Proceedings of the GermEval 2021 Workshop on the Identification of Toxic, Engaging, and Fact-Claiming Comments

17th Conference on Natural Language Processing
KONVENS 2021

Heinrich Heine University Düsseldorf
September 6, 2021

Edited by

Julian Risch
Anke Stoll
Lena Wilms
Michael Wiegand

Preface

User-generated content on the web, particularly on social media, has become a regular part of our everyday life. Given the heavy increase of such content within the last decade, the demand for approaches to classify online content automatically is more pressing than ever. To this end, we present the GermEval 2021 Workshop on the Identification of Toxic, Engaging, and Fact-Claiming Comments. This shared task deals with the classification of Facebook posts that were drawn from the Facebook page of a German political talk show of a national public television broadcaster. GermEval 2021 is the seventh workshop in a series of shared tasks on German processing that was started in 2014. Changing groups of researchers self-organize the shared tasks and they are endorsed by special interest groups within the German Society for Computational Linguistics (GSCL). The workshops are co-located with the Conference on Natural Language Processing (KONVENS), which is held in Düsseldorf in 2021. The results of this year's shared task show that state-of-the-art classification approaches perform well on all three subtasks and achieve macro-average F1-scores between 70% and 76% but still leave room for improvement. We received 87 submissions from 15 participating teams. The results and the full dataset can be found at the shared task website at https://germeval2021toxic.github.io/SharedTask/. We are grateful to the large number of participants whose enthusiastic participation made GermEval 2021 a great success. We would like to extend our gratitute to the KONVENS 2021 conference organizers for their support and to University Library Klagenfurt (netlibrary) for making the publication of the workshop proceedings possible.

Düsseldorf, September 2021

The organizing committee

Organizers:
Julian Risch (deepset)
Anke Stoll (Heinrich Heine University Düsseldorf)
Lena Wilms (Heinrich Heine University Düsseldorf)
Michael Wiegand (Alpen-Adria-Universität Klagenfurt)

Overview of the GermEval 2021 Shared Task on the Identification of Toxic, Engaging, and Fact-Claiming Comments

Julian Risch[1], Anke Stoll[2], Lena Wilms[2], and Michael Wiegand[3]

[1]deepset
[1]julian.risch@deepset.ai
[2]Department of Social Sciences, Heinrich Heine University Düsseldorf
[2]anke.stoll@hhu.de, lena.wilms@hhu.de
[3]Digital Age Research Center, Alpen-Adria-Universität Klagenfurt
[3]michael.wiegand@aau.at

Abstract

We present the GermEval 2021 shared task on the identification of toxic, engaging, and fact-claiming comments. This shared task comprises three binary classification subtasks with the goal to identify: toxic comments, engaging comments, and comments that include indications of a need for fact-checking, here referred to as fact-claiming comments. Building on the two previous GermEval shared tasks on the identification of offensive language in 2018 and 2019, we extend this year's task definition to meet the demand of moderators and community managers to also highlight comments that foster respectful communication, encourage in-depth discussions, and check facts that lines of arguments rely on. The dataset comprises 4,188 posts extracted from the Facebook page of a German political talk show of a national public television broadcaster. A theoretical framework and additional reliability tests during the data annotation process ensure particularly high data quality. The shared task had 15 participating teams submitting 31 runs for the subtask on toxic comments, 25 runs for the subtask on engaging comments, and 31 for the subtask on fact-claiming comments. The shared task website can be found at https://germeval2021toxic.github.io/SharedTask/.

1 Introduction

User-generated content on the web, particularly on social media, has become a regular part of our everyday life. Given the heavy increase of such content within the last decade, the demand for approaches to classify online content automatically is more pressing than ever. Two previous GermEval shared tasks (Wiegand et al., 2018; Struß et al., 2019) mark important references for research teams from both academia and industry that develop and evaluate approaches to detect offensive language in German-language online discussions. With this year's edition of GermEval, we want participants to go beyond the identification of offensive comments. To this end, we extend the focus to two other classes of comments that are highly relevant to moderators and community managers on online discussion platforms: engaging comments, which should be considered to be highlighted and fact-claiming comments, which should be considered as a priority for fact-checking. This shift aims to bridge the gap between the theoretical view on comment classification and the practical needs of discussion moderators.

GermEval is a series of shared task evaluation campaigns that focus on natural language processing for the German language and has been held since 2014. The topics of the individual shared tasks range from named entity recognition, over lexical substitution, sentiment analysis, and hierarchical classification of blurbs to the identification of offensive language. Teams from both academia and industry are invited to develop and evaluate their approaches on datasets provided by the organizers. The shared tasks are run informally by self-organized groups of interested researchers and are endorsed by special interest groups within the German Society for Computational Linguistics (GSCL).

The remainder of this paper is structured as follows. We describe the task in Section 2 and give an overview of related work addressing the subtasks in Section 3. The dataset is described in detail in Section 4. In Section 5, we briefly comment on the evaluation we conducted, while in Section 6, we discuss the results. Section 7 concludes the paper.

2 Task Description

In this section, we detail the different subtasks of the shared task. Teams could participate either in all three subtasks or just in one or two of the following subtasks. Every team was allowed to submit at most three runs per subtask.

Subtask 1: Toxic Comment Classification. Toxic, offensive, or hateful language in social media and online discussion platforms remains a widespread and particularly pressing problem. Research in the field of communication science has shown that the occurrence of hate speech in online discussions decreases quality perceptions of participants and observers and may trigger stereotypical thinking, hateful commenting behavior or even withdrawal from the debate (Hsueh et al., 2015; Prochazka et al., 2018; Ziegele et al., 2018). While the automatic detection of toxic content is considered to be a promising approach in tackling this problem, it remains challenging and new approaches are constantly being developed. With this subtask we continue the series of previous GermEval Shared Tasks on Offensive Language Identification (Wiegand et al., 2018; Struß et al., 2019).

Subtask 2: Engaging Comment Classification. Normative approaches such as Online Deliberation Theory (Friess and Eilders, 2015) assume that rational, respectful, and reciprocal comments contribute to fostering constructive and non-violent exchange among discussants (Stroud et al., 2015). Such comments can even increase the perceived quality of the related news articles (Ziegele et al., 2018). Therefore, community managers and moderators increasingly express interest in identifying such valuable user comments, for example, to highlight them and to give them more visibility (Risch and Krestel, 2020). We refer to these comments as engaging comments. Engaging comments have been previously defined as comments that make readers join a discussion, e.g. by posting a reply or reacting with a thumbs up/thumbs down (Risch and Krestel, 2020). In this shared task, we expand the definition in favor of comments that meet communication standards of deliberative quality (Ziegele et al., 2018), namely rationality, reciprocity, and mutual respect (Gutmann and Thompson, 1998).

Subtask 3: Fact-Claiming Comment Classification. Beyond the challenge to ensure non-hostile debates, platforms and moderators are under pressure to act due to the rapid spread of misinformation and disinformation. Platforms need to review and verify information that has been posted to meet their responsibility as information providers and distributors. As a result, there is an increasing demand for systems that automatically identify comments that should be fact-checked manually. Note that this subtask is neither about the fact-checking itself nor about the identification of fake news. Instead, the identification of fact-claiming comments should be regarded as an important preprocessing step for manual fact-checking.

3 Related Work

Detection of Toxic Comments. The detection of toxicity, which may also be referred to as *offensive language* (Razavi et al., 2010), *abusive language* (Nobata et al., 2016), *hate speech* (Warner and Hirschberg, 2012), or *incivility* (Stoll et al., 2020) is currently one of the most active fields in natural language processing. For a recent overview of different approaches, we refer the reader to Schmidt and Wiegand (2017) or Fortuna and Nunes (2018), and to Vidgen and Derczynski (2020); Risch et al. (2021) for a comprehensive overview of existing datasets. There has also been a high number of different shared tasks on this topic. For English, several of these shared tasks have been organized as part of the SemEval shared task series (Zampieri et al., 2019; Basile et al., 2019; Zampieri et al., 2020; Pavlopoulos et al., 2021). For German, there have also been two editions of GermEval focusing on this task (Wiegand et al., 2018; Struß et al., 2019). The major difference between those two editions and this year's subtask on toxic comments is the data source. While the data by Wiegand et al. (2018) and Struß et al. (2019) exclusively comprise tweets, this shared task deals with Facebook posts.

Detection of Engaging Comments. The task of detecting engaging comments is motivated by the idea to highlight comments that encourage and foster reasoned and civil discussions (Ziegele et al., 2018). Napoles et al. (2017b) laid groundwork by creating an annotated dataset of engaging, respectful, and informative conversations. They identified characteristics of these conversations, such as being on-topic of the discussed news article and persuasive but not sarcastic or mean. The authors used these characteristics in their follow-up work to automatically identify these conversations (Napoles et al., 2017a). Kolhatkar and Taboada (2017) introduce another publicly available dataset and use editor picks of comments posted on the website of the New York Times as examples of constructive comments. Examples of non-constructive comments comprise a subset of comments from non-constructive threads in the dataset by Napoles et al. (2017b). While Risch and Krestel (2020) applied deep learning methods to identify engaging comments automatically, there has been no related work on transformer-based models for this task.

Detection of Fact-Claiming Comments. Detecting check-worthy factual claims recently gained increasing attention – not least because of false claims spread in the context of presidential elections or COVID-19. Hassan et al. (2017) present a semi-automated approach for fact-checking, including automated querying of a knowledge base. Only if querying the knowledge base fails and if several other criteria are met, a claim is considered check-worthy according to their approach. As a follow-up work, they released the *ClaimBuster* dataset, which can be used as a training dataset for identifying check-worthy claims (Arslan et al., 2020). Another publicly available dataset comprises claims made in political debates (Patwari et al., 2017). There is a series of shared tasks on automatic identification and verification of claims in social media, called *CLEF - CheckThat! Lab* (Nakov et al., 2018; Elsayed et al., 2019; Barrón-Cedeno et al., 2020; Nakov et al., 2021). Note that fact-checking of news articles, often referred to as fake news detection, is different from fact-checking of user comments reacting to an article. These two tasks require different approaches, such as taking into account a much longer text or the reputation of the source.

