Towards Reducing Misinformation and Toxic Content Using Cross-Lingual Text Summarization

Hoai Nam Tran, Udo Kruschwitz

University of Regensburg, Germany

Abstract

Misinformation has long been considered a major problem in our digital world, but automatically identifying it still remains a challenging issue. It becomes even more of a problem when tackling content written in languages other than English. We also note that much progress has been made in classifying short social media posts, but there are many other types of misinformation. We present steps towards addressing the problem by adopting ideas that have shown to be promising in related prior work, namely applying extractive and abstractive text summarization methods so that we can process documents of any length and by incorporating machine translation as part of our overall architecture. We consider misinformation as just one out of many types of content that should be identified automatically on the way to a healthier digital ecosystem and see toxic content such as hate speech as naturally falling within the same scope of our work. We demonstrate on several benchmark collections covering both misinformation and toxic content that our approach is robust and achieves competitive performance on these datasets. This offers plenty of scope for future work. To foster reproducibility, we make all code and models available to the community via GitHub and Hugging Face.

Keywords

Misinformation, Text summarization, Toxic content detection, Cross-lingual

1. Introduction

Fake News and *Hate Speech* have one thing in common: the aim is to spread toxicity and to bring harm to the world. They have now become serious and significant social and political issues [1]. How did we get there? One aspect is that users have been shown to get more easily persuaded and influenced by social media posts, causing them to change their attitude [2]. In combination with the excessive usage of social media, the desire for validation and the fear of rejection negatively impact our mental health, especially for teenagers and children [3]. Much progress has been made recently in addressing the challenge, often focussing on social media. Searching for relevant information with common information retrieval systems and natural language processing pipelines gets more complicated with the amount of harmful misinformation. The flood of toxic content and polarization leads to distrust in any news channel; e.g., only 26% of American adults trust any news media [4, 5], which is why we need to improve the quality of the information we consume. Most competitive approaches include

ROMCIR 2023: The 3rd Workshop on Reducing Online Misinformation through Credible Information Retrieval, held as part of ECIR 2023: the 45th European Conference on Information Retrieval, April 2-6, 2023, Dublin, Ireland
Hoai-Nam.Tran@student.ur.de (H. N. Tran); Udo.Kruschwitz@ur.de (U. Kruschwitz)
0000-0002-5503-0341 (U. Kruschwitz)

incorporating transformer-based models like BERT [6] to assist social media moderators and fact-checkers in combating harmful content. However, since news articles and popular blog posts are also affected by misinformation and hateful assertions, and transformer-based models have a limited input size (e.g., 512 for BERT), the challenge here is to find a way to also use these models effectively for longer texts. This is where we propose text summarization. The second motivation is the fact that very few languages can be considered resource-rich, making it desirable to tap into such resources when tackling toxic content in other languages.

This paper presents a framework combining automatic machine translation, text summarization, and classification, tackling misinformation and toxic content. We provide experimental results for several common benchmarks using both binary and multi-class classification. To foster reproducibility, we make all code, hyperparameters, and detailed result tables available via GitHub[1].

2. Related Work

This section provides an overview of related work in fake news detection, hate speech detection, multilingual machine translation, and text summarization. Since misinformation leads to toxic polarization, which again leads to abusive language, and hate speech is generally considered harmful, we consider both fields to fall within the scope of detecting toxic content.

Fake news detection and **hate speech detection (HSD)** are two active research areas, with research often guided by shared tasks and competitions, e.g., as part of CLEF, SemEval, or GermEval. While users usually try to write more engaging comments to achieve more user interactions, the user's "dark side" [7] is the posting of hateful comments, including toxic and offensive language. There are monolingual and multilingual approaches to detect hate speech and toxic comments since online comments can be written in different languages and possibly a mix of several. A number of different approaches can be adopted to tackle this task, and at a high level one can distinguish *content-based* and *context-based* methods [8]. We are focusing on content-based approaches. Rather than providing a review of this massive body of literature we just point out that *Transformer models* with self-attention [9] like BERT [6], BART [10], and T5 [11] dominate the field. HateBERT [12] is a BERT variant pre-trained with abusive online community data from the social news and discussion platform Reddit[2]. A provided list of criteria from tweets as predictive features can help to identify racist and sexist insults [13]. They are all Twitter-based in the HSD domain, similar to some of the datasets we use for our approach. A survey of datasets on the topic of fake news detection and fact verification includes several fact-checking sites [14]. Full Fact[3] is an example of a fact-checking organisation that aims at identifying harmful content with intelligence and monitoring tools, e.g., CrowdTangle, which helps the user with manual fact-claim checking by raising alerts if exact user-defined keywords are triggered [15]. *Multilinguality* is commonly addressed by using transformer models trained on multiple languages, e.g., XLM-RoBERTa [16] has cross-lingual capabilities to work in several tasks and benchmarks containing harmful texts which can appear in multiple languages. Fusion

[1]https://github.com/HN-Tran/ROMCIR_2023
[2]https://www.reddit.com/
[3]https://fullfact.org/

strategies with mBERT and XLM-RoBERTa for multilingual toxic text detection [17] or deep learning ensembles for effective hate speech detection are other approaches that are similar to ours. Given that multilingual models still focus on a limited number of languages for pre-training, we explore automatic machine translation as an alternative.

