(Vernikou et al., 2022)	BiLSTM	Adam	[0.00001,0.01]	?	[8, 128]	1	?
(Kaliyar et al., 2020)	BiLSTM	Adam	0.2	0.2	64	?	?
(Agrawal et al., 2022)	BiLSTM	Adam	0.001	?	64	1	?
(Sharma and Garg, 2021)	BiLSTM	Adam	?	0.3	64	1	50
(Ilie et al., 2021b)	BiLSTM	Adam	0.0001	?	4	1	256
(Sachan et al., 2019)	BiLSTM	Adam	0.001	0.5	[2000, 15000]	1	512
(Yan et al., 2022)	BiLSTM	?	0.00005	0.2	32	?	768
(Mohammed and Kora, 2022)	LSTM	Adam	0.001	0.7	{400, 200, 10}	{1,2}	256
(Jindal et al., 2020)	BiLSTM	?	?	?	?	2	32
(Sharma and Garg, 2021)	BiLSTM	Adam	?	0.3	64	1	50
(Khan et al., 2021)	BiLSTM	Adam	0.001	/	128	?	?
(Jin et al., 2020)	BiLSTM	Adam	0.001	0.5	100	1	100
(Chandra and Krishna, 2021)	(Bi)LSTM	?	?	0.65	?	2	128 and 64

Table 12: Overview hyperparameter settings (Bi)LSTM.

Reference	Method	Optimizer	Learning Rate	Drop-out	Batch size	kernel size	filter size
(Kim, 2014)	CNN	?	?	0.5	50	{ 3, 4 , 5}	100
(Kaliyar et al., 2020)	CNN	Ada-delta	0.001	/	128	?	?
(Li et al., 2020a)	CNN	Adam	0.001	0.5	64	{ 3, 4, 5}	64
(Tan et al., 2021)	CNN	Adam	0.001	0.5	64	{ 3, 4, 5}	512
(Ilie et al., 2021a)	CNN	Adam	0.0001	?	4	{ 3, 5, 7}	16
(Yan et al., 2022)	CNN		0.00005	0.2	32	{2,3,4}	256
(Choudhary et al., 2021)	CNN	Adam	0.001	?	100	{2,3,4,5}	4
(Jindal et al., 2020)	CNN	?	0.3	?	64	{ 3, 4, 5}	?
(Galli et al., 2022)	CNN	Adam	?	?	32	2	?
(Wang, 2017)	CNN	?	?	0.8	64	{2,3,4}	128
(Khan et al., 2021)	CNN	?	?	0.8	{64, 512}	3	128
(Li et al., 2020b)	CNN	Adam	?	0.5	?	{1,2,3}	128
(De Caigny et al., 2020)	CNN	Adam	?	0.5	50	{ 3, 4 , 5}	100

Table 13: Overview hyperparameter settings CNN.

6.3. Results different performance measures performance

In Tables 14, 15, 16, 17 and 18, the average performance scores of the trained models are reported per dataset in terms of the accuracy, macro precision, macro recall, AUC, and AUCPR metrics, respectively.

		Fake News Classification				Topic Modeling			
model	**vec_rep**	**Gossipcop**	**CoAID**	**LIAR**	**McIntire**	**20News**	**AGNews**	**WOS**	**BBC**
LR	**FT**	0.76 ± 0.01	*0.92 ± 0.02*	***0.23 ± 0.01***	0.82 ± 0.01	*0.63 ± 0.00*	*0.89 ± 0.00*	*0.84 ± 0.00*	*0.97 ± 0*
	TF-IDF	***0.82 ± 0.00***	0.96 ± 0.01	*0.23 ± 0.00*	0.83 ± 0.01	**0.67 ± 0.00**	*0.91 ± 0.00*	*0.89 ± 0.00*	*0.97 ± 0.01*
SVM	**FT**	0.77 ± 0.01	*0.92 ± 0.02*	***0.23 ± 0.01***	0.82 ± 0.01	*0.61 ± 0.01*	*0.89 ± 0.00*	*0.84 ± 0.03*	*0.96 ± 0.01*
	TF-IDF	***0.81 ± 0.01***	0.96 ± 0.00	*0.23 ± 0.01*	0.82 ± 0.02	*0.64 ± 0.02*	*0.91 ± 0.01*	*0.89 ± 0.00*	*0.97 ± 0*
RF	**FT**	0.80 ± 0.02	***0.96 ± 0.01***	***0.24 ± 0.01***	0.8 ± 0.01	*0.57 ± 0.00*	*0.88 ± 0.00*	*0.8 ± 0.01*	*0.94 ± 0.01*
	TF-IDF	***0.84 ± 0.01***	***0.96 ± 0.01***	***0.25 ± 0.01***	*0.83 ± 0.01*	*0.63 ± 0.00*	*0.91 ± 0.01*	*0.83 ± 0.01*	*0.93 ± 0.01*
XGB	**FT**	0.83 ± 0.01	*0.96 ± 0.01*	*0.24 ± 0.01*	0.81 ± 0.01	*0.58 ± 0.03*	*0.87 ± 0.04*	*0.82 ± 0.03*	*0.94 ± 0*
	TF-IDF	***0.84 ± 0.01***	*0.92 ± 0.02*	***0.25 ± 0.01***	0.8 ± 0.01	*0.60 ± 0.01*	*0.87 ± 0.11*	*0.91 ± 0.02*	*0.94 ± 0.01*
BiLSTM		**0.84 ± 0.01**	***0.95 ± 0.00***	**0.25 ± 0.01**	0.82 ± 0.01	*0.63 ± 0.01*	**0.93 ± 0.00**	0.91 ± 0.01	*0.94 ± 0.01*
CNN		0.83 ± 0.01	***0.94 ± 0.01***	*0.24 ± 0.01*	0.81 ± 0.01	*0.57 ± 0.01*	*0.91 ± 0.00*	*0.88 ± 0.01*	*0.95 ± 0.01*
RoBERTa		***0.84 ± 0.01***	**0.96 ± 0.00**	*0.23 ± 0.01*	**0.91 ± 0.01**	*0.63 ± 0.00*	*0.89 ± 0.01*	**0.92 ± 0.01**	**0.98 ± 0**

		Sentiment Analysis							
		Emotion Detection				Polarity Detection			
model	**vec_rep**	**TweetEval**	**CARER**	**Silicone**	**MELD**	**IMDb**	**SST2**	**Movie Review**	**CR**
LR	**FT**	*0.69 ± 0.01*	*0.66 ± 0.00*	***0.50 ± 0.01***	*0.34 ± 0.01*	0.85 ± 0.00	0.80 ± 0.00	0.76 ± 0.01	0.77 ± 0.01
	TF-IDF	*0.66 ± 0.02*	*0.89 ± 0.00*	***0.61 ± 0.01***	*0.36 ± 0.01*	*0.87 ± 0.01*	0.79 ± 0.02	0.74 ± 0.01	0.76 ± 0.01
SVM	**FT**	*0.68 ± 0.02*	*0.67 ± 0.00*	***0.78 ± 0.01***	*0.35 ± 0*	0.85 ± 0.00	0.81 ± 0.00	0.76 ± 0.01	0.77 ± 0.01
	TF-IDF	*0.65 ± 0.00*	*0.87 ± 0.03*	***0.80 ± 0.01***	*0.37 ± 0.01*	*0.86 ± 0.02*	0.79 ± 0.01	0.73 ± 0.01	0.76 ± 0.02
RF	**FT**	*0.65 ± 0.02*	***0.58 ± 0.01***	***0.84 ± 0.00***	*0.47 ± 0.05*	0.81 ± 0.00	0.77 ± 0.01	0.75 ± 0.01	*0.75 ± 0.02*
	TF-IDF	*0.64 ± 0.02*	*0.88 ± 0.00*	***0.84 ± 0.00***	*0.51 ± 0.01*	0.83 ± 0.01	0.79 ± 0.01	0.74 ± 0.01	***0.75 ± 0.01***
XGB	**FT**	*0.71 ± 0.01*	*0.69 ± 0.01*	*0.83 ± 0.01*	***0.52 ± 0***	0.85 ± 0.00	0.79 ± 0.02	0.76 ± 0.01	0.78 ± 0.01
	TF-IDF	*0.63 ± 0.03*	*0.86 ± 0.13*	*0.84 ± 0.00*	***0.52 ± 0***	*0.87 ± 0.00*	0.78 ± 0.01	0.70 ± 0.01	***0.74 ± 0.01***
BiLSTM		*0.73 ± 0.01*	*0.90 ± 0.06*	**0.85 ± 0.00**	*0.52 ± 0.01*	***0.90 ± 0.00***	***0.84 ± 0.01***	*0.79 ± 0.01*	0.8 ± 0.01
CNN		*0.67 ± 0.01*	*0.87 ± 0.01*	*0.84 ± 0.00*	*0.5 ± 0.01*	***0.89 ± 0.00***	0.81 ± 0.01	***0.76 ± 0.01***	0.76 ± 0.01
RoBERTa		**0.80 ± 0.00**	**0.91 ± 0.01**	*0.83 ± 0.01*	**0.63 ± 0.01**	**0.93 ± 0.00**	**0.89 ± 0.02**	**0.87 ± 0.01**	**0.92 ± 0.01**