4 Data & Resources

We manually annotated a dataset of more than 4,000 Facebook user comments, which is drawn from the Facebook page of a German political talk show of a national public television broadcaster. The user comments usually revolve around the political topic discussed in a particular edition of the show and contain feedback to political standpoints, the performance of talk show guests and the TV format as a whole. The training dataset contains more than 3,000 comments that were posted in the time span from January to July 2019. To constitute a realistic use case, the test dataset includes comments on editions of the show that were aired after the period of the training dataset. It includes about 1,000 comments that were posted in the time span from September to December 2020. We deliberately decided against producing our training and test data via random sampling to avoid similar word distributions in both data sets. Further, since different people post comments to different editions of the talk show, it is unlikely that our dataset is dominated by the same person posting comments of a particular category (e.g. toxic comments) to any topic: our training data contain user comments of 157 especially active users debating in 141 discussion threads. Therefore, we consider a topic bias and person bias (Wiegand et al., 2019) unlikely. The dataset is released in anonymized form, which means that all user information and comment IDs have been removed.

For annotating our dataset, we made use of a theory-based annotation scheme, which is designed to identify fine-grained forms of toxic and engaging commentary behavior as well as fact-claiming in online discussions (Wilms et al., 2021). An overview of the resulting fine-grained subcategories used in the annotation can be found in Table 1. For the shared task, these subcategories have been subsumed to the three main categories of the subtasks (i.e. toxic, engaging and fact-claiming comments) in a second step. The publicly released dataset only contains the annotation for these three coarse-grained categories.

The dataset we release contains 4,188 Facebook comments (training data = 3,244, test data = 944), which were labeled by trained annotators. High annotation quality was ensured by intensive annotator training as well as intercoder reliability testing using Krippendorff's alpha.[1] Apart from the discussion topic and the user id of a comment, the annotators had no access to further context information. However, it must be noted, that during their annotation, the annotators gained a certain insight into the course of the discussion, which allowed them to interpret the correct meaning of ambiguous statements. Table 1 provides an extensive summary on annotation instructions, frequency distribution and intercoder reliability for both, the main categories as well as the fine-grained subcategories.

In the following, we provide a list of the fine-grained communication features that constitute each of the three main categories, i.e., toxic, engaging and fact-claiming comments. Annotators assigned a particular main category if they identified at least one underlying communication feature.

[1] Krippendorff's alpha corrects for random agreement between coders by relating the observed mean deviation to the assumed mean deviation of a random agreement (Krippendorff, 2018).

	Training Data			Test Data		
	Frequency		Intercoder Reliability	Frequency		Intercoder Reliability
	n	%	K-Alpha	n	%	K-Alpha
Subtask 1: Toxic comments	1122	34.5		504	46.2	
Screaming Implying volume by using all-caps at least twice	163	5.0	0.88	101	9.2	0.88
Vulgar language Use of obscene, foul or boorish language	190	5.8	0.73	37	3.4	0.86
Insults Swear words and derogatory statements	205	6.3	0.83	79	7.2	0.83
Sarcasm Ruthless, biting mockery	419	12.9	0.89	295	27.0	0.73
Discrimination Disparaging remarks about entire groups with sweeping condemnation	104	3.2	0.83	145	13.3	0.76
Discrediting Attempt to undermine the credibility of persons, groups or ideas, or deny their trustworthiness	360	11.0	0.83	26	2.4	-*
Accusation of lying Insinuation that ideas, plans, actions or policies are dishonest, subterfuge and misleading	136	4.1	0.84	75	6.9	0.76
Subtask 2: Engaging Comments	865	26.6		293	26.8	
Argument Statements to substantiate or refute theses	506	15.5	0.72	197	18.0	0.80
Additional information Additional information are cited as references for personal opinions	184	5.6	0.84	37	3.4	0.85
Personal experience Personal experiences or values are cited as references for personal opinions	125	3.8	0.86	25	2.3	0.69
Solution proposal Constructive solution proposals are democratic, realistic and rational in the broadest sense	89	2.7	0.88	58	5.3	0.77
Empathy Serious attempt to understand and acknowledge a perspective or emotion	31	0.9	0.86	10	0.9	0.79
Mutual Respect Giving credit or praising personality traits or accomplishments	59	1.7	0.86	24	2.2	0.85
Polite salutation Use of polite language indicated by e.g. polite salutation	30	0.9	1	11	1.0	0.90
Subtask 3: Fact-Claiming Comments	1103	34.0		353	32.3	
Assertion of facts Statements with a truth claim, which is accessible for proof	1013	31.2	0.73	343	31.4	0.82
Provision of evidence Additional information are cited as references for personal opinions	184	5.6	0.84	37	3.4	0.85
	N = 3244		n = 105 4 annotators	N = 1092		n = 123 6 annotators

Table 1: Overview of frequency distribution and reliability (Krippendorff's Alpha) of fine-grained class labels on training and test dataset. Annotation scheme was adapted from Wilms et al. (2021). Note that the test set used in the shared task is a subset of the test set listed in this table where we filtered out 148 of the samples. Thereby, we ensure a similar class distribution in the training and test set of the shared task. The size and class distribution of the downsampled test set are displayed in Table 2. *The category *Discrediting* was re-labeled in the test dataset by one person.

Please note, that a comment can be assigned to more than one main category at the same time. Figure 1 shows examples for all three classes.

Toxic Comments. Toxic comments comprise uncivil forms of communication that can violate the rules of polite behavior, such as insulting participants of a discussion, using vulgar or sarcastic language or implied volume via capital letters. Additionally, incivility can be characterized as a violation of democratic discourse values, e.g. by verbally attacking basic democratic principles or making it difficult for others to participate (Papacharissi, 2004). It includes discrimination or discreditation of participants as well as threats of violence or the accusation of lying.

Engaging Comments. Engaging comments include behavior that is in line with deliberative principles, namely rationality, reciprocity, and mutual respect (Gutmann and Thompson, 1998). The first category covers communication features, such as justification, solution proposals, or the sharing of personal experiences. The second category covers empathy with regard to other users' standpoints. The third category is present when the comment is in line with rules of polite interaction or includes the expression of mutual respect.

Fact-Claiming Comments. All comments that contain any assertion of facts are considered as fact-claiming comments. In addition, the provision of evidence by external sources that have been cited fall into the class of fact-claiming comment. Figure 1 shows example comments of each class.

Sampling for the Final Dataset For the shared task, we resampled the original test dataset as presented in Table 1 so that for all subtasks, there is a similar class distribution between the training and test dataset. This was achieved by downsampling the test set. We decided in favor of this modification to allow supervised machine-learning approaches to be effective. Table 2 shows the size and class distribution of the training and test dataset as used in this year's edition of GermEval and as publicly available via the shared task website.

"Na, welchem tech riesen hat er seine Eier verkauft..?" *TOXIC*

"Ich macht mich wütend, dass niemand den Schülerinnen Gehör schenkt" *NOT TOXIC*

(a) Subtask 1: identification of toxic comments.

"Wie wär's mit einer Kostenteilung. Schließlich haben beide Parteien (Verkäufer und Käufer) etwas von der Tätigkeit des Maklers. Gilt gleichermassen für Vermietungen. Die Kosten werden so oder so weiterverrechnet, eine Kostenreduktion ist somit nicht zu erwarten." *ENGAGING*

"Die aktuelle Situation zeigt vor allem eines: viele Kinder mussten erkennen, dass ihre Mütter bestenfalls das Niveau Grundschule, Klasse 3 haben." *NOT ENGAGING*

(b) Subtask 2: identification of engaging comments.

"Kinder werden nicht nur seltener krank, sie infizieren sich wohl auch seltener mit dem Coronavirus als ihre Eltern - das ist laut Ministerpräsident Winfried Kretschmann (Grüne) das Zwischenergebnis einer Untersuchung der Unikliniken Heidelberg, Freiburg und Tübingen." *FACT-CLAIMING*

"hmm...das kann ich jetzt nich nachvollziehen..." *NOT FACT-CLAIMING*

(c) Subtask 3: identification of fact-claiming comments.

Figure 1: Example comments and their class labels.

5 Evaluation

Following in the footsteps of the GermEval 2019 Shared on Hierarchical Classification of Blurbs (Remus et al., 2019) and the GermEval 2020 Shared Task on the Classification and Regression of Cognitive and Motivational Style (Johannßen et al., 2020), we use the platform codalab for evaluation.[2]

The evaluation uses precision, recall, and macro-average F1-score as metrics. Macro-average F1-scores give equal importance to each class, which is suited because classes in our dataset are not uniformly distributed but are equally important to identify. It is calculated as the harmonic mean of the arithmetic means of class-wise precision and recall:

$$F_1 = 2\frac{\bar{P}\bar{R}}{\bar{P}+\bar{R}} = 2\frac{(\frac{1}{n}\sum_i P_i)(\frac{1}{n}\sum_i R_i)}{\frac{1}{n}\sum_i P_i + \frac{1}{n}\sum_i R_i}$$

with P_i and R_i referring to precision and recall of class i out of n classes. We rank systems by

[2] The competition page is `https://competitions.codalab.org/competitions/32854`.

		Training Data		Test Data	
Subtask	**Class Label**	**Freq**	**%**	**Freq**	**%**
(1) toxic comments	toxic	1122	34.6	350	37.1
	not toxic	2122	65.4	594	62.9
(2) engaging comments	engaging	865	26.7	253	26.8
	not engaging	2379	73.3	691	73.2
(3) fact-claiming comments	fact-claiming	1103	34.0	314	33.3
	not fact-claiming	2141	66.0	630	66.7
total		3244	100.0	944	100.0

Table 2: Class distribution of the training and test dataset as used in the shared task.

their macro-average F1-score and do not consider accuracy in this shared task, since there is an imbalanced class distribution in each subtask. Accuracy typically rewards correct classification of the majority class. An evaluation tool computing all of the above mentioned evaluation measures is available on the website of the shared task.

6 Results

A high-level summary of the results by the participants in the different subtasks is given in Table 3. It provides summary statistics on the macro-average F1-score, which is the metric that was used as the official ranking criterion in the shared task. In comparison to subtask 1, the results of subtasks 2 and 3 are more tightly clustered suggesting that the methods pursued by the different participants are similarly effective. Overall, the best F1-scores reached in the different subtasks range from 69.98 (subtask 2) to 76.26 (subtask 3). These absolute numbers suggest that all three tasks are difficult and that there is still room for improvement.