Machine translation is an essential part of many online services nowadays. In a survey by CSA Research, 76% of online shoppers prefer information in their native language, and 40% would never buy from websites with only other languages [18]. Also, the global machine translation market has increased from 450 million USD in 2017 to 1.1 billion USD in 2022 [19, 20]. The two most popular translation services are Google Translate [21] and Microsoft Translator [22]. Both services are available in multiple languages and can be used for free. DeepL Translator is a relatively new translation service that uses proprietary neural networks to translate text [23]. They claim to surpass Google Translate and Microsoft Translator in terms of quality and speed in several European languages [24]. These services are however not always accurate and can even be exploited by malicious actors [25, 26]. Since transformer-based models and automatic translation services are limited in their input length, summarization is our approach to overcome this limitation.

Summarization is still an active research field that successfully utilizes **extractive** machine learning [27, 28] and **abstractive** approaches [29, 30, 31]. It has only recently been considered in this context with state-of-the-art performance for fake news detection using a common reference benchmark collection reported [32]. We propose to utilize progress in the field by providing a novel combination of established techniques, leaving plenty of room for future work to explore this idea further.

In summary, we observe that misinformation and toxic content detection are conceptually related areas that remain open problems despite progress that has been made in recent years. We are interested in exploring content-based ideas that have shown promise in previous work and see our contribution as one possible direction that utilises different types of automatic text summarisation as well as machine translation. Future work can then explore this in more depth and breadth.

3. Methodology

Our general framework is a pipeline-based architecture, as illustrated in Figure 1. It has three main components: automatic machine translation, summarization, and classification. Due to the availability of common benchmarks in German, it is our language of choice for the source documents (but we obviously envisage this approach to be applied to actually under-resourced languages in future work). Each dataset gets machine-translated into English which is followed by transformer-based text summarization. German-based models take the original texts/comments and summarized texts as the input in the fine-tuning process, while domain-specific and multilingual models use the translations. We train our models 5 times (runs) with a different seed, where the model with the highest macro F1 score in each 50 steps for each run gets chosen, and the inference outputs the predictions of each model. After that, all the 5 runs get ensembled in both majority voting types (hard and soft voting). Finally, the ensembled models are ensembled again. Finding the optimal number of models for the ensemble is difficult

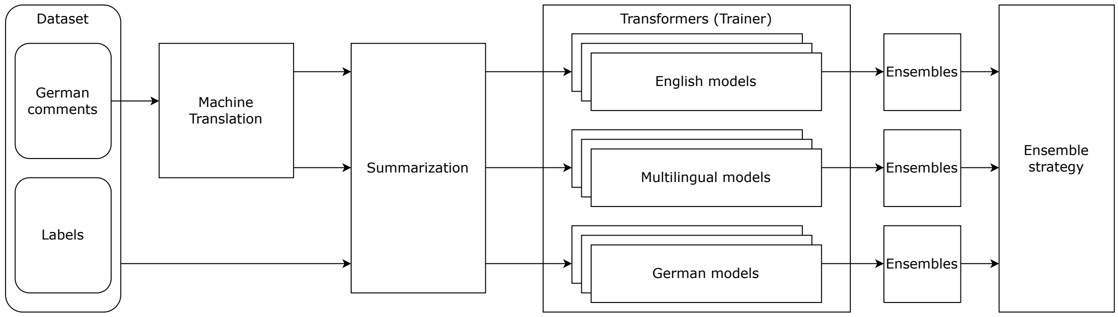

Figure 1: General Architecture

because of the danger of overfitting and averaging. Thus we set a fixed number of 5 runs.

German tasks and resources for training are sparse since English is the most represented language in many benchmarks (~1000 tasks). German (~30 tasks) is even less represented than other languages like Spanish (~60 tasks), Hindi (~45 tasks), and even Bengali (~35 tasks) [33]. Thus, we have decided on bilingual German and English datasets that can later be applied to other languages.

For fine-tuning, we use the recommended arithmetic mean over harmonic mean (see Equation 1) due to its robustness towards error type distribution [34]:

$$\mathcal{F}_1 = \frac{1}{n}\sum_x \mathrm{F1}_x = \frac{1}{n}\sum_x \frac{2P_xR_x}{P_x+R_x} \tag{1}$$

As the GermEval metric, we use the harmonic mean over arithmetic mean (see Equation 2):

$$\mathbb{F}_1 = H(\bar{P},\bar{R}) = \frac{2\bar{P}\bar{R}}{\bar{P}+\bar{R}} = 2\frac{(\frac{1}{n}\sum_x P_x)(\frac{1}{n}\sum_x R_x)}{\frac{1}{n}\sum_x P_x + \frac{1}{n}\sum_x R_x} \tag{2}$$

4. Experiments

Here we will briefly outline the experimental setup, choice of tools, and datasets.