		Sentiment Analysis			
		Sarcasm Detection			
model	**vec_rep**	**iSarcasm**	**SemEval A**	**SNH**	**GENsarc**
LR	**FT**	***0.56 ± 0.0***	0.62 ± 0.01	0.77 ± 0.00	0.72 ± 0.01
	TF-IDF	***0.70 ± 0.01***	**0.65 ± 0.01**	0.91 ± 0.04	0.71 ± 0.01
SVM	**FT**	***0.56 ± 0.02***	0.62 ± 0.01	0.76 ± 0.00	0.72 ± 0.01
	TF-IDF	***0.70 ± 0.03***	0.63 ± 0.01	0.90 ± 0.03	0.71 ± 0.02
RF	**FT**	***0.75 ± 0.06***	0.61 ± 0.01	0.95 ± 0.00	0.71 ± 0.01
	TF-IDF	***0.79 ± 0.07***	0.63 ± 0.01	**0.96 ± 0.00**	0.7 ± 0.01
XGB	**FT**	0.83 ± 0.01	0.61 ± 0.01	0.94 ± 0.025	0.72 ± 0.01
	TF-IDF	0.85 ± 0.01	0.63 ± 0.02	0.85 ± 0.02	0.68 ± 0.01
BiLSTM		0.85 ± 0.01	0.63 ± 0.015	0.93 ± 0.01	*0.73 ± 0.01*
CNN		0.86 ± 0.00	0.62 ± 0.031	0.93 ± 0.01	0.72 ± 0.01
RoBERTa		**0.86 ± 0**	0.62 ± 0.00	0.90 ± 0.00	**0.8 ± 0.01**

Table 14: Results accuracy performance. The best performances are underlined and put in bold face. We did Wilcoxon signed-rank test to check pairwise differences between the results. The italic results do not significantly differ from the best result with a confidence interval of 95%. The bold numbers do not differ with a 90% confidence interval.

		Fake News Classification				Topic Modeling			
model	**vec_rep**	**Gossipcop**	**CoAID**	**LIAR**	**McIntire**	**20News**	**AGNews**	**WOS**	**BBC**
LR	**FT**	***0.70 ± 0.01***	0.88 ± 0.02	***0.23 ± 0.01***	0.82 ± 0.01	*0.63 ± 0.00*	*0.89 ± 0.00*	*0.84 ± 0.00*	*0.97 ± 0*
	TF-IDF	***0.76 ± 0.01***	***0.95 ± 0.02***	***0.22 ± 0.00***	0.83 ± 0.01	**0.67 ± 0.00**	*0.91 ± 0.00*	*0.89 ± 0.00*	*0.97 ± 0.01*
SVM	**FT**	***0.70 ± 0.01***	0.88 ± 0.03	***0.23 ± 0.01***	0.82 ± 0.01	*0.61 ± 0.01*	*0.89 ± 0.00*	*0.84 ± 0.03*	*0.96 ± 0.01*
	TF-IDF	***0.75 ± 0.01***	***0.96 ± 0.01***	***0.22 ± 0.02***	0.82 ± 0.02	*0.66 ± 0.00*	*0.91 ± 0.01*	*0.89 ± 0.00*	*0.97 ± 0*
RF	**FT**	0.74 ± 0.05	0.96 ± 0.01	**0.26 ± 0.03**	0.8 ± 0.01	*0.55 ± 0.00*	*0.88 ± 0.00*	*0.80 ± 0.01*	*0.94 ± 0.01*
	TF-IDF	**0.81 ± 0.01**	***0.96 ± 0.01***	***0.26 ± 0.01***	*0.83 ± 0.01*	*0.63 ± 0.00*	*0.91 ± 0.01*	*0.84 ± 0.01*	*0.94 ± 0.01*
XGB	**FT**	0.79 ± 0.00	**0.96 ± 0.01**	***0.24 ± 0.01***	0.81 ± 0.01	*0.57 ± 0.03*	*0.87 ± 0.04*	*0.82 ± 0.03*	*0.94 ± 0*
	TF-IDF	***0.81 ± 0.00***	0.91 ± 0.02	***0.26 ± 0.02***	0.8 ± 0.01	*0.61 ± 0.01*	*0.88 ± 0.06*	*0.91 ± 0.02*	*0.94 ± 0.01*
BiLSTM		***0.79 ± 0.01***	0.92 ± 0.01	***0.25 ± 0.01***	0.82 ± 0.01	*0.62 ± 0.01*	**0.93 ± 0.00**	*0.91 ± 0.01*	*0.94 ± 0.01*
CNN		***0.77 ± 0.01***	0.9 ± 0	***0.22 ± 0.03***	0.81 ± 0.01	*0.57 ± 0.01*	*0.91 ± 0.00*	*0.88 ± 0.01*	*0.94 ± 0.01*
RoBERTa		***0.78 ± 0.02***	*0.94 ± 0.01*	*0.14 ± 0.06*	**0.91 ± 0.01**	*0.596 ± 0.01*	*0.89 ± 0.01*	**0.92 ± 0.01**	**0.97 ± 0**

		Sentiment Analysis							
		Emotion Detection				Polarity Detection			
model	**vec_rep**	**TweetEval**	**CARER**	**Silicone**	**MELD**	**IMDb**	**SST2**	**Movie Review**	**CR**
LR	**FT**	*0.65 ± 0.01*	*0.59 ± 0.00*	***0.23 ± 0.00***	***0.29 ± 0.01***	0.85 ± 0.00	0.80 ± 0.00	0.76 ± 0.01	0.76 ± 0.01
	TF-IDF	*0.61 ± 0.02*	*0.84 ± 0.00*	***0.26 ± 0.00***	***0.26 ± 0***	*0.87 ± 0.01*	0.80 ± 0.02	0.74 ± 0.01	0.74 ± 0.01
SVM	**FT**	*0.63 ± 0.02*	*0.60 ± 0.00*	***0.31 ± 0.02***	***0.29 ± 0.01***	0.85 ± 0.00	0.81 ± 0.00	0.76 ± 0.01	0.76 ± 0.01
	TF-IDF	*0.60 ± 0.00*	*0.81 ± 0.02*	***0.36 ± 0.02***	***0.26 ± 0.01***	*0.86 ± 0.01*	0.79 ± 0.01	0.73 ± 0.01	0.74 ± 0.03
RF	**FT**	*0.73 ± 0.04*	*0.59 ± 0.06*	**0.61 ± 0.01**	*0.35 ± 0.09*	0.81 ± 0.00	0.77 ± 0.01	0.75 ± 0.01	0.75 ± 0.02
	TF-IDF	*0.69 ± 0.02*	**0.88 ± 0.01**	***0.61 ± 0.01***	***0.37 ± 0.02***	0.83 ± 0.01	0.80 ± 0.01	0.74 ± 0.01	0.74 ± 0.02
XGB	**FT**	*0.71 ± 0.01*	*0.67 ± 0.01*	*0.53 ± 0.19*	**0.48 ± 0.06**	0.85 ± 0.00	0.79 ± 0.02	0.76 ± 0.01	0.77 ± 0.01
	TF-IDF	*0.66 ± 0.02*	*0.84 ± 0.01*	***0.57 ± 0.07***	*0.44 ± 0.08*	*0.87 ± 0.00*	0.78 ± 0.00	0.7 ± 0.01	0.72 ± 0.01
BiLSTM		*0.69 ± 0.02*	*0.86 ± 0.06*	***0.60 ± 0.03***	***0.41 ± 0.05***	***0.90 ± 0.00***	***0.84 ± 0.00***	*0.79 ± 0.01*	**0.78 ± 0.02**
CNN		*0.63 ± 0.02*	*0.82 ± 0.01*	***0.39 ± 0.07***	***0.28 ± 0.02***	***0.89 ± 0.00***	0.81 ± 0.01	0.76 ± 0.01	0.75 ± 0.01
RoBERTa		**0.78 ± 0.01**	*0.87 ± 0.01*	*0.20 ± 0.01*	***0.42 ± 0.01***	**0.93 ± 0.00**	**0.89 ± 0.02**	**0.87 ± 0.01**	0.58 ± 0.03

		Sentiment Analysis			
		Sarcasm Detection			
model	**vec_rep**	**iSarcasm**	**SemEval A**	**SNH**	**GENsarc**
LR	**FT**	0.52 ± 0.01	*0.62 ± 0.01*	0.76 ± 0.00	0.72 ± 0.01
	TF-IDF	*0.55 ± 0.01*	0.64 ± 0.01	0.91 ± 0.04	0.71 ± 0.01
SVM	**FT**	0.52 ± 0.01	*0.62 ± 0.01*	0.76 ± 0.00	0.73 ± 0.01
	TF-IDF	0.54 ± 0.02	0.63 ± 0.01	0.90 ± 0.03	0.71 ± 0.01
RF	**FT**	0.49 ± 0.02	***0.60 ± 0.01***	0.95 ± 0.00	0.71 ± 0.01
	TF-IDF	0.51 ± 0.03	0.62 ± 0.01	**0.96 ± 0.00**	0.7 ± 0.01
XGB	**FT**	0.52 ± 0.04	***0.60 ± 0.01***	0.94 ± 0.03	0.72 ± 0.01
	TF-IDF	**0.66 ± 0.06**	*0.62 ± 0.02*	0.85 ± 0.02	0.69 ± 0.01
BiLSTM		0.64 ± 0.05	0.63 ± 0.01	0.93 ± 0.01	*0.73 ± 0.01*
CNN		0.53 ± 0.18	0.63 ± 0.02	0.93 ± 0.01	0.73 ± 0.01
RoBERTa		0.43 ± 0	**0.70 ± 0.04**	0.90 ± 0.00	**0.8 ± 0.01**

Table 15: Results precision performance. The best performances are underlined and put in bold face. We did Wilcoxon signed-rank test to check pairwise differences between the results. The italic results do not significantly differ from the best result with a confidence interval of 95%. The bold numbers do not differ with a 90% confidence interval.