Toxic Comments. We received 31 different runs from twelve teams for subtask 1, i.e. the detection of toxicity. The results are shown in Table 4. As a baseline, we also included the performance of a majority-class classifier always predicting the majority class, which is the absence of toxicity.

Engaging Comments. We received 25 different runs from nine teams for subtask 2, i.e. the detection of engaging comments. The results are shown in Table 5. As a baseline, we also included the performance of a majority-class classifier always predicting the majority class, which is the absence of engaging comments.

Fact-Claiming Comments. We received 31 different runs from eleven teams for subtask 3, i.e. the detection of fact-claiming comments. The results are shown in Table 6. As a baseline, we also included the performance of a majority-class classifier always predicting the majority class, which is the absence of fact-claiming comments.

General Conclusions Drawn from the Evaluation. Given that the overwhelming majority of participants followed generic classification approaches for the different subtasks, we discuss the results in this section jointly. All teams that participated in this year's shared task tested some form of deep learning. All teams except one considered contextual embeddings, most predominantly some type of transformer (i.e. BERT (Devlin et al., 2019)). Since the participants made use of various publicly available pre-trained models and given that the models of the best performing systems are different, it is difficult to determine any publicly available model that is particularly effective. Other types of classifiers, be it traditional supervised classifiers (e.g. Support Vector Machines, Logistic Regression, Forests) or other deep learning algorithms (e.g. CNN, GRU, or LSTM) were only used by a handful of teams each. Only one participant also tested a rule-based classifier.

An additional method that has already proved effective in previous editions of GermEval (Wiegand et al., 2018; Struß et al., 2019) are ensemble methods. Slightly more than half of the participants employed some form of ensemble, including virtually all top-performing systems. However, we do not see a clear pattern what type of classifiers should be combined into an ensemble, be it simply different initializations of the same classifier (i.e.

Subtask	# Teams	# Runs	Min	Max	Median	Mean	SD
(1) toxic comments	12	31	35.97	71.75	66.85	63.63	8.49
(2) engaging comments	9	25	61.43	69.98	68.72	67.70	2.14
(3) fact-claiming comments	11	31	59.70	76.26	72.55	71.84	3.94

Table 3: Summary statistics for overall macro F1-scores in the three subtasks.

Team ID	Codalab Run ID	F1	P	R
FHAC	921610	71.75	73.10	70.44
FHAC	921609	71.61	70.87	72.37
FHAC	920735	71.27	70.55	72.00
FH-SWF SG	918686	70.73	74.28	67.51
WLV-RIT	921323	69.14	73.54	65.24
WLV-RIT	921321	69.14	72.56	66.03
ur-iw-hnt	921615	68.98	71.83	66.35
DFKI SLT	921619	68.59	68.99	68.18
TUW-Inf	921590	68.42	70.44	66.52
ur-iw-hnt	921616	68.33	71.68	65.29
ur-iw-hnt	921614	68.10	70.47	65.88
WLV-RIT	921318	67.96	71.74	64.56
TUW-Inf	921582	67.71	70.06	65.51
TUW-Inf	921594	67.46	69.22	65.79
Precog-LTRC-IIITH	920506	66.87	67.42	66.33
DFKI SLT	920147	66.85	66.35	67.35
Data Science Kitchen	921663	66.85	66.98	66.73
Precog-LTRC-IIITH	920089	66.54	67.17	65.92
FH-SWF SG	921306	65.81	67.77	63.95
DFKI SLT	921621	65.73	65.90	65.56
Data Science Kitchen	921319	64.79	65.95	63.67
Data Science Kitchen	921587	63.78	64.89	62.71
Universität Regensburg MaxS	921252	61.53	62.30	60.79
DeTox	921281	58.95	63.06	55.35
IRCologne	921157	57.63	58.24	57.03
IRCologne	921667	57.40	58.03	56.77
UR@NLP_A_Team	921640	55.59	55.71	55.47
UR@NLP_A_Team	919179	55.47	55.29	55.65
UR@NLP_A_Team	921263	55.45	55.50	55.40
DeTox	921278	38.12	38.54	37.71
DeTox	921282	35.97	36.22	35.72
majority-class classifier (baseline)		38.62	31.46	50.00

Table 4: Results of subtask 1: identification of toxic comments.

transformer), different pre-trained models or the combination of a transformer with a traditional supervised classifier. While the participants applied different methods to combine all predictions of the ensembled models into a single prediction, the most frequent method was simple (soft) majority voting.

Only three teams considered using the data from previous related GermEval editions (Wiegand et al., 2018; Struß et al., 2019) as additional training data. This low number does not come as a surprise since those previous editions addressed text from a different source, i.e. Twitter rather than Facebook. Being

Team ID	Codalab Run ID	F1	P	R
Data Science Kitchen	921663	69.98	71.71	68.34
FHAC	921609	69.91	68.39	71.51
FW-SWF SG	918686	69.69	69.41	69.97
WLV-RIT	921321	69.47	68.95	69.99
WLV-RIT	921323	69.34	69.44	69.24
ur-iw-hnt	921614	69.29	72.28	66.53
WLV-RIT	921318	69.26	68.27	70.27
FW-SWF SG	921306	69.02	68.42	69.63
FHAC	920735	69.01	67.52	70.56
Precog-LTRC-IIITH	920506	68.93	68.37	69.50
UPAppliedCL	921269	68.92	70.77	67.16
ur-iw-hnt	921615	68.75	71.24	66.42
Data Science Kitchen	921319	68.72	69.70	67.78
Precog-LTRC-IIITH	920089	68.60	68.21	69.00
Data Science Kitchen	921587	68.33	69.26	67.43
ur-iw-hnt	921616	67.64	70.03	65.42
UPAppliedCL	921271	66.91	68.49	65.39
UPAppliedCL	921270	66.88	70.07	63.97
TUW-Inf	921590	66.34	78.02	57.70
TUW-Inf	921582	66.34	78.02	57.70
TUW-Inf	921594	66.34	78.02	57.70
FHAC	921610	65.80	66.68	64.95
UR@NLP_A_Team	921263	64.28	64.06	64.50
UR@NLP_A_Team	919179	63.37	62.11	64.68
UR@NLP_A_Team	921640	61.43	61.07	61.80
majority-class classifier (baseline)		42.26	36.60	50.00

Table 5: Results of subtask 2: identification of engaging comments.

out-of-domain data, the data from those previous GermEval shared tasks are unlikely to produce a notable improvement for this year's shared task.

Only two teams considered exploiting the plethora of available English training datasets for this task by following some multilingual approach. This low number, too, is in line with recent findings. Even for subtask 1, i.e. toxicity detection, for which many English datasets exist (Vidgen and Derczynski, 2020; Risch et al., 2021), Nozza (2021) recently identified reasons why multilingual approaches are highly problematic. One team also explored harnessing synthetically generated training data. However, that approach did not produce the expected outcome. Despite the similarity of many approaches pursued by the different participants of this year's edition of GermEval, the difference in performance for subtask 1 is still fairly large (Table 3). We assume that due to the complexity of those state-of-the-art learning methods and frameworks, there is still a very high number of degrees of freedom (e.g. settings of hyperparameters) that apparently plays a significant role in the overall performance of classifiers. As a basis for our analysis of the results, we asked all participants to complete a survey in which we asked about details of their submission. A summary of the survey responses is available on the shared task website.

7 Conclusion

In this paper, we described the GermEval 2021 shared task on the identification of toxic, engaging, and fact-claiming comments. For each of the three classes of comments, there was an individual subtask that defined a binary classification problems. As part of this shared task, we introduced a hand-annotated dataset of 4,188 Facebook-posts. The results for all three subtasks show that state-of-the-art classification approaches perform well and achieve macro-average F1-scores between 70%

Team ID	**Codalab Run ID**	**F1**	**P**	**R**
FHAC	921609	76.26	74.97	77.59
ur-iw-hnt	921615	76.02	77.56	74.54
ur-iw-hnt	921616	75.79	77.25	74.38
ur-iw-hnt	921614	75.43	77.91	73.10
FHAC	920735	74.82	73.52	76.16
WLV-RIT	921318	74.72	74.50	74.95
WLV-RIT	921321	74.68	75.30	74.07
AITFHSTP	921165	74.62	74.13	75.11
Precog-LTRC-IIITH	920506	73.91	73.44	74.39
WLV-RIT	921323	73.69	73.54	73.83
Precog-LTRC-IIITH	920089	73.69	73.14	74.24
UPAppliedCL	921269	73.60	74.01	73.19
FH-SWF SG	921306	73.57	73.63	73.51
FH-SWF SG	918686	73.37	72.76	74.00
AITFHSTP	921162	72.84	72.71	72.96
Data Science Kitchen	921663	72.55	73.03	72.08
Data Science Kitchen	921587	72.44	73.39	71.52
Data Science Kitchen	921319	72.34	73.25	71.44
FHAC	921610	72.28	73.75	70.88
UPAppliedCL	921270	72.21	75.78	68.96
TUW-Inf	921590	72.07	71.18	72.97
TUW-Inf	921582	72.07	71.18	72.97
UPAppliedCL	921271	71.69	73.63	69.84
HunterSpeechLab	921571	71.50	72.72	70.32
HunterSpeechLab	921569	69.91	70.97	68.89
AITFHSTP	921168	69.27	68.45	70.11
TUW-Inf	921594	68.80	82.35	59.08
HunterSpeechLab	921565	68.51	69.24	67.78
UR@NLP_A_Team	919179	63.16	62.41	63.92
UR@NLP_A_Team	921640	61.50	61.10	61.91
UR@NLP_A_Team	921263	59.70	59.15	60.26
majority-class classifier (baseline)		40.03	33.37	50.00

Table 6: Results of subtask 3: identification of fact-claiming comments.

and 76%. However, all of them should be considered far from solved. In terms of methods, we cannot determine a clear winner. All participants employed some form of transformer-based neural network. Due to the complexity of that method, there is a large number of degrees of freedom, such as hyperparameters, which need to be carefully set. They still seem to have a significant impact upon the resulting overall classification performance.