4.1. Data

We use the following shared task datasets (there is clearly scope to explore other, less-resourced languages and classification tasks in future work):

- GermEval 2018 Subtask 1 [35]
- GermEval 2019 Task 2 Subtask 1 [36]
- GermEval 2021 Subtasks 1-3 [37]
- CLEF 2022 CheckThat! Lab Task 3 [38]

Table 1
GermEval Dataset Sizes

		GermEval 2021			GermEval 2018	GermEval 2019 Task 2
Dataset	**Label**	**Subtask 1**	**Subtask 2**	**Subtask 3**	**Subtask 1**	**Subtask 1**
Training	True	1122	865	1103	1688	1287
	False	2122	2379	2141	3321	2707
Test	True	350	253	314	1202	970
	False	594	691	630	2330	2061

Table 2
CLEF CheckThat! 2022 Dataset Sizes

	CLEF CheckThat! 2022			
	Training Set	Development Set	Test Set	
Label	**Subtask 3**	**Subtask 3**	**Subtask 3A**	**Subtask 3B**
True	142	69	210	243
False	465	113	315	191
Partially False	217	141	56	97
Other	76	41	31	55

Table 3
Character Length Distribution for CheckThat! 2022

Variables	**Statistics**	**Training Set**	**Development Set**	**Test Set 3A**	**Test Set 3B**
Title	Median	70	66	73	67
	Mean	286	171	78	71
	Minimum	3	3	11	3
	Maximum	9960	8092	200	234
Text	Median	3035	3115	3655	4009
	Mean	4167	4498	6052	5617
	Minimum	18	25	289	507
	Maximum	32767	44359	100000	45309

All datasets contain an imbalance in their class label distributions (see Table 1), and the number of characters are also very different. GermEval datasets fit the short text scenario since they are comments from social networks and the CheckThat! dataset fits the long text scenario with a maximum size of 100,000 characters (see Table 3). Since BERT models usually have a maximum token limit of 512, the input would automatically be truncated to the first 512 tokens, and thus relevant information might get lost in this process.

4.2. Automatic Machine Translation

For machine translation, we can choose between a text generation model like T5 [39] or commercial translation services for our purpose. Since the translation quality depends on

Table 4
Hyperparameters

	GermEval				CLEF CheckThat! 2022	
Hyperparameters	**GBERT GELECTRA**	**BERT-based**	**XLM-R BERTweet**	**T5-based**	**BERT-based**	**T5-based**
Learning rate	5e-6	2e-5	1e-5	1e-3	1e-5	4e-5
Max Steps	705			—	705	—
Max Epochs	—			200	—	200
Evaluation Steps	50			2000	50	2000
Early Stopping	no			yes	no	yes
Batch Size	32			2 – 32	32	4
Max Sequence Length	128 (GermEval 2021) 150 (GermEval 2018, 2019 Task 2)				256	

its pre-trained corpora quantity and quality, we decided on the two most popular machine translation services: Google Translate and DeepL Translator.

4.3. Splitting Methods

There are two standard options for deciding how to split our data for the fine-tuning process: Fixed random seeds and k-fold cross-validation. In [40], they used random seeding, and since we made five runs with an imbalanced dataset, a stratified 5-fold cross-validation was the other option. Our results show that using fixed random seed values is better than using stratified k-fold cross-validation.

4.4. Hyperparameter Optimization

Since searching for the optimal hyperparameters for our models is difficult, especially looking for ways to avoid overfitting, we use the Optuna [41] library, which can be integrated into the Hugging Face Trainer library as an option for hyperparameter search. Since even the default hyperparameters can lead to overfitting in specific benchmark datasets, the chance of having similar data points between the development and test set is given. We tested 100 combinations, which evaluates the best possible setting for our macro-F1 metric (see Equation 1). The question is whether a complete automatic hyperparameter search can be conducted by a tool like Optuna to work effectively without looking for any working hyperparameters.

After choosing the first three best runs, the results show that it is an appropriate way to find parameters for the development set but not for the test set. We have not tested this on a fixed amount of known default hyperparameters yet. Since hyperparameter optimization is also a very time-consuming process, we have decided to use each model's recommended hyperparameters if reported in the corresponding papers. If not, we use the exact parameters of their model architecture.

4.5. Summarization

We use both extractive and abstractive summarization separately and exclusively. Since we only have one textual input, we first concatenate the title and the text with a dot so that the title is considered the first sentence (see Equation 3). Sometimes, the title is written like clickbait, a sentence without any information-relevant value.

$$title + .\textvisiblespace + text \tag{3}$$

Extraction-based summarization aims to select the most relevant representations of the given text input. In the used library [42], we apply k-means clustering and use the Elbow method to find the optimal k [43, 44]. Our chosen model is DistilBART-CNN-12-6[4] which is based on BART [10] with distillation [45], fine-tuned with the CNN and DailyMail dataset [46].

Abstraction-based summarization aims to generate shorter text with the most relevant representations of the given text input. We use the version of T5 [11] with three billion parameters (T5-3B) to generate shorter text with the identical prompt template ("*summarize:*") used in the pre-training process for the CNN/DailyMail dataset [46]. With the use of relative positional embeddings, the utilization of much longer text at the cost of higher computing consumption is possible [47, 11].

4.6. Classification Tasks

Longer text can contain more information, therefore we often need more labels to classify them and thus show two different classification types: Binary classification for two labels, and multi-class classification for more than two class labels. At the end of the pipeline, we ensemble the results of the summarization and classification tasks to get the final result.