		Fake News Classification				Topic Modeling			
model	**vec_rep**	**Gossipcop**	**CoAID**	**LIAR**	**McIntire**	**20News**	**AGNews**	**WOS**	**BBC**
LR	**FT**	0.75 ± 0.00	0.92 ± 0.02	**0.26 ± 0.02**	0.82 ± 0.01	*0.62 ± 0.00*	*0.89 ± 0.00*	*0.84 ± 0.00*	*0.97 ± 0*
	TF-IDF	**0.79 ± 0.01**	**0.95 ± 0.01**	*0.23 ± 0.00*	0.83 ± 0.01	**0.66 ± 0.00**	*0.91 ± 0.00*	*0.89 ± 0.00*	*0.97 ± 0.01*
SVM	**FT**	0.75 ± 0.01	0.92 ± 0.02	***0.25 ± 0.02***	0.82 ± 0.01	*0.60 ± 0.01*	*0.89 ± 0.00*	*0.84 ± 0.03*	*0.96 ± 0.01*
	TF-IDF	0.78 ± 0.02	***0.95 ± 0.01***	***0.23 ± 0.02***	0.82 ± 0.02	*0.63 ± 0.02*	*0.91 ± 0.01*	*0.89 ± 0.01*	*0.97 ± 0*
RF	**FT**	0.64 ± 0.01	*0.93 ± 0.01*	***0.21 ± 0.01***	0.8 ± 0.01	*0.55 ± 0.00*	*0.88 ± 0.00*	*0.79 ± 0.01*	*0.94 ± 0.01*
	TF-IDF	***0.71 ± 0.01***	***0.93 ± 0.01***	***0.22 ± 0.01***	*0.83 ± 0.01*	*0.62 ± 0.00*	*0.91 ± 0.01*	*0.81 ± 0.01*	*0.93 ± 0.01*
XGB	**FT**	***0.71 ± 0.01***	***0.93 ± 0.02***	***0.22 ± 0.00***	0.81 ± 0.01	*0.57 ± 0.03*	*0.87 ± 0.04*	*0.81 ± 0.04*	*0.94 ± 0*
	TF-IDF	***0.71 ± 0.02***	0.87 ± 0.03	***0.23 ± 0.00***	0.8 ± 0.01	*0.59 ± 0.01*	*0.87 ± 0.11*	*0.91 ± 0.03*	*0.94 ± 0.01*
BiLSTM		***0.75 ± 0.01***	0.90 ± 0.01	***0.24 ± 0.01***	0.82 ± 0.01	*0.61 ± 0.01*	**0.93 ± 0.00**	*0.91 ± 0.01*	*0.94 ± 0.01*
CNN		***0.74 ± 0.01***	0.89 ± 0.01	***0.21 ± 0.01***	0.81 ± 0.01	*0.55 ± 0.01*	*0.91 ± 0.00*	*0.88 ± 0.01*	*0.95 ± 0.01*
RoBERTa		***0.76 ± 0.01***	0.94 ± 0.01	***0.20 ± 0.01***	**0.91 ± 0.01**	*0.61 ± 0.01*	*0.89 ± 0.01*	**0.92 ± 0.01**	**0.97 ± 0**

		Sentiment Analysis							
		Emotion Detection				Polarity Detection			
model	**vec_rep**	**TweetEval**	**CARER**	**Silicone**	**MELD**	**IMDb**	**SST2**	**Movie Review**	**CR**
LR	**FT**	*0.68 ± 0.01*	*0.68 ± 0.00*	*0.46 ± 0.01*	*0.34 ± 0.01*	0.85 ± 0.00	0.80 ± 0.00	0.76 ± 0.01	0.77 ± 0.01
	TF-IDF	*0.63 ± 0.01*	*0.86 ± 0.01*	**0.53 ± 0.00**	*0.3 ± 0.01*	*0.87 ± 0.01*	0.79 ± 0.02	0.74 ± 0.01	0.75 ± 0.01
SVM	**FT**	*0.65 ± 0.03*	*0.67 ± 0.00*	***0.38 ± 0.03***	*0.34 ± 0.01*	0.85 ± 0.00	0.81 ± 0.00	0.76 ± 0.01	0.77 ± 0.01
	TF-IDF	*0.59 ± 0.01*	*0.87 ± 0.03*	***0.45 ± 0.03***	*0.29 ± 0.01*	*0.86 ± 0.01*	0.78 ± 0.01	0.73 ± 0.01	0.75 ± 0.03
RF	**FT**	*0.52 ± 0.02*	*0.35 ± 0.01*	***0.31 ± 0.00***	*0.21 ± 0*	0.81 ± 0.00	0.77 ± 0.01	0.75 ± 0.01	0.7 ± 0.02
	TF-IDF	*0.52 ± 0.02*	***0.78 ± 0.01***	***0.32 ± 0.01***	*0.23 ± 0*	0.83 ± 0.01	0.79 ± 0.00	0.74 ± 0.01	0.69 ± 0.02
XGB	**FT**	*0.62 ± 0.01*	*0.55 ± 0.02*	***0.24 ± 0.07***	***0.23 ± 0.01***	0.85 ± 0.00	0.79 ± 0.02	0.76 ± 0.01	0.75 ± 0.01
	TF-IDF	*0.53 ± 0.03*	*0.80 ± 0.15*	***0.23 ± 0.01***	***0.21 ± 0.01***	*0.87 ± 0.00*	0.78 ± 0.01	0.70 ± 0.01	0.69 ± 0.02
BiLSTM		*0.65 ± 0.02*	*0.85 ± 0.09*	***0.29 ± 0.02***	*0.27 ± 0.01*	***0.90 ± 0.00***	***0.84 ± 0.01***	*0.79 ± 0.01*	**0.79 ± 0.01**
CNN		*0.61 ± 0.03*	*0.81 ± 0.03*	***0.23 ± 0.01***	*0.26 ± 0.01*	***0.89 ± 0.00***	0.81 ± 0.01	0.76 ± 0.01	0.75 ± 0.01
RoBERTa		**0.75 ± 0.01**	**0.87 ± 0.01**	*0.19 ± 0.01*	**0.45 ± 0.01**	**0.93 ± 0.00**	**0.89 ± 0.02**	**0.87 ± 0.01**	0.54 ± 0.03

		Sentiment Analysis			
		Sarcasm Detection			
model	**vec_rep**	**iSarcasm**	**SemEval A**	**SNH**	**GENsarc**
LR	**FT**	0.53 ± 0.02	0.63 ± 0.01	0.77 ± 0.00	0.72 ± 0.01
	TF-IDF	**0.58 ± 0.02**	**0.64 ± 0.01**	0.91 ± 0.04	0.71 ± 0.01
SVM	**FT**	0.55 ± 0.02	0.62 ± 0.01	0.77 ± 0.00	0.73 ± 0.01
	TF-IDF	0.56 ± 0.03	0.64 ± 0.01	0.90 ± 0.03	0.7 ± 0.02
RF	**FT**	0.49 ± 0.015	0.61 ± 0.01	0.95 ± 0.00	0.71 ± 0.01
	TF-IDF	***0.50 ± 0.01***	0.62 ± 0.01	**0.96 ± 0.00**	0.7 ± 0.01
XGB	**FT**	***0.51 ± 0.01***	0.61 ± 0.01	0.94 ± 0.03	0.72 ± 0.01
	TF-IDF	*0.55 ± 0.02*	0.61 ± 0.01	0.84 ± 0.02	0.68 ± 0.01
BiLSTM		***0.54 ± 0.01***	0.64 ± 0.01	0.93 ± 0.01	*0.73 ± 0.01*
CNN		***0.50 ± 0.00***	0.63 ± 0.02	0.93 ± 0.01	0.72 ± 0.01
RoBERTa		***0.5 ± 0***	0.53 ± 0.01	0.90 ± 0.01	**0.8 ± 0.01**

Table 16: Results recall performance. The best performances are underlined and put in bold face. We did Wilcoxon signed-rank test to check pairwise differences between the results. The italic results do not significantly differ from the best result with a confidence interval of 95%. The bold numbers do not differ with a 90% confidence interval.