Acknowledgments

We are grateful to the large number of participants whose enthusiastic participation made GermEval 2021 a great success. We would like to thank Marc Ziegele and Dominique Heinbach for the provision of annotated data as well as their valuable insight and support during data annotation. We would also like to thank our student assistants Sebastian Joppien, Saskia Jende, Alena Palkowski, Charlotte Pape and Noah Schmitt, who performed parts of the data annotation. Parts of this research were conducted in the project “AI-supported collective-social moderation of online discussions” (KOSMO) supported by the Federal Ministry of Education and Research Grant 01IS19040C to Marc Ziegele.

References

Fatma Arslan, Naeemul Hassan, Chengkai Li, and Mark Tremayne. 2020. A benchmark dataset of check-worthy factual claims. In *Proceedings of the International Conference on Web and Social Media (ICWSM)*, pages 821–829. AAAI Press.

Alberto Barrón-Cedeno, Tamer Elsayed, Preslav Nakov, Giovanni Da San Martino, Maram Hasanain, Reem Suwaileh, Fatima Haouari, Nikolay Babulkov, Bayan Hamdan, Alex Nikolov, Shaden Shaar, and Zien Sheikh Ali. 2020. Overview of checkthat! 2020: Automatic identification and verification of claims in social media. In *Proceedings of the International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 215–236. Springer.

Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. 2019. Semeval-2019 task 5: Multilingual detection of hate speech against immigrants and women in twitter. In *Proceedings of the International Workshop on Semantic Evaluation (SemEval@NAACL)*, pages 54–63. ACL.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 4171–4186. ACL.

Tamer Elsayed, Preslav Nakov, Alberto Barrón-Cedeno, Maram Hasanain, Reem Suwaileh, Giovanni Da San Martino, and Pepa Atanasova. 2019. Overview of the clef-2019 checkthat! lab: Automatic identification and verification of claims. In *Proceedings of the International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 301–321. Springer.

Paula Fortuna and Sérgio Nunes. 2018. A Survey on Automatic Detection of Hate Speech in Text. *ACM Computing Surveys*, 51(4):85:1–85:30.

Dennis Friess and Christiane Eilders. 2015. A systematic review of online deliberation research. *Policy & Internet*, 7(3):319–339.

Amy Gutmann and Dennis F Thompson. 1998. *Democracy and disagreement*. Harvard University Press.

Naeemul Hassan, Fatma Arslan, Chengkai Li, and Mark Tremayne. 2017. Toward automated fact-checking: Detecting check-worthy factual claims by claimbuster. In *Proceedings of the Conference on Knowledge Discovery and Data Mining (KDD)*, pages 1803–1812. ACM.

Mark Hsueh, Kumar Yogeeswaran, and Sanna Malinen. 2015. "Leave your comment below": Can biased online comments influence our own prejudicial attitudes and behaviors? *Human Communication Research*, 41(4):557–576.

Dirk Johannßen, Chris Biemann, Steffen Remus, Timo Baumann, and David Scheffer. 2020. Germeval 2020 task 1 on the classification and regression of cognitive and motivational style from text: Companion paper. In *Proceedings of the Conference on Natural Language Processing (KONVENS)*, pages 1–10. CEUR-WS.org.

Varada Kolhatkar and Maite Taboada. 2017. Using new york times picks to identify constructive comments. In *Proceedings of the Natural Language Processing meets Journalism Workshop (NLPmJ@EMNLP)*, pages 100–105. ACL.

Klaus Krippendorff. 2018. *Content analysis: An introduction to its methodology*. Sage publications.

Preslav Nakov, Alberto Barrón-Cedeño, Tamer Elsayed, Reem Suwaileh, Lluís Màrquez, Wajdi Zaghouani, Pepa Atanasova, Spas Kyuchukov, and Giovanni Da San Martino. 2018. Overview of the clef-2018 checkthat! lab on automatic identification and verification of political claims. In *Proceedings of the International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 372–387. Springer.

Preslav Nakov, Giovanni Da San Martino, Tamer Elsayed, Alberto Barrón-Cedeño, Rubén Míguez, Shaden Shaar, Firoj Alam, Fatima Haouari, Maram Hasanain, Nikolay Babulkov, Alex Nikolov, Gautam Kishore Shahi, Julia Maria Struß, and Thomas Mandl. 2021. The clef-2021 checkthat! lab on detecting check-worthy claims, previously fact-checked claims, and fake news. In *Proceedings of the European Conference on Information Retrieval (ECIR)*, pages 639–649. Springer.

Courtney Napoles, Aasish Pappu, and Joel R Tetreault. 2017a. Automatically identifying good conversations online (yes, they do exist!). In *Proceedings of the International Conference on Web and Social Media (ICWSM)*, pages 628–631. AAAI Press.

Courtney Napoles, Joel Tetreault, Aasish Pappu, Enrica Rosato, and Brian Provenzale. 2017b. Finding good conversations online: The yahoo news annotated comments corpus. In *Proceedings of the Linguistic Annotation Workshop (LAW@EACL)*, pages 13–23.

Chikashi Nobata, Joel R. Tetreault, Achint Thomas, Yashar Mehdad, and Yi Chang. 2016. Abusive language detection in online user content. In *Proceedings of the International Conference on World Wide Web (WWW)*, pages 145–153. ACM.

Debora Nozza. 2021. Exposing the limits of Zero-shot Cross-lingual Hate Speech Detection. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*. ACL.

Zizi Papacharissi. 2004. Democracy online: Civility, politeness, and the democratic potential of online political discussion groups. *New media & society*, 6(2):259–283.

Ayush Patwari, Dan Goldwasser, and Saurabh Bagchi. 2017. Tathya: A multi-classifier system for detecting check-worthy statements in political debates. In *Proceedings of the Conference on Information and Knowledge Management (CIKM)*, pages 2259–2262. ACM.

John Pavlopoulos, Jeffrey Sorensen, Leo Laugier, and Ion Androutsopoulos. 2021. SemEval-2021 Task 5: Toxic Spans Detection. In *Proceedings of the International Workshop on Semantic Evaluation (SemEval@ACL-IJCNLP)*, pages 59–69. ACL.

Fabian Prochazka, Patrick Weber, and Wolfgang Schweiger. 2018. Effects of civility and reasoning in user comments on perceived journalistic quality. *Journalism studies*, 19(1):62–78.

Amir H. Razavi, Diana Inkpen, Sasha Uritsky, and Stan Matwin. 2010. Offensive language detection using multi-level classification. In *Proceedings of the Canadian Conference on Advances in Artificial Intelligence (Canadian AI)*, pages 16–27. Springer.

Steffen Remus, Rami Aly, and Chris Biemann. 2019. Germeval 2019 task 1: Hierarchical classification of blurbs. In *Proceedings of the Conference on Natural Language Processing (KONVENS)*, pages 280–292. German Society for Computational Linguistics and Language Technology (GSCL).

Julian Risch and Ralf Krestel. 2020. Top comment or flop comment? predicting and explaining user engagement in online news discussions. In *Proceedings of the International Conference on Web and Social Media (ICWSM)*, pages 579–589. AAAI Press.

Julian Risch, Philipp Schmidt, and Ralf Krestel. 2021. Data integration for toxic comment classification: Making more than 40 datasets easily accessible in one unified format. In *Proceedings of the Workshop on Online Abuse and Harms (WOAH@ACL)*, pages 157–163. ACL.

Anna Schmidt and Michael Wiegand. 2017. A survey on hate speech detection using natural language processing. In *Proceedings of the International Workshop on Natural Language Processing for Social Media (SocialNLP@EACL)*, pages 1–10. ACL.

Anke Stoll, Marc Ziegele, and Oliver Quiring. 2020. Detecting impoliteness and incivility in online discussions: Classification approaches for german user comments. *Computational Communication Research*, 2(1):109–134.

Natalie Jomini Stroud, Joshua M Scacco, Ashley Muddiman, and Alexander L Curry. 2015. Changing deliberative norms on news organizations' facebook sites. *Journal of Computer-Mediated Communication*, 20(2):188–203.

Julia Maria Struß, Melanie Siegel, Josef Ruppenhofer, Michael Wiegand, and Manfred Klenner. 2019. Overview of GermEval task 2, 2019 shared task on the identification of offensive language. In *Proceedings of the Conference on Natural Language Processing (KONVENS)*, pages 352–363. German Society for Computational Linguistics and Language Technology (GSCL).

Bertie Vidgen and Leon Derczynski. 2020. Directions in Abusive Language Training Data. *PLoS One*, 15(12).

William Warner and Julia Hirschberg. 2012. Detecting hate speech on the world wide web. In *Proceedings of the Workshop on Language in Social Media (LSM@ACL)*, pages 19–26. ACL.

Michael Wiegand, Josef Ruppenhofer, and Thomas Kleinbauer. 2019. Detection of Abusive Language: the Problem of Biased Datasets. In *Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 602–608. ACL.

Michael Wiegand, Melanie Siegel, and Josef Ruppenhofer. 2018. Overview of the GermEval 2018 shared task on the identification of offensive language. In *Proceedings of the Conference on Natural Language Processing (KONVENS)*, pages 1–10. Austrian Academy of Sciences.

Lena Wilms, Dominique Heinbach, and Marc Ziegele. 2021. Annotation guidelines for GermEval 2021 shared task on the identification of toxic, engaging, and fact-claiming comments. Excerpt of an unpublished codebook of the DEDIS research group at Heinrich-Heine-University Düsseldorf (full version available on request).

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. SemEval-2019 task 6: Identifying and categorizing offensive language in social media (OffensEval). In *Proceedings of the International Workshop on Semantic Evaluation (SemEval@NAACL)*, pages 75–86. ACL.

Marcos Zampieri, Preslav Nakov, Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Hamdy Mubarak, Leon Derczynski, Zeses Pitenis, and Çağrı Çöltekin. 2020. SemEval-2020 Task 12: Multilingual Offensive Language Identification in Social Media (OffensEval 2020). In *Proceedings of the International Workshop on Semantic Evaluation (SemEval@COLING)*, pages 1425–1447. ACL.