4.6.1. Binary Classification

If the text is short, possibly containing a single sentence (as is often the case in social media), the labels might be "true" and "false" or "toxic" and "non-toxic". However, other labels might be used (such as "*other*"), turning the task into a multi-label classification. The chosen GermEval datasets have two labels; thus, only the machine translation before is needed for fine-tuning. We have decided for BERT [6] as our English-based model, GBERT and GELECTRA [40] as our German-based model, XLM-RoBERTa [16] as our multilingual model with both German and English input, and BERTweet [48] as our Twitter-based model. After five runs, they get ensembled together in hard and soft majority voting (see Table 8). Then again, we choose the best five model ensembles (in GermEval 2018 and 2019, the best three) and ensemble them in three different ensembling strategies: Majority Voting (both hard and soft voting), Gradient Boosting Machines and Logistic Regression (see Table 8).

4.6.2. Multi-Class Classification

Long texts typically contain more sentences and possibly a broader spread of topics. This leads to classification tasks that go beyond a simple binary decision (e.g., one might consider

[4] https://huggingface.co/sshleifer/distilbart-cnn-12-6

Table 5
Machine Translation Performance

Translation Service	Run 1	Run 2	Run 3	Run 4	Run 5	Hard	Soft
Google Translate	69.48	67.08	67.67	68.74	68.28	68.39	68.42
DeepL Translator	70.01	70.22	69.24	68.26	67.67	70.13	70.09

"partially false" or "partially true"). The CheckThat! 2022 dataset has four different class labels with imbalanced distributions. For the classification process, we use three large models: BERT Uncased [6], XLM-RoBERTa [16], and T5-3B [11].

4.7. General Setup

All of our experiments are conducted on the following datasets with the following GPUs: the GermEval 2018 and 2019 datasets with GTX/RTX 1080/2080 Ti (11 GB VRAM) including GermEval 2021 base models, Tesla V100S (32 GB VRAM) for the large models in the GermEval 2021 datasets, and the CheckThat! 2022 datasets with RTX A6000 with 48 GB VRAM. We use the SimpleTransformers library[5] for the T5 model and all other transformer models with the Hugging Face Transformers library[6]. For the summarization task, we use the BERT Extractive Summarizer library[7] [42], and for machine translation, we use the deep-translator library[8] in combination with the free public Google Translate service[9] and the pro version of the DeepL Translator service[10]. Our hyperparameters are in Table 4.

5. Results

We observe that our approach is highly competitive and robust for both types of classification and all datasets.

5.1. Machine Translation

We would first like to report some insights into the choice of Machine Translation tools. The results show that for this experiment DeepL Translator appears to be a better choice than Google Translate, but the score difference is very close, so both are solid choices (see Table 5). For the GermEval datasets, we apply the machine translations of the DeepL Translator service. For the CheckThat! 2022 dataset, since the maximum of the text, can be at 100,000 characters, we use the free Google Translate service as a financial constraint. Since the service has an internal character limit, we only take the first 5,000 characters for translation.

[5]https://simpletransformers.ai/
[6]https://huggingface.co/transformers
[7]https://github.com/dmmiller612/bert-extractive-summarizer
[8]https://github.com/nidhaloff/deep-translator
[9]https://translate.google.com/
[10]https://www.deepl.com/en/pro#developer

Table 6
Random Hyperparameter Tuning with Optuna

Run	Dataset	With Hyperparameter Tuning	Without Hyperparameter Tuning
1	Development Set	70.49	70.57
	Test Set	67.61	67.67
2	Development Set	70.44	70.43
	Test Set	68.15	70.22
3	Development Set	69.81	69.81
	Test Set	67.25	68.26

Table 7
Splitting Strategy

Splitting Strategy	Run 1	Run 2	Run 3	Run 4	Run 5	Hard	Soft
Stratified K-Fold Cross-Validation	68.78	67.79	68.21	69.62	67.29	68.99	69.59
Random Seed	70.01	70.22	69.24	68.26	67.67	70.13	70.09

5.2. Hyperparameter Optimization

Table 6 shows that the difference between the use of hyperparameter search is marginal and even worse on the test set. It shows that tuning more into the development set leads to worse results on the test set, especially visible in the third run. While this indicates overfitting and the general preference for generalization, the generally worse results on the development set with hyperparameter tuning cannot be explained with overfitting but with too many other factors to consider. Also, the drawback of the search duration makes this step insignificant and redundant. That is why we continued with the default hyperparameters.

5.3. Splitting Methods

As shown in Table 7, the difference after majority voting is minor, and thus both strategies are eligible. If we look at each run, the difference is also very narrow. Thus, picking up a splitting strategy is not essential and is not a deciding factor in the system architecture. We decide to continue with random seeding.

5.4. Binary Classification and Ensembling

For all GermEval datasets, we observe the potential for improvement over previously reported SOTA results (see Table 8). For GermEval 2021 Subtask 1, the score improvement is noticeable at 4.48% compared to the highest score reported so far. Except for GermEval 2021 Subtask 2, where all results are more or less on par (which might in part be an issue with the gold standard labels), all other results demonstrate the added value our approach offers.