		Fake News Classification				Topic Modeling			
model	**vec_rep**	**Gossipcop**	**CoAID**	**LIAR**	**McIntire**	**20News**	**AGNews**	**WOS**	**BBC**
LR	**FT**	0.82 ± 0.01	0.96 ± 0.01	-	0.89 ± 0.01	-	-	-	-
	TF-IDF	**0.87 ± 0.01**	***0.99 ± 0.00***	-	***0.9 ± 0.01***	-	-	-	-
SVM	**FT**	0.82 ± 0.01	0.96 ± 0.01	-	0.89 ± 0.01	-	-	-	-
	TF-IDF	0.86 ± 0.01	***0.99 ± 0.00***	-	0.9 ± 0.02	-	-	-	-
RF	**FT**	0.76 ± 0.02	0.99 ± 0.01	-	0.88 ± 0.01	-	-	-	-
	TF-IDF	***0.85 ± 0.01***	**0.99 ± 0.00**	-	**0.91 ± 0.01**	-	-	-	-
XGB	**FT**	***0.84 ± 0.01***	***0.99 ± 0.01***	-	0.9 ± 0.01	-	-	-	-
	TF-IDF	***0.85 ± 0.00***	0.97 ± 0.00	-	0.87 ± 0.01	-	-	-	-
BiLSTM		***0.85 ± 0.01***	0.98 ± 0.00	-	0.9 ± 0.01	-	-	-	-
CNN		***0.84 ± 0.01***	0.98 ± 0.00	-	0.88 ± 0.01	-	-	-	-
RoBERTa		***0.76 ± 0.01***	0.94 ± 0.01	-	*0.91 ± 0.01*	-	-	-	-

		Sentiment Analysis							
		Emotion Detection				Polarity Detection			
model	**vec_rep**	**TweetEval**	**CARER**	**Silicone**	**MELD**	**IMDb**	**SST2**	**Movie Review**	**CR**
LR	**FT**	-	-	-	-	0.93 ± 0.00	0.88 ± 0.00	0.85 ± 0.00	0.85 ± 0.01
	TF-IDF	-	-	-	-	0.94 ± 0.00	0.87 ± 0.02	0.82 ± 0.01	0.85 ± 0.01
SVM	**FT**	-	-	-	-	0.93 ± 0.00	0.88 ± 0.00	0.85 ± 0.01	0.86 ± 0.01
	TF-IDF	-	-	-	-	0.94 ± 0.01	0.85 ± 0.01	0.80 ± 0.02	0.83 ± 0.03
RF	**FT**	-	-	-	-	0.89 ± 0.00	0.86 ± 0.00	0.82 ± 0.01	0.84 ± 0.01
	TF-IDF	-	-	-	-	0.91 ± 0.00	0.88 ± 0.00	0.82 ± 0.01	0.82 ± 0.02
XGB	**FT**	-	-	-	-	0.93 ± 0.00	0.87 ± 0.02	0.84 ± 0.01	0.86 ± 0.01
	TF-IDF	-	-	-	-	0.94 ± 0.00	0.86 ± 0.00	0.77 ± 0.01	0.82 ± 0.01
BiLSTM		-	-	-	-	**0.96 ± 0.00**	**0.92 ± 0.01**	**0.87 ± 0.01**	**0.88 ± 0.01**
CNN		-	-	-	-	0.96 ± 0.00	0.89 ± 0.01	0.84 ± 0.01	0.84 ± 0.01
RoBERTa		-	-	-	-	***0.93 ± 0.00***	***0.89 ± 0.02***	*0.87 ± 0.01*	0.06 ± 0.03

		Sentiment Analysis			
		Sarcasm Detection			
model	**vec_rep**	**iSarcasm**	**SemEval A**	**SNH**	**GENsarc**
LR	**FT**	0.55 ± 0.03	0.68 ± 0.01	0.85 ± 0.00	0.8 ± 0.01
	TF-IDF	*0.61 ± 0.02*	0.70 ± 0.01	0.96 ± 0.02	0.79 ± 0.01
SVM	**FT**	0.57 ± 0.01	0.68 ± 0.01	0.84 ± 0.00	0.8 ± 0.01
	TF-IDF	0.59 ± 0.02	***0.68 ± 0.02***	0.96 ± 0.02	0.77 ± 0.02
RF	**FT**	0.47 ± 0.02	0.66 ± 0.01	0.99 ± 0.00	0.78 ± 0.01
	TF-IDF	0.54 ± 0.04	0.69 ± 0.01	**0.99 ± 0.00**	0.77 ± 0.01
XGB	**FT**	0.55 ± 0.06	0.66 ± 0.01	0.98 ± 0.01	0.8 ± 0.01
	TF-IDF	**0.61 ± 0.023**	0.66 ± 0.01	0.93 ± 0.01	0.76 ± 0.01
BiLSTM		0.59 ± 0.03	**0.70 ± 0.01**	0.98 ± 0.00	**0.81 ± 0.01**
CNN		0.49 ± 0.03	0.69 ± 0.01	0.98 ± 0.00	0.8 ± 0.02
RoBERTa		0.5 ± 0	0.53 ± 0.01	0.90 ± 0.01	*0.8 ± 0.01*

Table 17: Results AUC performance part. The best performances are underlined and put in bold face. We did Wilcoxon signed-rank test to check pairwise differences between the results. The italic results do not significantly differ from the best result with a confidence interval of 95%. The bold numbers do not differ with a 90% confidence interval.

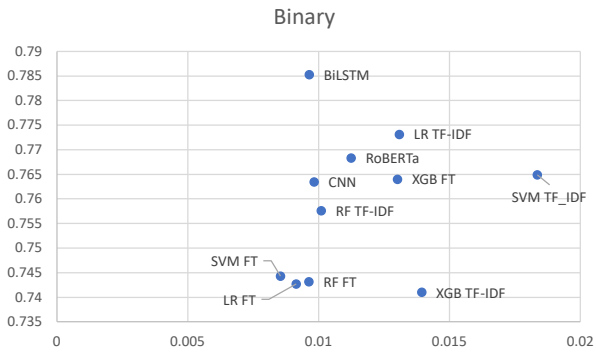
		Fake News Classification				Topic Modeling			
model	**vec_rep**	**Gossipcop**	**CoAID**	**LIAR**	**McIntire**	**20News**	**AGNews**	**WOS**	**BBC**
LR	**FT**	0.92 ± 0.00	0.98 ± 0.01	-	0.89 ± 0.02	-	-	-	-
	TF-IDF	**0.95 ± 0.00**	***1.00 ± 0.00***	-	***0.9 ± 0.01***	-	-	-	-
SVM	**FT**	0.92 ± 0.01	0.98 ± 0.01	-	0.89 ± 0.01	-	-	-	-
	TF-IDF	0.94 ± 0.01	***1.00 ± 0.00***	-	0.9 ± 0.02	-	-	-	-
RF	**FT**	0.89 ± 0.01	1.00 ± 0.00	-	0.87 ± 0.02	-	-	-	-
	TF-IDF	***0.93 ± 0.00***	**1.00 ± 0.00**	-	**0.91 ± 0.01**	-	-	-	-
XGB	**FT**	***0.93 ± 0.01***	***1.00 ± 0.00***	-	0.9 ± 0.01	-	-	-	-
	TF-IDF	***0.94 ± 0.00***	0.99 ± 0.00	-	0.87 ± 0.02	-	-	-	-
BiLSTM		***0.94 ± 0.00***	0.99 ± 0.00	-	0.9 ± 0.01	-	-	-	-
CNN		***0.93 ± 0.01***	0.99 ± 0.00	-	0.89 ± 0	-	-	-	-
RoBERTa		***0.87 ± 0.01***	0.98 ± 0.00	-	*0.88 ± 0.02*	-	-	-	-

		Sentiment Analysis							
		Emotion Detection				Polarity Detection			
model	**vec_rep**	**TweetEval**	**CARER**	**Silicone**	**MELD**	**IMDb**	**SST2**	**Movie Review**	**CR**
LR	**FT**	-	-	-	-	0.93 ± 0.00	0.88 ± 0.00	0.85 ± 0.01	0.91 ± 0.01
	TF-IDF	-	-	-	-	0.940 ± 0.01	0.86 ± 0.02	0.83 ± 0.01	0.91 ± 0.01
SVM	**FT**	-	-	-	-	0.93 ± 0.00	0.89 ± 0.00	0.85 ± 0.01	0.91 ± 0.01
	TF-IDF	-	-	-	-	0.93 ± 0.01	0.85 ± 0.02	0.80 ± 0.02	0.9 ± 0.02
RF	**FT**	-	-	-	-	0.89 ± 0.00	0.86 ± 0.00	0.82 ± 0.01	0.9 ± 0.01
	TF-IDF	-	-	-	-	0.90 ± 0.01	0.88 ± 0.01	0.82 ± 0.01	0.88 ± 0.02
XGB	**FT**	-	-	-	-	0.93 ± 0.00	0.88 ± 0.02	0.85 ± 0.01	0.92 ± 0.01
	TF-IDF	-	-	-	-	0.94 ± 0.00	0.85 ± 0.01	0.77 ± 0.01	0.9 ± 0
BiLSTM		-	-	-	-	**0.96 ± 0.00**	**0.93 ± 0.01**	**0.88 ± 0.01**	**0.93 ± 0.01**
CNN		-	-	-	-	0.96 ± 0.00	0.89 ± 0.01	0.85 ± 0.01	0.91 ± 0.01
RoBERTa		-	-	-	-	***0.90 ± 0.00***	***0.85 ± 0.03***	*0.83 ± 0.012*	0.65 ± 0.01