Marc Ziegele, Mathias Weber, Oliver Quiring, and Timo Breiner. 2018. The dynamics of online news discussions: effects of news articles and reader comments on users' involvement, willingness to participate, and the civility of their contributions. *Information, Communication & Society*, 21(10):1419–1435.

8 Appendix

Team ID	**Affiliation**	**Paper Title**
AITFHSTP	Austrian Institute of Technology GmbH/St. Pölten University of Applied Sciences	AITFHSTP at GermEval 2021: Automatic Fact Claiming Detection with Multilingual Transformer Models
Data Science Kitchen	Data Science Kitchen	Data Science Kitchen at GermEval 2021: A Fine Selection of Hand-Picked Features, Delivered Fresh from the Oven
DeTox	Darmstadt University of Applied Sciences/Fraunhofer Institute for Secure Information Technology	DeTox at GermEval 2021: Toxic Comment Classification
FHAC	FH Aachen University of Applied Sciences	FHAC at GermEval 2021: Identifying German toxic, engaging, and fact-claiming comments with ensemble learning
FH-SWF SG	Fachhochschule Südwestfalen	FH-SWF SG at GermEval 2021: Using Transformer-Based Language Models to Identify Toxic, Engaging, & Fact-Claiming Comments
HunterSpeechLab	City University of New York	HunterSpeechLab at GermEval 2021: Does Your Comment Claim A Fact? Contextualized Embeddings for German Fact-Claiming Comment Classification
IRCologne	TH Köln	IRCologne at GermEval 2021: Toxicity Classification
Precog-LRTC-IIITH	International Institute of Information Technology, Hyderabad, India	Precog-LTRC-IIITH at GermEval 2021: Ensembling Pre-Trained Language Models with Feature Engineering
DFKI SLT	DFKI GmbH	DFKI SLT at GermEval 2021: Multilingual Pre-training and Data Augmentation for the Classification of Toxicity in Social Media Comments
Universität Regensburg MaxS	Universität Regensburg	Universität Regensburg MaxS at GermEval 2021 Task 1: Toxic Comment Classification
UPAppliedCL	University of Potsdam	UPAppliedCL at GermEval 2021: Identifying Fact-Claiming and Engaging Facebook Comments Using Transformers
ur-iw-hnt	University of Regensburg	ur-iw-hnt at GermEval 2021: An Ensembling Strategy with Multiple BERT Models
UR@NLP_A_Team	University of Regensburg	UR@NLP_A_Team @ GermEval 2021: Ensemble-based Classification of Toxic, Engaging and Fact-Claiming Comments
TUW-Inf	TU Wien	TUW-Inf at GermEval2021: Rule-based and Hybrid Methods for Detecting Toxic, Engaging, and Fact-Claiming Comments
WLV-RIT	University of Wolverhampton/Rochester Institute of Technology	WLV-RIT at GermEval: Multitask Learning with Transformers to Detect Toxic, Engaging, and Fact-Claiming Comments

Table 7: Team ID, affiliation and paper title.

UPAppliedCL at GermEval 2021: Identifying Fact-Claiming and Engaging Facebook Comments Using Transformers

Robin Schaefer
Applied Computational Linguistics
University of Potsdam
Potsdam, Germany
robin.schaefer@uni-potsdam.de

Manfred Stede
Applied Computational Linguistics
University of Potsdam
Potsdam, Germany
stede@uni-potsdam.de

Abstract

In this paper we present UPAppliedCL's contribution to the GermEval 2021 Shared Task. In particular, we participated in Subtasks 2 (Engaging Comment Classification) and 3 (Fact-Claiming Comment Classification). While acceptable results can be obtained by using unigrams or linguistic features in combination with traditional machine learning models, we show that for both tasks transformer models trained on fine-tuned BERT embeddings yield best results.

1 Introduction

In the last decade social media platforms, like Facebook[1], have gained a notable momentum, which is reflected by the increasing number of users of social media.[2] While facilitating communication across the globe, from the perspective of NLP however, systems need to be specifically adapted to social media for the following reasons.

First, social media data is unedited and contains certain conventions which can pose challenges for systems trained on more well-formed texts (Šnajder, 2016). Second, social media platforms are used for different kinds of communication ranging from everyday conversations to sophisticated evidence-based argumentation on political issues. While the latter have the potential to contribute to public political discourse in general, social media has been found to contain not only respectful and engaging discussions but also hateful speech, which threatens the respectful exchange and possibly also the mental well-being of its participants. The GermEval 2021 Shared Task (Risch et al., 2021) aims to stimulate research on this issue, while also going beyond the single task of toxic comment classification.

In this paper we present UPAppliedCL's contribution[3] to the GermEval 2021 Shared Task which consists of three subtasks revolving around the mentioned characteristics of social media discussions: 1. Toxic Comment Classification; 2. Engaging Comment Classification; 3. Fact-Claiming Comment Classification. Here we especially focus on Subtask 3 (fact-claiming comments), which is also relevant for tasks in the field of argument mining (AM) (Dusmanu et al., 2017; Schaefer and Stede, 2021). In addition, we also participate in Subtask 2 (engaging comments), which we consider as a first albeit facultative step in an AM system in order to identify potential argumentative comments. As we consider Subtask 1 (toxic comments) as a task which is more independent of AM, we will not attend to it in this work.

The paper is structured as follows: in Section 2 we give a short overview of relevant previous work. We present the dataset provided by the organizers in Section 3. In Section 4 we describe our approach including the developed baselines, and in Section 5 we continue with the obtained results, which are discussed in Section 6. We conclude the paper in Section 7.

2 Related Work

Given that we do not participate in Subtask 1 (toxic comments) we will not go further into details here. For surveys on tackling this issue using NLP techniques we refer the reader to Schmidt and Wiegand (2017) and Mishra et al. (2019).

Subtask 2 (engaging comments) may be seen as a complement task to toxic comment classification as it focuses more on the identification of respectful

[1] https://www.facebook.com/

[2] https://www.statista.com/statistics/278414/number-of-worldwide-social-network-users/

[3] Code Repository: https://github.com/RobinSchaefer/GermEval2021

conversation. Approaches include work by Risch and Krestel (2020) who propose a system that relies on upvotes and replies in order to identify news comments that potentially attract user engagement. A neural network model obtained classification accuracies ranging from 0.68 to 0.72.

Subtask 3 (fact-claiming comments) can be approached from the perspective of AM, i.e., identifying fact-claiming content can be an important first step for further proving its actual correctness. Related work was published by Dusmanu et al. (2017) who investigated the classification of factual and opinionated tweets, which is defined as a pre-task for later checks of correctness, e.g., via source identification. A logistic regression model trained on a set of lexical, twitter-specific, syntactic/semantic and sentiment features yielded an F1 score of 0.80. Note, however, that no information is given whether micro or macro F1 scores are reported.

Factual information can also be used as evidence for claims. In that sense, fact-claiming comment classification can be interpreted as a pre-task for evidence detection, which has previously been investigated for different text sources including social media. For instance, in our previous work, we investigated different AM tasks including evidence detection on an expert and crowd annotated German tweet dataset. To this end we used classification and sequence labeling techniques. For evidence detection on the expert annotated dataset we obtained macro F1 scores of 0.60-0.75 for classification (XGBoost) and 0.61-0.72 for sequence labeling (CRF) (Iskender et al., 2021).

3 Data

The provided training set consists of 3244 German comments, which were collected from the Facebook page of a German political talk show. The comments were posted from February to July 2019 on two shows. All comments were anonymized. This includes replacement of user links with @USER, show links with @MEDIUM and moderator links with @MODERATOR. The comments were annotated by four trained expert annotators. For measuring inter annotator agreement (IAA) the Krippendorff's α metric was used. In total three binary annotation layers were created, each for one of the three subtasks of GermEval 2021.

Toxic Comments: Toxic comments include different types of uncivil behavior like insults, sarcastic language, discrimination, and threats of violence. It also comprises attacks on democratic principles (IAA: $0.73 < \alpha < 0.90$).

Engaging Comments: Engaging comments comprise language centering around rationality, mutual respect, empathy for others and their standpoints, and mediation (IAA: $0.71 < \alpha < 1.0$).

Fact-Claiming Comments: Fact-claiming comments focus on the assertion of facts, or evidence provided by external sources (IAA: $0.73 < \alpha < 0.84$).

For the development of our system we conducted a stratified split on the provided training set in order to obtain training, development and test sets. Both development and test set consisted of about 12.5% of the former training set. We used the development set to experiment with different feature sets and hyperparameters, while the test set was only used to calculate the preliminary test results presented in this paper.

For final system evaluation, 944 additional unlabeled comments were provided. These were drawn from discussions on a different show to avoid a topical bias.

4 System Description

In this paper we follow a machine learning (ML) approach based both on traditional ML methods and more recent deep learning (DL) techniques. We define three baselines against which we compare our submitted systems. All systems are evaluated using macro F1, precision and recall scores.

4.1 Baselines

As the first baseline (**majority**) we consider a simplistic model that outputs the most frequent class for all comments. Proportions of the most frequent class are 0.73 for Subtask 2 (non engaging) and 0.66 for Subtask 3 (non fact-claiming),[4] which indicates some imbalance in both datasets.

We define two more baselines which we had first considered for submission. However, given

[4] Importantly, these values equal the micro F1 score obtained by the first baseline model. Given that the subtasks are evaluated using macro scores, we calculate these for the baselines as well. This leads to results that diverge from the proportions but are directly comparable to the system run evaluations.

Linguistic Feature	Definition
Citation Ratio	ratio of citations
Comma Ratio	ratio of commas
First Person Ratio	ratio of 1st person pronouns
Initial Capital Ratio	ratio of tokens starting with capital
Medium Ratio	ratio of medium links
Modal Ratio	ratio of modal verbs
Moderator Ratio	ratio of moderator links
Question Ratio	ratio of question marks
Sentiment	the comment's sentiment
Text Length	the comment length
Token Length	the average token length
User Ratio	ratio of user links

Table 1: Definitions of Linguistic Features

that they cannot compete against the more sophisticated DL approaches we decided on using them for mere comparison. For baseline 2 (**unigram**) we derive unigrams from the data. We experimented with different variations of n-grams but simple unigrams perform best. During preprocessing we set all tokens to lower case and removed stopwords. Final vocabulary size is 19085. Baseline 3 (**linguistic features**) is based on a set of linguistic and text-related features which was compiled manually (see Table 1). Features for baseline 3 are partly inspired by Krüger et al. (2017). However, features *medium ratio*, *moderator ratio* and *user ratio* are based on the anonymization of the comments conducted by the organizers.