Of all the ensembling strategies, the popular majority voting is still the most effective one. Since Gradient Boosting Machines and Logistic Regression are both linear models, we expect

Table 8
Binary Classification on GermEval datasets (new best performance in **bold**)

	GermEval '18 Subtask 1		GermEval '19 T2 Subtask 1		GermEval '21 Subtask 1		GermEval '21 Subtask 2		GermEval '21 Subtask 3	
Model	**Hard**	**Soft**	**Hard**	**Soft**	**Hard**	**Soft**	**Hard**	**Soft**	**Hard**	**Soft**
$\text{GBERT}_{\text{base}}$	76.28	75.91	76.50	76.64	68.11	67.84	68.22	68.30	74.23	74.78
$\text{GELECTRA}_{\text{base}}$	75.45	75.37	75.15	74.92	69.38	69.68	67.96	67.60	76.52	77.11
$\text{BERTweet}_{\text{base}}$	78.02	78.05	77.23	77.44	70.13	70.09	68.23	68.84	75.51	75.47
$\text{BERT}_{\text{base}}$	77.23	77.17	76.63	76.55	64.68	64.71	68.89	69.39	73.36	72.71
$\text{XLM-R}_{\text{base}}$ (de)	75.71	76.00	75.51	75.17	67.37	67.21	68.49	67.90	73.84	74.26
$\text{XLM-R}_{\text{base}}$ (en)	76.67	77.04	77.35	77.11	68.24	68.20	69.11	69.72	74.35	74.61
$\text{GBERT}_{\text{large}}$	80.74	80.63	80.06	80.23	72.09	72.69	69.45	68.89	75.77	76.10
$\text{GELECTRA}_{\text{large}}$	80.06	79.85	80.80	80.79	71.62	71.72	70.16	70.24	75.06	74.26
$\text{BERTweet}_{\text{large}}$	79.97	79.86	79.56	79.86	73.60	72.24	69.82	**70.36**	75.14	75.48
$\text{BERT}_{\text{large}}$	78.34	78.32	77.79	77.79	67.00	65.26	69.86	69.47	74.58	75.07
$\text{XLM-R}_{\text{large}}$ (de)	-	-	-	-	69.04	69.12	69.51	68.60	76.36	76.82
$\text{XLM-R}_{\text{large}}$ (en)	-	-	-	-	71.71	71.48	68.77	69.99	76.54	77.44
Ensemble	**Hard**	**Soft**	**Hard**	**Soft**	**Hard**	**Soft**	**Hard**	**Soft**	**Hard**	**Soft**
Gradient Boosting	79.97	80.95	80.28	81.77	**76.23**	74.03	68.25	69.47	75.82	76.12
Logistic Regression	79.97	80.91	81.14	81.52	74.11	75.09	68.56	69.65	75.61	74.03
Majority Voting	80.99	**81.48**	82.06	**82.36**	75.22	74.72	69.22	70.09	**77.82**	76.89
SOTA	80.70 [40]		76.95 [49]		71.75 [50]		69.98 [51]		76.26 [50]	

that the linear combination of their predictions will be more effective than the majority voting. However, the results show that the majority voting is still the best strategy.

5.5. Summarization and Multi-Class Classification

As shown in Table 9, the combination of summarization and classification leads to noticeable improvements (e.g., 5.63% for Task 3A). Unlike in the previous experiments, here we do not apply ensembling, which could lead to further improvements in robustness and overall results. An interesting observation here is the discrepancy between development sets and the test set results.

6. Limitations and Future Work

The results show that our approach is robust and achieves state-of-the-art performance on these datasets. That offers plenty of directions for future work. However, before we can start with future work, we need to discuss the limitations of our approach.

The first limitation is the fact that the hyperparameter search was random. A fixed scope of hyperparameters might have led to better results for training. Another limitation is that we have summarized every data point in the dataset. That means that even short text snippets were summarized. We do not use the DeepL Translator service for all experiments because

Table 9
Multi-Class Classification on CheckThat! 2022 dataset (new SOTA in **bold**)

Summarization Model	Classification Model	Run Nr.	Dev	Dev-Test	Test 3A	Test 3B
DistilBART-CNN-12-6 (extractive)	$BERT_{large}$	1	52.40	52.18	28.33	28.99
		2	46.43	39.96	26.87	19.46
		3	48.77	52.78	30.70	28.69
		4	49.21	48.44	32.31	25.32
		5	53.25	51.85	30.19	20.46
	$XLM\text{-}R_{large}$	1	50.53	41.04	30.42	27.40
		2	50.93	44.54	33.11	28.01
		3	49.08	48.56	30.82	26.09
		4	50.80	43.99	28.23	21.94
		5	50.95	40.29	32.47	23.34
	T5-3B	1	48.05	46.52	**39.54**	29.58
T5-3B (abstractive)	$BERT_{large}$	1	56.33	51.15	28.89	21.34
		2	45.85	37.87	32.88	23.43
		3	55.08	46.80	35.24	28.33
		4	52.15	47.08	36.48	27.01
		5	51.32	46.91	30.56	21.77
	$XLM\text{-}R_{large}$	1	51.54	44.81	31.66	28.99
		2	49.36	42.84	35.63	**30.06**
		3	49.73	44.91	35.67	27.82
		4	50.59	44.79	36.01	26.86
		5	51.78	40.25	35.29	28.09
	T5-3B	1	52.08	43.82	29.72	23.72
SOTA					33.91 [52]	28.99 [53]

the free version is limited to 500,000 characters per month[11] and one data point of the CLEF CheckThat! 2022 dataset already hits 100,000 characters. Since the performance difference is very close, we decided to use the free Google Translate service for the CheckThat! 2022 dataset. We also want to warn about possible outputs caused by "model hallucination," which is not yet usable for production.