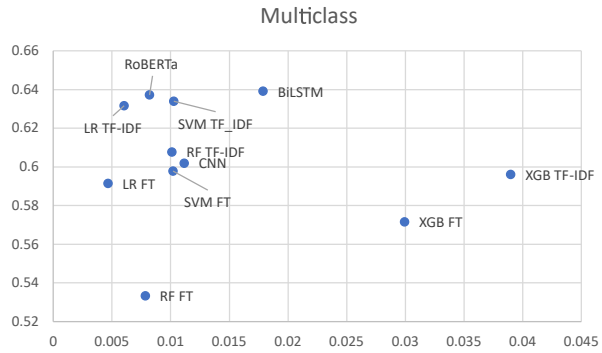
		Sentiment Analysis			
		Sarcasm Detection			
model	**vec_rep**	**iSarcasm**	**SemEval A**	**SNH**	**GENsarc**
LR	**FT**	0.15 ± 0.01	0.60 ± 0.01	0.81 ± 0.00	0.79 ± 0.01
	TF-IDF	*0.22 ± 0.02*	**0.61 ± 0.01**	0.95 ± 0.02	0.78 ± 0.01
SVM	**FT**	0.16 ± 0.01	0.60 ± 0.01	0.80 ± 0.00	0.8 ± 0.01
	TF-IDF	0.20 ± 0.02	0.58 ± 0.02	0.94 ± 0.03	0.77 ± 0.03
RF	**FT**	0.14 ± 0.00	0.55 ± 0.01	0.99 ± 0.00	0.77 ± 0.02
	TF-IDF	0.16 ± 0.02	0.60 ± 0.01	**0.99 ± 0.00**	0.77 ± 0.01
XGB	**FT**	0.17 ± 0.02	0.56 ± 0.02	0.98 ± 0.01	0.79 ± 0.01
	TF-IDF	**0.27 ± 0.04**	0.58 ± 0.02	0.92 ± 0.02	0.75 ± 0.01
BiLSTM		0.25 ± 0.03	0.61 ± 0.01	0.97 ± 0.00	**0.8 ± 0.02**
CNN		0.15 ± 0.01	0.59 ± 0.01	0.97 ± 0.00	0.78 ± 0.02
RoBERTa		0.14 ± 0	0.42 ± 0.01	0.84 ± 0.01	*0.75 ± 0.02*

Table 18: Results AUCPR performance. The best performances are underlined and put in bold face. We did Wilcoxon signed-rank test to check pairwise differences between the results. The italic results do not significantly differ from the best result with a confidence interval of 95%. The bold numbers do not differ with a 90% confidence interval.

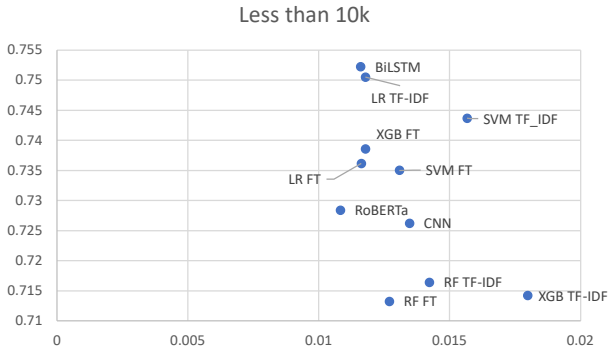
6.4. F1-standard deviation trade-off Dataset Specifications



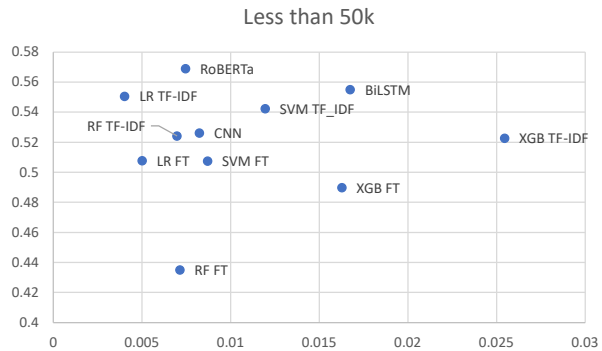
(a) Performance-Variance trade-off binary datasets



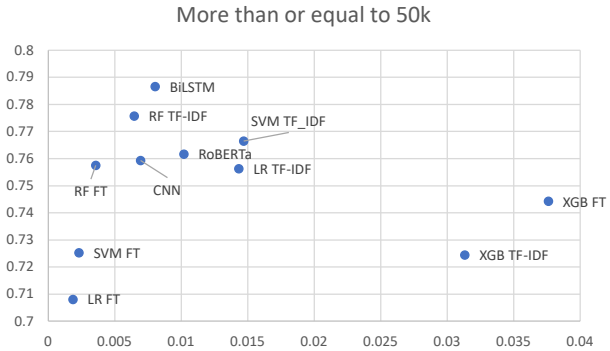
(b) Performance-Variance trade-off multi-class datasets



(c) Performance-Variance trade-off datasets smaller than 10,000



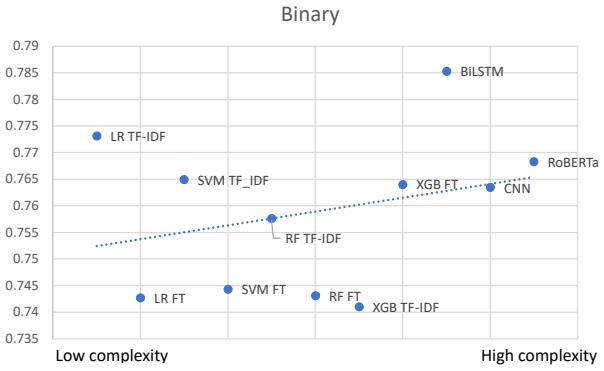
(d) Performance-Variance trade-off datasets smaller than 50,000



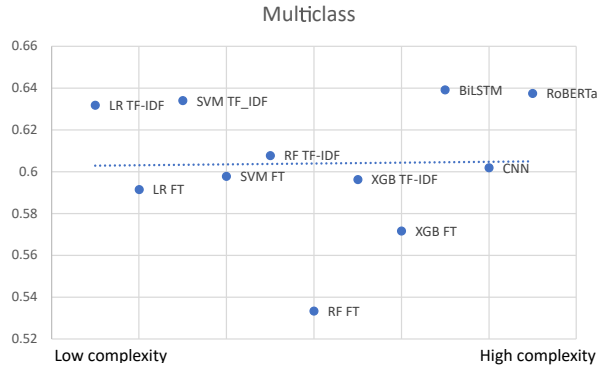
(e) Performance-Variance trade-off datasets larger than or equal to 50,000

Figure 5: Overview Performance-Variance trade-off per dataset specification.

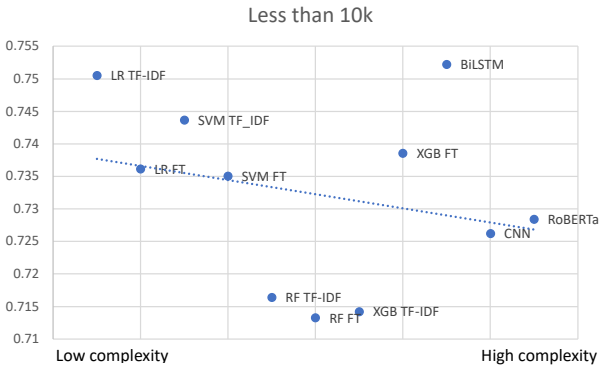
6.5. *Performance-Complexity trade-off Dataset Specifications*



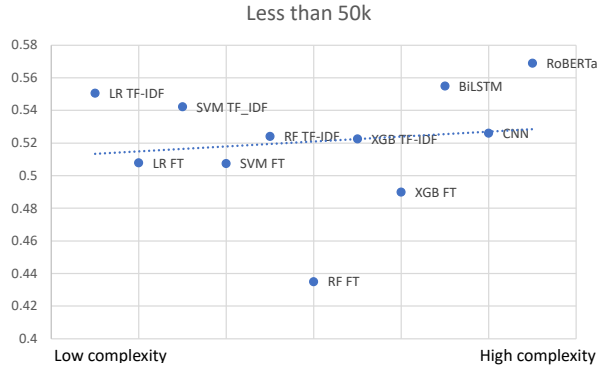
(a) Performance-Complexity trade-off binary datasets



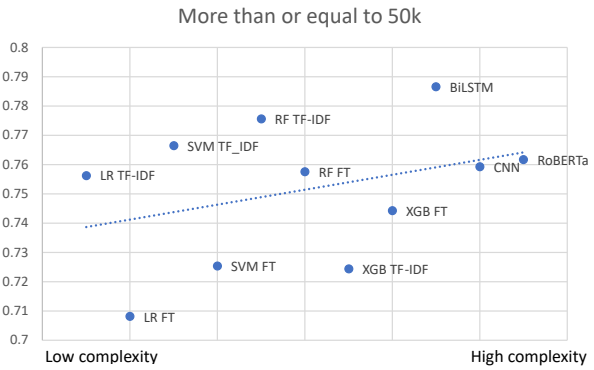
(b) Performance-Complexity trade-off multi-class datasets



(c) Performance-Complexity trade-off datasets smaller than 10,000



(d) Performance-Complexity trade-off datasets smaller than 50,000



(e) Performance-Complexity trade-off datasets larger than or equal to 50,000

Figure 6: Overview Performance-Complexity trade-off per dataset specification.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X., 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. URL: https://www.tensorflow.org/. software available from tensorflow.org.

Abu Farha, I., Oprea, S.V., Wilson, S., Magdy, W., 2022. SemEval-2022 task 6: iSarcasmEval, intended sarcasm detection in English and Arabic, in: Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), Association for Computational Linguistics, Seattle, United States. pp. 802–814. URL: https://aclanthology.org/2022.semeval-1.111.

Agrawal, C., Pandey, A., Goyal, S., 2022. Fake news detection system based on modified bi-directional long short term memory. Multimedia Tools and Applications , 1–25.