In addition to different feature sets we experimented with different classification algorithms: AdaBoost, Decision Trees (DT), eXtreme Gradient Boosting (XGBoost) (Chen and Guestrin, 2016), Gaussian Naive Bayes, Logistic Regression (LR), Support Vector Machines (SVM) and Random Forest (RF). Except for XGBoost[5] all algorithms are implemented using Scikit-Learn (Pedregosa et al., 2011). We only present results of the best systems.

4.2 Submitted Approaches

Our submitted approaches are more heavily based on DL techniques (see Table 2). All three approaches make use of pretrained German BERT document embeddings[6], which were published by *deepset.ai*. Note that the embeddings were pretrained on a set of Wikipedia texts, legal texts and news articles and not on social media data.

For submissions I and II we trained transformer models (Vaswani et al., 2017) using Flair (Akbik et al., 2019), an NLP framework which provides simple interfaces for different tasks including the creation of text embeddings and training of classification models. In addition, BERT embeddings used for submission I are fine-tuned during training, whereas for submission II the pretrained BERT embeddings are directly used for feature extraction.

Recall that the final evaluation set diverges from the training set with respect to the discussed show, i.e., the topic. To account for the possibility that during fine-tuning the BERT embeddings overfit to the training data, we decided against fine-tuning in submissions II and III.

For submission III we employed the same pretrained BERT embeddings. Instead of training a transformer model, however, we trained the same set of ML models on the encoded comments that we used for baselines 2 and 3. Our experiments revealed that XGBoost models perform best for this feature type, which is why, in the following, we will exclusively focus on this classifier. This approach is comparable to other previous work of ours, which focused on argument detection in tweets (Iskender et al., 2021; Schaefer and Stede, 2020).

We hypothesize the following ranking of submitted approaches for both subtasks:

1. Fine-tuned BERT Embeddings + Transformer
2. BERT Embeddings + Transformer
3. BERT Embeddings + XGBoost

Despite the possibility of overfitting we assume that the classifier will actually benefit from fine-tuning as the embeddings were not originally pretrained on social media data. We further hypothesize that transformers will obtain better results than traditional ML models given their success in recent years.

5 Results

Our results are based on two different datasets: 1. The test set that we obtained from our own splitting

[5] https://xgboost.readthedocs.io/en/latest/index.html

[6] https://huggingface.co/bert-base-german-cased

Submission	Features	Classifier
I	BERT Emb (FT)	Transformer
II	BERT Emb	Transformer
III	BERT Emb	XGBoost

Table 2: Submitted Approaches (Emb=Embeddings; FT=fine-tuned)

of the provided training set (henceforth **Test Set**); 2. The evaluation set we were provided with for creation of the submitted runs that were evaluated by the organizers (henceforth **Evaluation Set**). We present results obtained by both baseline and submitted models. Recall that we only participated in subtasks 2 and 3 and that all presented results are macro scores.

5.1 Test Set Results

Table 3 shows results obtained from baseline and submitted models that were applied to the test set. Due to the macro analysis the simple majority model only obtains weak results. Both the unigram baseline and the linguistic feature baseline yield substantially higher scores. Importantly, the unigram baseline performs better for both tasks than the linguistic feature baseline (Subtask 2: F1 0.728 vs 0.694; Subtask 3: F1 0.705 vs 0.704), although the better score for Subtask 3 is likely due to chance. Interestingly, precision is higher than recall.

F1 scores reveal that transformer models trained on fine-tuned BERT embeddings yield best results for both subtasks (Subtask 2: 0.775; Subtask 3: 0.790). It is noteworthy, however, that highest precision scores are obtained by the transformer models that were trained without embedding fine-tuning (Subtask 2: 0.845; Subtask 3: 0.817), while fine-tuning led to higher recall. Interestingly, an XGBoost model performs more successfully on Subtask 2 than a transformer if both are trained without fine-tuning (0.751 vs 0.737). For Subtask 3, however, the outcome was vice versa (0.754 vs 0.761). In general, scores for Subtask 3 tend to be higher than scores for Subtask 2 with the exception of precision.

5.2 Evaluation Set Results

Results obtained from finally evaluating the submitted runs are shown in Table 4. For comparison we also evaluated the unigram and linguistic feature baselines. This was possible as the organizers provided us with the labels of the evaluation set, once the deadline for the submission runs had passed. We ignore the majority baseline, as class distributions in the evaluation set are comparable to the training set.

Both baseline models show reduced F1 scores on both subtasks compared to the model outcomes from the test set. Notably, the reduction for the unigram model is larger than for the linguistic feature model. The unigram model further shows a higher recall, while the linguistic feature model benefits from a higher precision.

The first submitted system, i.e., fine-tuned BERT embeddings with transformer, yield best results (Subtask 2: 0.689; Subtask 3: 0.736), although F1 scores are again somewhat reduced compared to the test set results. Scores are higher for Subtask 3 than for Subtask 2 including precision, which contrasts with results obtained from the test set.

This pattern repeats for Submissions II (BERT embeddings (not fine-tuned) with Transformer) and III (BERT embeddings (not fine-tuned) with XGBoost classifier). Notably the XGBoost approach yields equal results in Subtask II as the transformer approach (F1: 0.669).

6 Discussion

In this section we discuss some of the results obtained by the submitted models.

As shown in Section 5 transformers trained on fine-tuned BERT embeddings yield best F1 scores, which indicates that fine-tuning does not lead to overfitting. This is the case for testing with the in-domain testing set, evaluating with the final evaluation set and for both subtasks. Further, this is in line with our ranking hypothesis.

Interestingly, however, an XGBoost model performs better on the test set of Subtask 2 than a transformer if both are trained on non-fine-tuned BERT embeddings, which contradicts our ranking hypothesis. In contrast, a transformer is more successful than an XGBoost model on Subtask 3. Model differences on the evaluation set, however, are less substantial. Evaluation F1 scores on Subtask 2 are equal. It is difficult to argue why these patterns arise. However, from these results we can carefully conclude that DL models like transformers do not necessarily outperform traditional ML models.

Furthermore, precision appears to be reduced if embeddings are fine-tuned while recall benefits

	Subtask (ST) 2			**Subtask (ST) 3**		
Approach	F1	Precision	Recall	F1	Precision	Recall
Majority	0.423	0.367	0.500	0.398	0.330	0.500
Unigram SVM (ST 2)/LR (ST 3)	**0.728**	**0.817**	**0.700**	**0.705**	**0.778**	0.691
Linguistic Features XGBoost (ST 2)/RF (ST 3)	0.694	0.729	0.678	0.704	0.728	**0.694**
BERT Emb (FT) Transformer	**0.775**	0.817	**0.752**	**0.790**	0.807	**0.780**
BERT Emb Transformer	0.737	**0.845**	0.706	0.761	**0.817**	0.742
BERT Emb XGBoost	0.751	0.818	0.724	0.754	0.796	0.738

Table 3: Test Set Results (Emb=Embeddings; FT=fine-tuned)

		Subtask (ST) 2			**Subtask (ST) 3**		
Submission	Approach	F1	Precision	Recall	F1	Precision	Recall
-	Unigram SVM (ST 2)/LR (ST 3)	**0.671**	0.665	**0.688**	0.654	0.667	**0.688**
-	Linguistic Features XGBoost (ST 2)/RF (ST 3)	0.670	**0.681**	0.664	**0.693**	**0.710**	0.685
I	BERT Emb (FT) Transformer	**0.689**	**0.708**	**0.672**	**0.736**	0.740	**0.732**
II	BERT Emb Transformer	0.669	0.701	0.640	0.722	**0.758**	0.690
III	BERT Emb XGBoost	0.669	0.685	0.654	0.717	0.736	0.698

Table 4: Evaluation Set Results (Emb=Embeddings; FT=fine-tuned)

from it. This may have interesting implications with respect to the application's focus. The results suggest that a model needing a high recall can benefit from embedding fine-tuning, while ML practitioners requiring a higher precision may refrain from fine-tuning. This finding, of course, requires more investigation before making generalisations, especially as it is less pronounced in the evaluation results.

Scores yielded for Subtask 3 tend to be higher than for Subtask 2. We argue that this might be related to the class distribution, which is more balanced in Subtask 3.

Scores obtained by evaluation are lower than by testing. This, however, is expected due to the different topics covered in training and evaluation data. Recall that the test data is topically closer related to the training set than the evaluation set. Given that we still achieved good results, especially for Subtask 3, we argue that our models are capable of solving both tasks to a promising degree.

7 Conclusion

In this paper we presented approaches to fact-claiming and engaging comment classification. We applied different combinations of features (unigrams, linguistic features, BERT embeddings) and classification algorithms including more traditional ML techniques like SVM, RF or XGBoost and more recent DL techniques like transformer models. Our experiments show that best results can be achieved by using fine-tuned BERT embeddings in combination with a transformer. We also found that fine-tuning leads to a higher recall while precision benefits from refraining from fine-tuning. As this pattern is less obvious in the evaluation set we do not argue that this finding necessarily generalizes to other datasets. However, it may be fruitful to shed more light on this in future work.

References

Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. 2019. Flair: An easy-to-use framework for state-of-the-art nlp. In *NAACL 2019, 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59.

Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 785–794, New York, NY, USA. Association for Computing Machinery.

Mihai Dusmanu, Elena Cabrio, and Serena Villata. 2017. Argument mining on Twitter: Arguments, facts and sources. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2317–2322, Copenhagen, Denmark. Association for Computational Linguistics.

Neslihan Iskender, Robin Schaefer, Tim Polzehl, and Sebastian Möller. 2021. Argument Mining in Tweets: Comparing Crowd and Expert Annotations for Automated Claim and Evidence Detection. In H. Horacek E. Métais, F. Meziane and E. Kapetanios, editors, *Natural Language Processing and Information Systems (NLDB)*, Lecture Notes in Computer Science. Springer, Cham.