As future work, the investigation of text generation with bigger models like GPT-3 [54], ChatGPT [55], PaLM [56], Flan-T5 [33], and others is interesting to see if our approach will improve by simply having more parameters and more pre-trained data. Text generation tasks like machine translation or summarization would benefit the increased accuracy of the models and thus would lead to a real-world production environment to tackle fake news and hate speech. Especially in the summarization task, we want to understand if summarizing text snippets below 512 tokens makes a difference in performance. The increased performance by summarization opens the question of why exactly it works and remains contentious. Another open question is what the optimal amount of models for the ensemble is, where a correlation between amount of dataset and diversity of models needs to be explored. Another important question is how each module of the pipeline, especially summarization and machine translation, work separately on a larger scale. Different benchmark datasets for each different tasks are needed to investigate the performance of each module.

[11] https://www.deepl.com/en/pro#developer

7. Conclusion

We propose a general architecture to deal with text classification in a cross-lingual context tapping into resources available for high-resourced languages and making use of abstractive and extractive summarization. We demonstrate the potential that this approach offers using existing non-English benchmark collections for fake news and hate speech classification. This lays the groundwork for future work, which should look at a range of low-resource languages.

Acknowledgments

This work was supported by the project *COURAGE: A Social Media Companion Safeguarding and Educating Students* funded by the Volkswagen Foundation, grant number 95564. We want to thank all reviewers for their insightful comments that helped us to improve our work.

References

[1] P. Nakov, D. P. A. Corney, M. Hasanain, F. Alam, T. Elsayed, A. Barrón-Cedeño, P. Papotti, S. Shaar, G. D. S. Martino, Automated fact-checking for assisting human fact-checkers, in: Z. Zhou (Ed.), Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021, ijcai.org, 2021, pp. 4551–4558.

[2] Y. Wang, Y. Dai, H. Li, L. Song, Social media and attitude change: Information booming promote or resist persuasion?, Frontiers in Psychology 12 (2021).

[3] H. Fersko, Is social media bad for teens' mental health?, 2018. URL: https://www.unicef.org/stories/social-media-bad-teens-mental-health.

[4] N. Newman, R. Fletcher, C. T. Robertson, K. Eddy, R. K. Nielsen, Digital news report 2022, 2022. URL: https://reutersinstitute.politics.ox.ac.uk/sites/default/files/2022-06/Digital_News-Report_2022.pdf.

[5] Reuters Institute for the Study of Journalism, Share of adults who trust news media most of the time in selected countries worldwide as of February 2022 [Graph], 2022. URL: https://www.statista.com/statistics/308468/importance-brand-journalist-creating-trust-news/.

[6] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. URL: https://aclanthology.org/N19-1423. doi:10.18653/v1/N19-1423.

[7] G. L. de Holanda Coelho, P. H. P. Hanel, R. P. Monteiro, R. Vilar, V. V. Gouveia, The dark side of human values: How values are related to bright and dark personality traits, The Spanish Journal of Psychology 24 (2021).

[8] K. Sharma, F. Qian, H. Jiang, N. Ruchansky, M. Zhang, Y. Liu, Combating fake news: A survey on identification and mitigation techniques, ACM Transactions on Intelligent Systems and Technology (TIST) 10 (2019) 1–42. Publisher: ACM New York, NY, USA.

[9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS'17, Curran Associates Inc., USA, 2017, pp. 6000–6010. URL: http://dl.acm.org/citation.cfm?id=3295222.3295349.

[10] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, L. Zettlemoyer, BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 7871–7880. URL: https://aclanthology.org/2020.acl-main.703. doi:10.18653/v1/2020.acl-main.703.

[11] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, Exploring the limits of transfer learning with a unified text-to-text transformer, Journal of Machine Learning Research 21 (2020) 1–67. URL: http://jmlr.org/papers/v21/20-074.html.

[12] T. Caselli, V. Basile, J. Mitrović, M. Granitzer, HateBERT: Retraining BERT for abusive language detection in English, in: Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021), Association for Computational Linguistics, Online, 2021, pp. 17–25.

[13] Z. Waseem, D. Hovy, Hateful symbols or hateful people? predictive features for hate speech detection on Twitter, in: Proceedings of the NAACL Student Research Workshop, Association for Computational Linguistics, San Diego, California, 2016, pp. 88–93.

[14] T. Murayama, Dataset of fake news detection and fact verification: A survey, ArXiv abs/2111.03299 (2021).

[15] P. Arnold, The challenges of online fact checking, Technical Report, Full Fact, London, UK, 2020. URL: https://fullfact.org/media/uploads/coof-2020.pdf.