Aka Uymaz, H., Kumova Metin, S., 2022. Vector based sentiment and emotion analysis from text: A survey. Engineering Applications of Artificial Intelligence 113, 104922. URL: https://www.sciencedirect.com/science/article/pii/S0952197622001452, doi:https://doi.org/10.1016/j.engappai.2022.104922.

Alaparthi, S., Mishra, M., 2021. Bert: A sentiment analysis odyssey. Journal of Marketing Analytics 9, 118–126.

Aldunate, Á., Maldonado, S., Vairetti, C., Armelini, G., 2022. Understanding customer satisfaction via deep learning and natural language processing. Expert Systems with Applications 209, 118309.

Alswaidan, N., Menai, M.E.B., 2020. A survey of state-of-the-art approaches for emotion recognition in text. Knowledge and Information Systems 62, 2937–2987.

Armendariz, C.S., Purver, M., Pollak, S., Ljubešić, N., Ulčar, M., Vulić, I., Pilehvar, M.T., 2020. SemEval-2020 task 3: Graded word similarity in context, in: Proceedings of the Fourteenth Workshop on Semantic Evaluation, International Committee for Computational Linguistics, Barcelona (online). pp. 36–49. URL: https://aclanthology.org/2020.semeval-1.3, doi:10.18653/v1/2020.semeval-1.3.

Arrieta, A.B., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., García, S., Gil-López, S., Molina, D., Benjamins, R., et al., 2020. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. Information fusion 58, 82–115.

Arslan, Y., Allix, K., Veiber, L., Lothritz, C., Bissyandé, T.F., Klein, J., Goujon, A., 2021. A comparison of pre-trained language models for multi-class text classification in the financial domain, in: Companion Proceedings of the Web Conference 2021, pp. 260–268.

Bannour, N., Ghannay, S., Névéol, A., Ligozat, A.L., 2021. Evaluating the carbon footprint of nlp methods: a survey and analysis of existing tools, in: Proceedings of the Second Workshop on Simple and Efficient Natural Language Processing, pp. 11–21.

Barbieri, F., Camacho-Collados, J., Espinosa Anke, L., Neves, L., 2020. TweetEval: Unified benchmark and comparative evaluation for tweet classification, in: Findings of the Association for Computational Linguistics: EMNLP 2020, Association for Computational Linguistics, Online. pp. 1644–1650. URL: https://aclanthology.org/2020.findings-emnlp.148, doi:10.18653/v1/2020.findings-emnlp.148.

Bergstra, J., Bengio, Y., 2012. Random search for hyper-parameter optimization. Journal of machine learning research 13.

Biewald, L., 2020. Experiment tracking with weights and biases. URL: https://www.wandb.com/. software available from wandb.com.

Boom, C.D., Reusens, M., 2023. Changing data sources in the age of machine learning for official statistics. arXiv:2306.04338.

Capuano, N., Fenza, G., Loia, V., Nota, F.D., 2023. Content-based fake news detection with machine and deep learning: a systematic review. Neurocomputing 530, 91–103. URL: https://www.sciencedirect.com/science/article/pii/S0925231223001376, doi:https://doi.org/10.1016/j.neucom.2023.02.005.

Chandra, R., Krishna, A., 2021. Covid-19 sentiment analysis via deep learning during the rise of novel cases. PloS one 16, e0255615.

Chapuis, E., Colombo, P., Manica, M., Labeau, M., Clavel, C., 2020. Hierarchical pretraining for sequence labelling in spoken dialog, in: Findings of the Association for Computational Linguistics: EMNLP 2020, Association for Computational Linguistics, Online. pp. 2636–2648. URL: https://www.aclweb.org/anthology/2020.findings-emnlp.239, doi:10.18653/v1/2020.findings-emnlp.239.

Charalampakis, B., Spathis, D., Kouslis, E., Kermanidis, K., 2016. A comparison between semi-supervised and supervised text mining techniques on detecting irony in greek political tweets. Engineering Applications of Artificial Intelligence 51, 50–57. URL: https://www.sciencedirect.com/science/article/pii/S0952197616000117, doi:https://doi.org/10.1016/j.engappai.2016.01.007. mining the Humanities: Technologies and Applications.

Chen, L., Jiang, L., Li, C., 2021. Using modified term frequency to improve term weighting for text classification. Engineering Applications of Artificial Intelligence 101, 104215. doi:https://doi.org/10.1016/j.engappai.2021.104215.

Chen, T., Guestrin, C., 2016. Xgboost: A scalable tree boosting system, in: Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, pp. 785–794.

Chhabra, A., Vishwakarma, D.K., 2023. A literature survey on multimodal and multilingual automatic hate speech identification. Multimedia Systems , 1–28.

Choudhary, M., Chouhan, S.S., Pilli, E.S., Vipparthi, S.K., 2021. Berconvonet: A deep learning framework for fake news classification. Applied Soft Computing 110, 107614.

Comito, C., Caroprese, L., Zumpano, E., 2023. Multimodal fake news detection on social media: a survey of deep learning techniques. Social Network Analysis and Mining 13, 101.

da Costa, L.S., Oliveira, I.L., Fileto, R., 2023. Text classification using embeddings: a survey. Knowledge and Information Systems 65, 2761–2803.

Cui, L., Lee, D., 2020. Coaid: Covid-19 healthcare misinformation dataset. arXiv:2006.00885.

De Caigny, A., Coussement, K., De Bock, K.W., Lessmann, S., 2020. Incorporating textual information in customer churn prediction models based on a convolutional neural network. International Journal of Forecasting 36, 1563–1578. URL: https://www.sciencedirect.com/science/article/pii/S0169207019301499, doi:https://doi.org/10.1016/j.ijforecast.2019.03.029.

Demšar, J., 2006. Statistical comparisons of classifiers over multiple data sets. The Journal of Machine learning research 7, 1–30.

Devlin, J., Chang, M.W., Lee, K., Toutanova, K., 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 .

Ding, X., Liu, B., Yu, P.S., 2008. A holistic lexicon-based approach to opinion mining, in: Proceedings of the 2008 international conference on web search and data mining, pp. 231–240.

Escalante, H.J., Montes y Gomez, M., Villasenor, L., Errecalde, M.L., 2016. Early text classification: a naïve solution, in: Proceedings of the 7th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, Association for Computational Linguistics, San Diego, California. pp. 91–99. URL: https://aclanthology.org/W16-0416, doi:10.18653/v1/W16-0416.

Falkner, S., Klein, A., Hutter, F., 2017. Combining hyperband and bayesian optimization, in: NIPS 2017 Bayesian Optimization Workshop (Dec 2017).

Fortuna, P., Nunes, S., 2018. A survey on automatic detection of hate speech in text. ACM Computing Surveys (CSUR) 51, 1–30.

Galke, L., Diera, A., Lin, B.X., Khera, B., Meuser, T., Singhal, T., Karl, F., Scherp, A., 2023. Are we really making much progress in text classification? a comparative review. arXiv:2204.03954.

Galli, A., Masciari, E., Moscato, V., Sperli, G., 2022. A comprehensive benchmark for fake news detection. Journal of Intelligent Information Systems 59, 237–261.

Ghosh, A., 2022. Sentiment analysis of imdb movie reviews: A comparative study on performance of hyperparameter-tuned classification algorithms, in: 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS), IEEE. pp. 289–294.

Gravanis, G., Vakali, A., Diamantaras, K., Karadais, P., 2019. Behind the cues: A benchmarking study for fake news detection. Expert Systems with Applications 128, 201–213.

Greene, D., Cunningham, P., 2006. Practical solutions to the problem of diagonal dominance in kernel document clustering, in: Proc. 23rd International Conference on Machine learning (ICML'06), ACM Press. pp. 377–384.

Gutiérrez-Batista, K., Campaña, J.R., Vila, M.A., Martin-Bautista, M.J., 2019. Using word embeddings and deep learning for supervised topic detection in social networks, in: Flexible Query Answering Systems: 13th International Conference, FQAS 2019, Amantea, Italy, July 2–5, 2019, Proceedings 13, Springer. pp. 155–165.

Hasan, M., Rundensteiner, E., Agu, E., 2021. Deepemotex: Classifying emotion in text messages using deep transfer learning, in: 2021 IEEE International Conference on Big Data (Big Data), IEEE. pp. 5143–5152.

He, G., Gao, Z., Jiang, Z., Kang, Y., Sun, C., Liu, X., Lu, W., 2020. Think beyond the word: Understanding the implied textual meaning by digesting context, local, and noise, in: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 2297–2306.

Herbold, S., 2020. Autorank: A python package for automated ranking of classifiers. Journal of Open Source Software 5, 2173. URL: https://doi.org/10.21105/joss.02173, doi:10.21105/joss.02173.

Hershcovich, D., Webersinke, N., Kraus, M., Bingler, J.A., Leippold, M., 2022. Towards climate awareness in nlp research. arXiv preprint arXiv:2205.05071 .

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural computation 9, 1735–1780.

Huang, F., Li, X., Yuan, C., Zhang, S., Zhang, J., Qiao, S., 2021. Attention-emotion-enhanced convolutional lstm for sentiment analysis. IEEE transactions on neural networks and learning systems .

Ilie, V.I., Truică, C.O., Apostol, E.S., Paschke, A., 2021a. Context-aware misinformation detection: A benchmark of deep learning architectures using word embeddings. IEEE Access 9, 162122–162146.