Katarina Krüger, Anna Lukowiak, Jonathan Sonntag, and Manfred Stede. 2017. Classifying news versus opinions in newspapers: Linguistic features for domain independence. *Natural Language Enginnering*, 23(5):687–707.

Pushkar Mishra, Helen Yannakoudakis, and Ekaterina Shutova. 2019. Tackling online abuse: A survey of automated abuse detection methods. *CoRR*, abs/1908.06024.

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

Julian Risch and Ralf Krestel. 2020. Top comment or flop comment? predicting and explaining user engagement in online news discussions. *Proceedings of the International AAAI Conference on Web and Social Media*, 14(1):579–589.

Julian Risch, Anke Stoll, Lena Wilms, and Michael Wiegand. 2021. 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 co-located with KONVENS*, pages 1–12.

Robin Schaefer and Manfred Stede. 2020. Annotation and detection of arguments in tweets. In *Proceedings of the 7th Workshop on Argument Mining*, pages 53–58, Online. Association for Computational Linguistics.

Robin Schaefer and Manfred Stede. 2021. Argument Mining on Twitter: A survey. *it - Information Technology*, 63(1):45–58.

Anna Schmidt and Michael Wiegand. 2017. A survey on hate speech detection using natural language processing. In *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*, pages 1–10, Valencia, Spain. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, undefinedukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

Jan Šnajder. 2016. Social media argumentation mining: The quest for deliberateness in raucousness. ArXiv:1701.00168.

FH-SWF_SG at GermEval 2021: Using Transformer-Based Language Models to Identify Toxic, Engaging, & Fact-Claiming Comments

Christian Gawron
Fachhochschule Südwestfalen
Frauenstuhlweg 31
58644 Iserlohn
gawron.christian@fh-swf.de

Sebastian Schmidt
Fachhochschule Südwestfalen
Frauenstuhlweg 31
58644 Iserlohn
schmidt.sebastian2@fh-swf.de

Abstract

In this paper we describe the methods we used for our submissions to the GermEval 2021 shared task on the identification of toxic, engaging, and fact-claiming comments. For all three subtasks we fine-tuned freely available transformer-based models from the Huggingface model hub. We evaluated the performance of various pre-trained models after fine-tuning on 80% of the training data with different hyperparameters and submitted predictions of the two best performing resulting models. We found that this approach worked best for subtask 3, for which we achieved an F1-score of 0.736.

1 Introduction

Compared to the detection of offensive language in GermEval 2018 (Wiegand et al., 2019) and 2019 (Struß et al., 2019), this year's task adds two important additional categories found in social media comments, namely *engaging* and *fact-claiming* comments (Risch et al., 2021). With federal elections being held in 2021, identifying fact-claiming statements (subtask 3) in German social media posts has gained additional relevance as "fake news" might have had an influence on other important elections, e. g. the 2016 US presidential elections (Allcott and Gentzkow, 2017; Bovet and Makse, 2019). A system identifying fact-claiming comments could help to identify potential attempts to spread false factual statements.

The identification of *engaging* comments (subtask 2) is potentially interesting for the ranking algorithms used by social network providers. Increasing the visibility of these comments might help improving the attractiveness of a social network by encouraging the users to employ a more respectful and rational style of discussion.

With the classification of toxic comments (subtask 1), the GermEval Shared Tasks on the identification of offensive language mentioned above are continued. This category is also useful for the ranking algorithms of social media providers and could be used to decrease the visibility of such comments. However, we have made the experience that this year's *toxic* category is harder to identify than the former offensive categories – at least by our approach.

The best performing systems in GermEval 2019 were based on BERT (Devlin et al., 2019). Leveraging the transformer architecture (Vaswani et al., 2017) with its attention mechanism, BERT is able to model relations between words and to create semantic embeddings of sentences (Feng et al., 2020). In the last two years, various modifications of BERT like RoBERTa (Liu et al., 2019) or ELECTRA (Clark et al., 2020) have been proposed and shown to achieve state-of-the-art results on various NLP tasks. Other transformer-based models, especially GPT-2 (Radford et al., 2019) and its successor GPT-3, even made it into the press (Drösser, 2020) due to their ability to create high-quality artificial text or to create source code for various programming languages (Metz, 2020).

Probably the most important feature of these models is that they allow transfer learning: After an unsupervised *pre-training*, the resulting models can be *fine-tuned* for various NLP tasks like token classification (e. g. NER) and sequence classification. Pre-training a language model for German imposes two challenges: It requires a large corpus of text and is computationally expensive. According to Brown et al. (2020), GPT-3 was trained on a corpus of 400 billion byte-pair-encoded tokens or roughly 570 GB of text. Compared to this, the "Huge German Corpus"[1] with 204 million tokens is rather small. BERT-large was trained on 64 TPU chips for four days at an estimated cost of $7,000

[1] See https://www.ims.uni-stuttgart.de/forschung/ressourcen/korpora/hgc

(Schwartz et al., 2020), the training of GPT-3 took 3.640 petaflop-days (Brown et al., 2020). Due to the high computational effort and costs to train a model from scratch, we decided to evaluate freely available pre-trained models for our system.

For English, pre-trained models of high quality are freely available for most of the model architectures mentioned above (with the notable exception of GPT-3). Unfortunately, the groups which developed and trained these models and the companies behind them do not deem German important enough to provide pre-trained models for German. Although there is currently no active academic community in Germany training and publishing these language models, there is a growing number of companies and individuals publishing such pre-trained models. For example, Deepset.ai has published a German ELECTRA model achieving an F1-score (macro average) of 80.70% on GermEval 2018 Coarse and an F1-score (micro average) of 88.95% on GermEval 2014 (Chan et al., 2020). Philipp Reissel and Philip May have published both a German ELECTRA model (Reissel and May, 2020) and a "German colossal, cleaned Common Crawl corpus" (GC4) (Reissel and May, 2021) with about 540 GB of German text from the web It would be helpful for the development of language models for German if an extensive and high-quality corpus of German language text would be available through infrastructure projects like CLARIN-D (Hinrichs and Trippel, 2017).

2 Setup

Our experiments were performed using Jupiter Notebooks (Kluyver et al., 2016). This had the advantage that we could use local computing resources and cloud platforms like Google Colaboratory (Bisong, 2019) without modifications to the code. The code used to generate our submissions is available on GitHub[2].

We used the web application *Weights & Biases* (Biewald, 2020) to record and compare the results of experiments with different language models and hyperparameters (learning rate, number of training epochs), which was of great help especially when using cloud-based computing resources without a persistent storage medium.

3 Model Library

A large repository of pre-trained transformer based language models along with an open-source library of implementations of them is operated by Huggingface (Wolf et al., 2020). As of July 2021, the *model hub* contains about 2,900 pre-trained models for English and more than 200 pre-trained models for German provided by a fast-growing number of contributors, including the groups mentioned above. Due to the large number of available pre-trained models for German, we decided to use the Huggingface transfer library for our submission and to choose among the models available on the model hub.

The transformer library makes it very easy to use and to fine-tune the models provided on the hub. Besides the model implementations, it also contains recent optimization algorithms like AdamW (Loshchilov and Hutter, 2017) and Adafactor (Shazeer and Stern, 2018), provides integration with the experiment-tracking software *Weights & Biases* (Biewald, 2020), code for loading and handling training data, and commonly used metrics.

4 Data Preprocessing

The transformer-based language models we used for our experiments use either SentencePiece (Kudo and Richardson, 2018) or byte pair encoding (Gage, 1994) for tokenization and can handle rare words and emojis. So we did actually not preprocess the texts in any way.

One of the models we used in our experiments, `german-nlp-group/electra-base-german-uncased`, is an uncased model that converts all characters to lower case during tokenization. Unlike other 'uncased' models published on the model hub, this model does not remove accents.

5 Model Selection

With more than 200 pre-trained models for German available on the model hub, we needed to do some preselection for our experiments. Philip May, one of the authors of `german-nlp-group/electra-base-german-uncased`, has evaluated several models on the GermEval 2018 dataset (see figure 1).

We chose the best three models of this evaluation as our candidates. Due to the success of GPT-2 on various NLP tasks (Radford et al., 2019), we also included `benjamin/gerpt2-large`, a German

[2] The repository `https://github.com/fhswf/GermEval2021` will be made public after the submission of this paper.

	Sub1_Toxic			Sub2_Engaging			Sub3_FactClaiming		
Submission	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.
deepset/gelectra-large	**0.707**	0.743	0.675	**0.697**	0.694	0.700	0.734	0.728	0.740
benjamin/gerpt2-large	0.658	0.678	0.640	0.690	0.684	0.696	**0.736**	0.736	0.735

Table 1: Results of our submissions based on the models deepset/gelectra-large and benjamin/gerpt2-large.

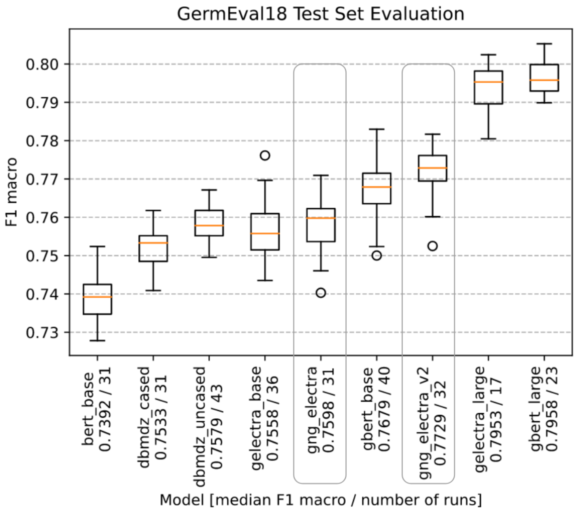


Figure 1: Results of some German language models on the GermEval 2018 dataset. Figure by Philip May, taken from the `german-nlp-group/electra-base-german-uncased` model card.

GPT-2 model recently published by Minixhofer (2020), an AI student from Johannes Kepler Universität Linz.

The following list contains some information on these models. Since we are not sure how to calculate the number of model parameters from the specification in the model configuration file, we specify the size of the binary file containing the model parameters as a measure of model complexity.