[16] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised cross-lingual representation learning at scale, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 8440–8451. URL: https://aclanthology.org/2020.acl-main.747. doi:10.18653/v1/2020.acl-main.747.

[17] G. Song, D. Huang, Z. Xiao, A study of multilingual toxic text detection approaches under imbalanced sample distribution, Information (Switzerland) 12 (2021) 1–16.

[18] CSA Research, Survey of 8,709 Consumers in 29 Countries Finds that 76% Prefer Purchasing Products with Information in their Own Language, 2020. URL: https://csa-research.com/Blogs-Events/CSA-in-the-Media/Press-Releases/Consumers-Prefer-their-Own-Language.

[19] Global Market Insights Inc., Machine translation market size worldwide, from 2016 to 2024 (in million U.S. dollars) [Graph], 2017. URL: https://www.statista.com/statistics/748358/worldwide-machine-translation-market-size/.

[20] P. Wadhwani, S. Gankar, Machine translation market size: Industry analysis, 2022-2030, 2022. URL: https://www.gminsights.com/industry-analysis/machine-translation-market-size.

[21] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, J. Klingner, A. Shah, M. Johnson, X. Liu, L. Kaiser, S. Gouws, Y. Kato, T. Kudo, H. Kazawa, K. Stevens, G. Kurian, N. Patil, W. Wang, C. Young, J. R. Smith, J. Riesa, A. Rudnick, O. Vinyals, G. S. Corrado, M. Hughes, J. Dean, Google's neural machine

translation system: Bridging the gap between human and machine translation, ArXiv abs/1609.08144 (2016).
[22] M. Junczys-Dowmunt, Microsoft translator at WMT 2019: Towards large-scale document-level neural machine translation, in: Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1), Association for Computational Linguistics, Florence, Italy, 2019, pp. 225–233. URL: https://aclanthology.org/W19-5321. doi:10.18653/v1/W19-5321.
[23] D. Coldewey, F. Lardinois, DeepL schools other online translators with clever machine learning, 2017. URL: https://techcrunch.com/2017/08/29/deepl-schools-other-online-translators-with-clever-machine-learning/.
[24] DeepL, Translation quality, ???? URL: https://www.deepl.com/en/quality.html.
[25] J. Fuchs, Spoofing Google Translate to Steal Credentials, 2022. URL: https://www.avanan.com/blog/spoofing-google-translate-to-steal-credentials.
[26] E. Montalbano, Cyberattackers spoof google translate in unique phishing tactic, 2022. URL: https://www.darkreading.com/threat-intelligence/cyberattackers-spoof-google-translate-unique-phishing-tactic.
[27] J. L. Neto, A. A. Freitas, C. A. A. Kaestner, Automatic text summarization using a machine learning approach, in: Brazilian Symposium on Artificial Intelligence, 2002.
[28] R. Nallapati, F. Zhai, B. Zhou, Summarunner: A recurrent neural network based sequence model for extractive summarization of documents, ArXiv abs/1611.04230 (2016).
[29] S. Chopra, M. Auli, A. M. Rush, Abstractive sentence summarization with attentive recurrent neural networks, in: North American Chapter of the Association for Computational Linguistics, 2016.
[30] R. Nallapati, B. Zhou, C. N. dos Santos, Çaglar Gülçehre, B. Xiang, Abstractive text summarization using sequence-to-sequence rnns and beyond, in: Conference on Computational Natural Language Learning, 2016.
[31] R. Paulus, C. Xiong, R. Socher, A deep reinforced model for abstractive summarization, ArXiv abs/1705.04304 (2017).
[32] P. Hartl, U. Kruschwitz, Applying automatic text summarization for fake news detection, in: Proceedings of the 13th Language Resources and Evaluation Conference (LREC 2022), 2022, pp. 2702-–2713.
[33] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, E. Li, X. Wang, M. Dehghani, S. Brahma, A. Webson, S. S. Gu, Z. Dai, M. Suzgun, X. Chen, A. Chowdhery, D. Valter, S. Narang, G. Mishra, A. W. Yu, V. Zhao, Y. Huang, A. M. Dai, H. Yu, S. Petrov, E. H. hsin Chi, J. Dean, J. Devlin, A. Roberts, D. Zhou, Q. Le, J. Wei, Scaling instruction-finetuned language models, ArXiv abs/2210.11416 (2022).
[34] J. Opitz, S. Burst, Macro f1 and macro f1, arXiv preprint arXiv:1911.03347 (2019).
[35] M. Wiegand, M. Siegel, J. Ruppenhofer, Overview of the germeval 2018 shared task on the identification of offensive language, Proceedings of GermEval 2018, 14th Conference on Natural Language Processing (KONVENS 2018), Vienna, Austria – September 21, 2018, Austrian Academy of Sciences, Vienna, Austria, 2019, pp. 1 – 10. URL: https://nbn-resolving.org/urn:nbn:de:bsz:mh39-84935.
[36] J. M. Struß, M. Siegel, J. Ruppenhofer, M. Wiegand, M. Klenner, Overview of germeval task 2, 2019 shared task on the identification of offensive language, Preliminary proceedings