Ilie, V.I., Truică, C.O., Apostol, E.S., Paschke, A., 2021b. Context-aware misinformation detection: A benchmark of deep learning architectures using word embeddings. IEEE Access 9, 162122–162146. doi:10.1109/ACCESS.2021.3132502.

Jin, Q., Xue, X., Peng, W., Cai, W., Zhang, Y., Zhang, L., 2020. Tblc-rattention: A deep neural network model for recognizing the emotional tendency of chinese medical comment. IEEE Access 8, 96811–96828.

Jindal, S., Sood, R., Singh, R., Vatsa, M., Chakraborty, T., 2020. Newsbag: A multimodal benchmark dataset for fake news detection, in: CEUR Workshop Proc., pp. 138–145.

Joshi, A., Bhattacharyya, P., Carman, M.J., 2017. Automatic sarcasm detection: A survey. ACM Computing Surveys (CSUR) 50, 1–22.

Kaliyar, R.K., Goswami, A., Narang, P., 2021. Fakebert: Fake news detection in social media with a bert-based deep learning approach. Multimedia tools and applications 80, 11765–11788.

Kaliyar, R.K., Goswami, A., Narang, P., Sinha, S., 2020. Fndnet–a deep convolutional neural network for fake news detection. Cognitive Systems Research 61, 32–44.

Kang, M., Ahn, J., Lee, K., 2018. Opinion mining using ensemble text hidden markov models for text classification. Expert Systems with Applications 94, 218–227. URL: https://www.sciencedirect.com/science/article/pii/S0957417417304979, doi:https://doi.org/10.1016/j.eswa.2017.07.019.

Kayalvizhi, S., Thenmozhi, D., Kumar, B.S., Aravindan, C., 2019. Ssn_nlp@ idat-fire-2019: Irony detection in arabic tweets using deep learning and features-based approaches.

Khan, J.Y., Khondaker, M.T.I., Afroz, S., Uddin, G., Iqbal, A., 2021. A benchmark study of machine learning models for online fake news detection. Machine Learning with Applications 4, 100032.

Khatri, A., Pranav, P., 2020. Sarcasm detection in tweets with bert and glove embeddings, in: Proceedings of the Second Workshop on Figurative Language Processing, pp. 56–60.

Kim, B., Park, J., Suh, J., 2020a. Transparency and accountability in ai decision support: Explaining and visualizing convolutional neural networks for text information. Decision Support Systems 134, 113302.

Kim, J., Jang, S., Park, E., Choi, S., 2020b. Text classification using capsules. Neurocomputing 376, 214–221.

Kim, Y., 2014. Convolutional neural networks for sentence classification, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Doha, Qatar. pp. 1746–1751. URL: https://aclanthology.org/D14-1181, doi:10.3115/v1/D14-1181.

Kowsari, K., Brown, D.E., Heidarysafa, M., Meimandi, K.J., Gerber, M.S., Barnes, L.E., 2017. Hdltex: Hierarchical deep learning for text classification, in: 2017 16th IEEE international conference on machine learning and applications (ICMLA), IEEE. pp. 364–371.

Kratzwald, B., Ilić, S., Kraus, M., Feuerriegel, S., Prendinger, H., 2018. Deep learning for affective computing: Text-based emotion recognition in decision support. Decision Support Systems 115, 24–35.

Kraus, M., Feuerriegel, S., Oztekin, A., 2020. Deep learning in business analytics and operations research: Models, applications and managerial implications. European Journal of Operational Research 281, 628–641. URL: https://www.sciencedirect.com/science/article/pii/S0377221719307581, doi:https://doi.org/10.1016/j.ejor.2019.09.018. featured Cluster: Business Analytics: Defining the field and identifying a research agenda.

Lai, V., Cai, J.Z., Tan, C., 2019. Many faces of feature importance: Comparing built-in and post-hoc feature importance in text classification. arXiv preprint arXiv:1910.08534 .

Lê, N.C., Lam, N.T., Nguyen, S.H., Nguyen, D.T., 2020. On vietnamese sentiment analysis: A transfer learning method, in: 2020 RIVF International Conference on Computing and Communication Technologies (RIVF), IEEE. pp. 1–5.

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. nature 521, 436–444.

Lei, Z., Yang, Y., Yang, M., 2018. Saan: A sentiment-aware attention network for sentiment analysis, in: The 41st international ACM SIGIR conference on research & development in information retrieval, pp. 1197–1200.

Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A., Talwalkar, A., 2017. Hyperband: A novel bandit-based approach to hyperparameter optimization. The Journal of Machine Learning Research 18, 6765–6816.

Li, Q., Hu, Q., Lu, Y., Yang, Y., Cheng, J., 2020a. Multi-level word features based on cnn for fake news detection in cultural communication. Personal and Ubiquitous Computing 24, 259–272.

Li, Q., Li, P., Mao, K., Lo, E.Y.M., 2020b. Improving convolutional neural network for text classification by recursive data pruning. Neurocomputing 414, 143–152.

Li, Z., Liu, F., Yang, W., Peng, S., Zhou, J., 2021. A survey of convolutional neural networks: analysis, applications, and prospects. IEEE transactions on neural networks and learning systems .

Li, Z., Yang, Z., Luo, L., Xiang, Y., Lin, H., 2020c. Exploiting adversarial transfer learning for adverse drug reaction detection from texts. Journal of biomedical informatics 106, 103431.

Liu, C., Mengchao, Z., Zhibing, F., Hou, P., Li, Y., 2021. Flitext: A faster and lighter semi-supervised text classification with convolution networks, in: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp. 2481–2491.

Liu, W., Xiao, J., Hong, M., 2020a. Comparison on feature selection methods for text classification, in: Proceedings of the 2020 4th international conference on management engineering, software engineering and service sciences, pp. 82–86.

Liu, Y., Ju, S., Wang, J., Su, C., 2020b. A new feature selection method for text classification based on independent feature space search. Mathematical Problems in Engineering 2020.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V., 2019. Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 .

Maas, A.L., Daly, R.E., Pham, P.T., Huang, D., Ng, A.Y., Potts, C., 2011. Learning word vectors for sentiment analysis, in: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Portland, Oregon, USA. pp. 142–150. URL: `http://www.aclweb.org/anthology/P11-1015`.

Majeed, A., Beg, M.O., Arshad, U., Mujtaba, H., 2022. Deep-emoru: Mining emotions from roman urdu text using deep learning ensemble. Multimedia Tools and Applications 81, 43163–43188.

Mandal, R., Chen, J., Becken, S., Stantic, B., 2021. Empirical study of tweets topic classification using transformer-based language models, in: Intelligent Information and Database Systems: 13th Asian Conference, ACIIDS 2021, Phuket, Thailand, April 7–10, 2021, Proceedings 13, Springer. pp. 340–350.

Mehta, D., Dwivedi, A., Patra, A., Anand Kumar, M., 2021. A transformer-based architecture for fake news classification. Social network analysis and mining 11, 1–12.

Mieskes, M., Fort, K., Névéol, A., Grouin, C., Cohen, K., 2019. Community perspective on replicability in natural language processing, in: Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019), INCOMA Ltd., Varna, Bulgaria. pp. 768–775. URL: https://aclanthology.org/R19-1089, doi:10.26615/978-954-452-056-4_089.

Mikolov, T., Chen, K., Corrado, G., Dean, J., 2013a. Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 .

Mikolov, T., Grave, E., Bojanowski, P., Puhrsch, C., Joulin, A., 2018. Advances in pre-training distributed word representations, in: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), European Language Resources Association (ELRA), Miyazaki, Japan. URL: https://aclanthology.org/L18-1008.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J., 2013b. Distributed representations of words and phrases and their compositionality. Advances in neural information processing systems 26.

Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., Gao, J., 2021. Deep learning–based text classification: a comprehensive review. ACM computing surveys (CSUR) 54, 1–40.

Misra, R., Arora, P., 2019. Sarcasm detection using hybrid neural network. arXiv preprint arXiv:1908.07414 .

Misra, R., Grover, J., 2021. Sculpting Data for ML: The first act of Machine Learning.

Mohammed, A., Kora, R., 2022. An effective ensemble deep learning framework for text classification. Journal of King Saud University-Computer and Information Sciences 34, 8825–8837.

Moreo, A., Esuli, A., Sebastiani, F., 2021. Word-class embeddings for multiclass text classification. Data Mining and Knowledge Discovery 35, 911–963.

Naseem, U., Dunn, A.G., Khushi, M., Kim, J., 2022. Benchmarking for biomedical natural language processing tasks with a domain specific albert. BMC bioinformatics 23, 1–15.

Nemenyi, P.B., 1963. Distribution-free multiple comparisons. Princeton University.

Oraby, S., Harrison, V., Reed, L., Hernandez, E., Riloff, E., Walker, M., 2016. Creating and characterizing a diverse corpus of sarcasm in dialogue, in: Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue, Association for Computational Linguistics, Los Angeles. pp. 31–41. URL: https://aclanthology.org/W16-3604, doi:10.18653/v1/W16-3604.

Otter, D.W., Medina, J.R., Kalita, J.K., 2020. A survey of the usages of deep learning for natural language processing. IEEE transactions on neural networks and learning systems 32, 604–624.

Palomino, D., Ochoa-Luna, J., 2020. Spanish sentiment analysis using universal language model fine-tuning: A detailed case of study, in: Information Management and Big Data: 6th International Conference, SIMBig 2019, Lima, Peru, August 21–23, 2019, Proceedings 6, Springer. pp. 207–217.

Pang, B., Lee, L., 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales, in: Proceedings of ACL, pp. 115–124.

Pang, B., Lee, L., Vaithyanathan, S., 2002. Thumbs up? sentiment classification using machine learning techniques, in: Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10, pp. 79–86.

Parida, U., Nayak, M., Nayak, A.K., 2021. News text categorization using random forest and naive bayes, in: 2021 1st Odisha International Conference on Electrical Power Engineering, Communication and Computing Technology (ODICON), IEEE. pp. 1–4.

Pattanayak, R.M., Behera, H.S., Panigrahi, S., 2021. A novel probabilistic intuitionistic fuzzy set based model for high order fuzzy time series forecasting. Engineering Applications of Artificial Intelligence 99, 104136.

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

Pennington, J., Socher, R., Manning, C.D., 2014. Glove: Global vectors for word representation, in: Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543. URL: http://www.aclweb.org/anthology/D14-1162.

Poria, S., Hazarika, D., Majumder, N., Naik, G., Cambria, E., Mihalcea, R., 2018. Meld: A multimodal multi-party dataset for emotion recognition in conversations. arXiv preprint arXiv:1810.02508 .

Qureshi, M.A., Asif, M., Hassan, M.F., Abid, A., Kamal, A., Safdar, S., Akber, R., 2022. Sentiment analysis of reviews in natural language: Roman urdu as a case study. IEEE Access 10, 24945–24954.

Rahman, R., 2020. A benchmark study on machine learning methods using several feature extraction techniques for news genre detection from bangla news articles & titles, in: Proceedings of the 7th International Conference on Networking, Systems and Security, pp. 25–35.

Razali, M.S., Halin, A.A., Ye, L., Doraisamy, S., Norowi, N.M., 2021. Sarcasm detection using deep learning with contextual features. IEEE Access 9, 68609–68618. doi:10.1109/ACCESS.2021.3076789.

Reusens, M., Reusens, M., Callens, M., vanden Broucke, S., Baesens, B., 2022. Comparison of different modeling techniques for flemish twitter sentiment analysis. Analytics 1, 117–134.

Riduan, G.M., Soesanti, I., Adji, T.B., 2021. A systematic literature review of text classification: Datasets and methods, in: 2021 IEEE 5th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE), IEEE. pp. 71–77.

Sachan, D.S., Zaheer, M., Salakhutdinov, R., 2019. Revisiting lstm networks for semi-supervised text classification via mixed objective function, in: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 6940–6948.

Saravia, E., Liu, H.C.T., Huang, Y.H., Wu, J., Chen, Y.S., 2018. CARER: Contextualized affect representations for emotion recognition, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Brussels, Belgium. pp. 3687–3697. URL: https://www.aclweb.org/anthology/D18-1404, doi:10.18653/v1/D18-1404.

Shahriari, B., Swersky, K., Wang, Z., Adams, R.P., De Freitas, N., 2015. Taking the human out of the loop: A review of bayesian optimization. Proceedings of the IEEE 104, 148–175.

Sharma, D.K., Garg, S., 2021. Ifnd: a benchmark dataset for fake news detection. Complex & Intelligent Systems , 1–21.

Shu, K., Mahudeswaran, D., Wang, S., Lee, D., Liu, H., 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 .

Snoek, J., Larochelle, H., Adams, R.P., 2012. Practical bayesian optimization of machine learning algorithms. Advances in neural information processing systems 25.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C.D., Ng, A., Potts, C., 2013. Recursive deep models for semantic compositionality over a sentiment treebank, in: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Seattle, Washington, USA. pp. 1631–1642. URL: https://www.aclweb.org/anthology/D13-1170.

Soleymani, M., Garcia, D., Jou, B., Schuller, B., Chang, S.F., Pantic, M., 2017. A survey of multimodal sentiment analysis. Image and Vision Computing 65, 3–14. URL: https://www.sciencedirect.com/science/article/pii/S0262885617301191, doi:https://doi.org/10.1016/j.imavis.2017.08.003. multimodal Sentiment Analysis and Mining in the Wild Image and Vision Computing.

Sun, X., Li, X., Li, J., Wu, F., Guo, S., Zhang, T., Wang, G., 2023. Text classification via large language models. arXiv preprint arXiv:2305.08377 .

Sutoyo, E., Rifai, A.P., Risnumawan, A., Saputra, M., 2022. A comparison of text weighting schemes on sentiment analysis of government policies: a case study of replacement of national examinations. Multimedia Tools and Applications 81, 6413–6431.

Tan, Z., Chen, J., Kang, Q., Zhou, M., Abusorrah, A., Sedraoui, K., 2021. Dynamic embedding projection-gated convolutional neural networks for text classification. IEEE Transactions on Neural Networks and Learning Systems 33, 973–982.

Thangaraj, M., Sivakami, M., 2018. Text classification techniques: A literature review. Interdisciplinary journal of information, knowledge, and management 13, 117.

Ulmer, D., Bassignana, E., Müller-Eberstein, M., Varab, D., Zhang, M., van der Goot, R., Hardmeier, C., Plank, B., 2022. Experimental standards for deep learning in natural language processing research, in: Findings of the Association for Computational Linguistics: EMNLP 2022, pp. 2673–2692.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. Advances in neural information processing systems 30.

Vernikou, S., Lyras, A., Kanavos, A., 2022. Multiclass sentiment analysis on covid-19-related tweets using deep learning models. Neural Computing and Applications , 1–13.

Wahba, Y., Madhavji, N., Steinbacher, J., 2023. Attention is not always what you need: Towards efficient classification of domain-specific text. arXiv preprint arXiv:2303.17786 .

Wang, C., Fan, X., 2020. Adaptive convolution kernel for text classification via multi-channel representations, in: Artificial Neural Networks and Machine Learning–ICANN 2020: 29th International Conference on Artificial Neural Networks, Bratislava, Slovakia, September 15–18, 2020, Proceedings, Part II 29, Springer. pp. 708–720.

Wang, W.Y., 2017. " liar, liar pants on fire": A new benchmark dataset for fake news detection. arXiv preprint arXiv:1705.00648 .

Wang, Y., Wang, C., Zhan, J., Ma, W., Jiang, Y., 2023. Text fcg: Fusing contextual information via graph learning for text classification. Expert Systems with Applications 219, 119658. URL: https://www.sciencedirect.com/science/article/pii/S0957417423001598, doi:https://doi.org/10.1016/j.eswa.2023.119658.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T.L., Gugger, S., Drame, M., Lhoest, Q., Rush, A.M., 2020. Transformers: State-of-the-art natural language processing, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics, Online. pp. 38–45. URL: https://www.aclweb.org/anthology/2020.emnlp-demos.6.

Worsham, J., Kalita, J., 2018. Genre identification and the compositional effect of genre in literature, in: Proceedings of the 27th international conference on computational linguistics, pp. 1963–1973.

Wu, C., Wu, F., Liu, J., Huang, Y., Xie, X., 2019a. Sentiment lexicon enhanced neural sentiment classification, in: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pp. 1091–1100.

Wu, J., Chen, X.Y., Zhang, H., Xiong, L.D., Lei, H., Deng, S.H., 2019b. Hyperparameter optimization for machine learning models based on bayesian optimization. Journal of Electronic Science and Technology 17, 26–40.

Yan, C., Liu, J., Liu, W., Liu, X., 2022. Research on public opinion sentiment classification based on attention parallel dual-channel deep learning hybrid model. Engineering Applications of Artificial Intelligence 116, 105448. URL: https://www.sciencedirect.com/science/article/pii/S0952197622004389, doi:https://doi.org/10.1016/j.engappai.2022.105448.

Yogatama, D., Kong, L., Smith, N.A., 2015. Bayesian optimization of text representations, in: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Lisbon, Portugal. pp. 2100–2105. URL: https://aclanthology.org/D15-1251, doi:10.18653/v1/D15-1251.

Yousef, R.N., Tiun, S., Omar, N., Alshari, E.M., 2020. Enhance medical sentiment vectors through document embedding using recurrent neural network. International Journal of Advanced Computer Science and Applications 11.

Yu, L.C., Wang, J., Lai, K.R., Zhang, X., 2017. Refining word embeddings for sentiment analysis, in: Proceedings of the 2017 conference on empirical methods in natural language processing, pp. 534–539.

Yue, C., Cao, H., Xu, G., Dong, Y., 2020. Attention model with multi-layer supervision for text classification, in: Proceedings of the 2020 5th International Conference on Mathematics and Artificial Intelligence, pp. 103–109.

Zhang, H., Zhang, J., 2020. Text graph transformer for document classification, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Online. pp. 8322–8327. URL: https://aclanthology.org/2020.emnlp-main.668, doi:10.18653/v1/2020.emnlp-main.668.

Zhang, X., Zhao, J., LeCun, Y., 2015. Character-level convolutional networks for text classification, in: Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., Garnett, R. (Eds.), Advances in Neural Information Processing Systems, Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper_files/paper/2015/file/250cf8b51c773f3f8dc8b4be867a9a02-Paper.pdf.

Zhou, X., Zafarani, R., 2020. A survey of fake news: Fundamental theories, detection methods, and opportunities. ACM Computing Surveys (CSUR) 53, 1–40.