of the 15th Conference on Natural Language Processing (KONVENS 2019), October 9 – 11, 2019 at Friedrich-Alexander-Universität Erlangen-Nürnberg, German Society for Computational Linguistics & Language Technology und Friedrich-Alexander-Universität Erlangen-Nürnberg, München [u.a.], 2019, pp. 352 – 363. URL: https://nbn-resolving.org/urn:nbn:de:bsz:mh39-93197.
[37] J. Risch, A. Stoll, L. Wilms, M. Wiegand, Overview of the GermEval 2021 shared task on the identification of toxic, engaging, and fact-claiming comments, in: Proceedings of the GermEval 2021 Shared Task on the Identification of Toxic, Engaging, and Fact-Claiming Comments, Association for Computational Linguistics, Duesseldorf, Germany, 2021, pp. 1–12.
[38] J. Köhler, G. K. Shahi, J. M. Struß, M. Wiegand, M. Siegel, T. Mandl, M. Schütz, Overview of the clef-2022 checkthat! lab task 3 on fake news detection, Working Notes of CLEF (2022).
[39] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, Exploring the limits of transfer learning with a unified text-to-text transformer, Journal of Machine Learning Research 21 (2020) 1–67.
[40] B. Chan, S. Schweter, T. Möller, German's next language model, in: Proceedings of the 28th International Conference on Computational Linguistics, International Committee on Computational Linguistics, Barcelona, Spain (Online), 2020, pp. 6788–6796.
[41] T. Akiba, S. Sano, T. Yanase, T. Ohta, M. Koyama, Optuna: A next-generation hyperparameter optimization framework, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD '19, Association for Computing Machinery, New York, NY, USA, 2019, p. 2623–2631.
[42] D. Miller, Leveraging bert for extractive text summarization on lectures, arXiv preprint arXiv:1906.04165 (2019).
[43] J. MacQueen, et al., Some methods for classification and analysis of multivariate observations, in: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, volume 1, Oakland, CA, USA, 1967, pp. 281–297.
[44] T. M. Kodinariya, P. R. Makwana, Review on determining number of cluster in k-means clustering, International Journal 1 (2013) 90–95.
[45] S. Shleifer, A. M. Rush, Pre-trained summarization distillation, ArXiv abs/2010.13002 (2020).
[46] K. M. Hermann, T. Kocisky, E. Grefenstette, L. Espeholt, W. Kay, M. Suleyman, P. Blunsom, Teaching machines to read and comprehend, in: C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, R. Garnett (Eds.), Advances in Neural Information Processing Systems, volume 28, Curran Associates, Inc., 2015. URL: https://proceedings.neurips.cc/paper/2015/file/afdec7005cc9f14302cd0474fd0f3c96-Paper.pdf.
[47] P. Shaw, J. Uszkoreit, A. Vaswani, Self-attention with relative position representations, in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), Association for Computational Linguistics, New Orleans, Louisiana, 2018, pp. 464–468. URL: https://aclanthology.org/N18-2074. doi:10.18653/v1/N18-2074.
[48] D. Q. Nguyen, T. Vu, A. T. Nguyen, BERTweet: A pre-trained language model for English Tweets, in: Q. Liu, D. Schlangen (Eds.), Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, EMNLP 2020 - Demos,

Online, November 16-20, 2020, Association for Computational Linguistics, 2020, pp. 9–14.
[49] A. Paraschiv, D.-C. Cercel, Upb at germeval-2019 task 2: Bert-based offensive language classification of german tweets., in: KONVENS, 2019.
[50] T. Bornheim, N. Grieger, S. Bialonski, FHAC at GermEval 2021: Identifying German toxic, engaging, and fact-claiming comments with ensemble learning, in: Proceedings of the GermEval 2021 Shared Task on the Identification of Toxic, Engaging, and Fact-Claiming Comments, Association for Computational Linguistics, Duesseldorf, Germany, 2021, pp. 105–111.
[51] N. Hildebrandt, B. Boenninghoff, D. Orth, C. Schymura, Data science kitchen at GermEval 2021: A fine selection of hand-picked features, delivered fresh from the oven, in: Proceedings of the GermEval 2021 Shared Task on the Identification of Toxic, Engaging, and Fact-Claiming Comments, Association for Computational Linguistics, Duesseldorf, Germany, 2021, pp. 88–94. URL: https://aclanthology.org/2021.germeval-1.13.
[52] B. Taboubi, M. A. B. Nessir, H. Haddad, icompass at checkthat! 2022: combining deep language models for fake news detection, Working Notes of CLEF (2022).
[53] H. N. Tran, U. Kruschwitz, ur-iw-hnt at CheckThat! 2022: cross-lingual text summarization for fake news detection, in: G. Faggioli, N. Ferro, A. Hanbury, M. Potthast (Eds.), Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum, CLEF '2022, Bologna, Italy, 2022.
[54] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei, Language models are few-shot learners, in: H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin (Eds.), Advances in Neural Information Processing Systems, volume 33, Curran Associates, Inc., 2020, pp. 1877–1901. URL: https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.
[55] OpenAI, ChatGPT: Optimizing Language Models for Dialogue, 2022. URL: https://openai.com/blog/chatgpt/.
[56] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, N. Fiedel, PaLM: Scaling Language Modeling with Pathways, arxiv:2204.02311 (2022). URL: https://arxiv.org/abs/2204.02311.