`gbert-large` has been published by Chan et al. (2020). It is a large BERT model with a binary size of 1.3 GB.

`gelectra-large` by the same group is a German ELECTRA model. The binary size is also 1.3 GB.

`electra-base-german-uncased` by Reissel and May (2020) is a smaller ELECTRA model with a binary size of 424 MB.

`gerpt2-large` published by Minixhofer (2020) is a GPT-2 model using an embedding dimension of 1280, 1024 position encodings and 20 attention heads. Although GPT-2 is mainly used for text generation, it also produces sentence embeddings which can be used for text classification. The transformer library provides the class `GPT2ForSequenceClassification` for this purpose. With a size of 3.2 GB it is the largest model we used.

6 Computing Resources

Most calculations were done on a local server using a Tesla V100S GPU card. We used fp16 precision for the training runs on the V100S for better performance as some tests with double precision did not show better results. In addition, we used cloud-based computing resources provided by GraphCore and Google Colaboratory.

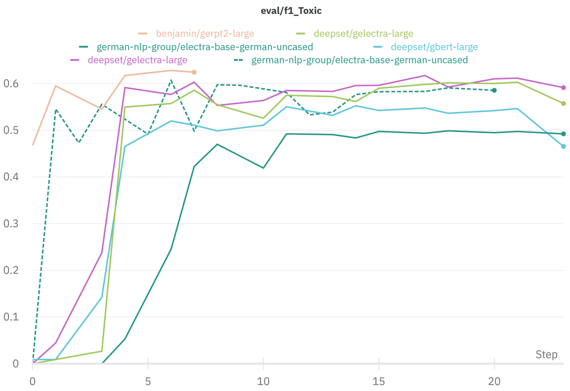


Figure 2: F1 scores of different experiments for subtask 1 with a train-test split of 0.8.

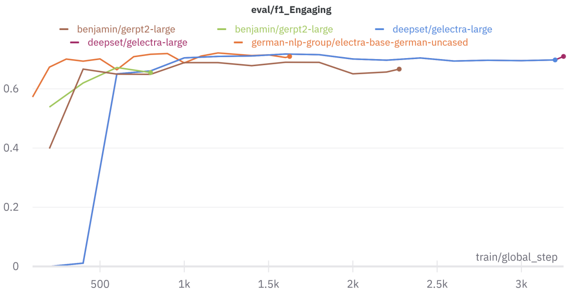


Figure 3: F1 scores of different experiments for subtask 2 with a train-test split of 0.8.

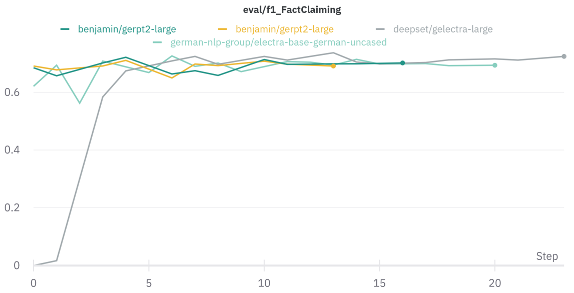


Figure 4: F1 scores of different experiments for subtask 3 with a train-test split of 0.8.

7 Results

Using the four models (see section 5) we performed several training runs with a train-test split of 80%. We did not have the time and computing resources to do a systematic hyperparameter optimization but rather tried different learning rates and number of training epochs. Figures 2 – 4 show the resulting F1-scores of several runs ans models for the three subtasks. Unfortunately, the fluctuations of the F1-scores measured on the 20% test split during the training were about as large as the differences between the different models. At this point, we would have needed more time and resources to perform a larger number of training runs and a statistical analysis similar to the one shown in figure 1. In some runs, declining F1-scores at the end of the training runs indicated overfitting – additional training data would probably have improved the results.

Overall, we achieved the best results by fine-tuning `deepset/gelectra-large` and `bjamin/gerpt2-large`. For the final system submissions, we fine-tuned these two models using the complete training dataset for all three subtasks. Table 1 shows the scores of the two submissions on the test data of the Shared Task.

8 Using Additional Training Data

Assuming that offensive language is also considered toxic, we tried to add data from GermEval 2018 and 2019 to our training dataset for subtask 1. However, compared to experiments without this additional training data, accuracy and F1-score on our validation dataset (i. e. 20% of this year's training data) were worse for these experiments. At least for an AI, toxic comments on facebook seem to be quite different from offensive language used on twitter.

9 Error Analysis

Before the gold labels were released, we compared our model predictions with our personal predictions for the first test comments. When we looked at the gold labels, we were surprised by some of the labels, especially with respect to examples having more than one label.

For example, our system flagged a fact claim in comment 3246

> @USER , ich glaube,Sie verkrnnen gründlich die Situation. Deutschland mischt sich nicht ein, weil die letzte Einmischung in der Ukraine noch nicht bereinigt ist. Es geht nicht ums Militär

which we considered correct. We did not expect that this comment is also considered engaging.

In the case of comment 3248

> Als jemand, der im real existierenden Sozialismus aufgewachsen ist, kann ich über George Weineberg nur sagen, dass er ein Voll...t ist. Finde es schon gut, dass der eingeladen wurde. Hat gezeigt, dass er viel Meinung hat, aber offensichtlich wenig Ahnung. Er hat sich eben so gut wie er kann, für alle sichtbar, zum Trottel gemacht.

we agreed with our system that the second sentence ("I think it's good that he was invited") could be considered engaging, but according to the gold labels, this comment is only toxic. On the other hand, comment 3269

> Sry aber Preetz hat nicht viel beizutragen. Er MUSS der Politik in den Hintern kriechen damit sein Verein Zuschauer ins Stadion bekommt. Er ist abhängig von der Politik.

is both toxic and engaging according to the gold labels, while we agreed with our system that this is only toxic.

These three examples demonstrate that this year's task is really hard – even for humans. It would be interesting to measure the score of human annotators getting just the category names and the training examples.

10 Conclusion

When we first looked at the development data, our impression was that fact-claiming statements would be the hardest category to recognize for an NLP system due to the wide range of different facts in the statements. The rather low range of annotator agreement of $0.73 < \alpha < 0.84$ for subtask 3 also suggests that this should be the "hard" category. We were quite surprised that our system actually achieved the best F1-score (0.736 in the case of `benjamin/gerpt2-large`) for this category.

Regarding the toxic category, the F1-score of 0.707 on subtask 1 is surprisingly low considering the F1-score of `deepset/gelectra-large` of about 0.80 reported by Chan et al. (2020) on GermEval 2018 (coarse). This year's 'toxic' category seems to be quite different from the offensive language category of the GermEval tasks in 2018 and 2019 and – at least for an AI – more difficult to recognize.

The approach we used to create our submissions is a rather simple one that did not require preprocessing of the training data or much programming. Free libraries containing implementations of a wide range of language models and the availability of an increasing number of pre-trained model instances make it quite easy to apply state-of-the-art language models for NLP tasks like text classification. It still, however, requires some coding to train and select models and to create predictions for the test dataset. Integrated tools like the recently announced AutoNLP[3] will probably enable non-experts (and non-coders) to train such models in the next few years.

Acknowledgments

This research was supported by grants from NVIDIA and utilized NVIDIA CUDA on Tesla & Ampere GPUs. This research also used free computing resources provided by the GraphCore Academic Program and Google Colab.

References

Hunt Allcott and Matthew Gentzkow. 2017. Social media and fake news in the 2016 election. *Journal of Economic Perspectives*, 31(2):211–36.

Lukas Biewald. 2020. Experiment tracking with weights and biases. Company website. Application available on wandb.com, Last accessed on 2021-07-12.

Ekaba Bisong. 2019. Google colaboratory. In *Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners*, pages 59–64. Apress, Berkeley, CA.

Alexandre Bovet and Hernán A. Makse. 2019. Influence of fake news in twitter during the 2016 us presidential election. *Nature Communications*, 10(1):7.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Branden Chan, Stefan Schweter, and Timo Möller. 2020. German's next language model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6788–6796, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: pre-training text encoders as discriminators rather than generators. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Christoph Drösser. 2020. Sie klingt wie wir. Eine Software vermittelt die Illusion eines Zwiegesprächs. *Die Zeit*, 54/2020.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2020. Language-agnostic BERT sentence embedding. *CoRR*, abs/2007.01852.

Philip Gage. 1994. A new algorithm for data compression. *C Users J.*, 12(2):23–38.

Erhard Hinrichs and Thorsten Trippel. 2017. CLARIN-D: eine Forschungsinfrastruktur für die sprachbasierte Forschung in den Geistes- und Sozialwissenschaften. *Bibliothek Forschung und Praxis*, 41(1):45–54.

[3] See `https://huggingface.co/autonlp`

Thomas Kluyver, Benjamin Ragan-Kelley, Fernando Pérez, Brian Granger, Matthias Bussonnier, Jonathan Frederic, Kyle Kelley, Jessica Hamrick, Jason Grout, Sylvain Corlay, Paul Ivanov, Damián Avila, Safia Abdalla, and Carol Willing. 2016. Jupyter notebooks – a publishing format for reproducible computational workflows. In *Positioning and Power in Academic Publishing: Players, Agents and Agendas*, pages 87–90. IOS Press.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018: System Demonstrations, Brussels, Belgium, October 31 - November 4, 2018*, pages 66–71. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, M. Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *ArXiv*, abs/1907.11692.

Ilya Loshchilov and Frank Hutter. 2017. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101.

Cade Metz. 2020. Meet GPT-3. It has learned to code (and blog and argue). *New York Times*.

Benjamin Minixhofer. 2020. GerPT2-large – A large German GPT2. Huggingface model hub. `https://huggingface.co/benjamin/gerpt2-large`, Last accessed on 2021-07-12.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*.

Philipp Reissel and Philip May. 2020. German Electra Uncased. Huggingface model hub. `https://huggingface.co/german-nlp-group/electra-base-german-uncased`, Last accessed on 2021-07-12.

Philipp Reissel and Philip May. 2021. GC4 Corpus. GitHub pages. `https://german-nlp-group.github.io/projects/gc4-corpus.html`, Last accessed on 2021-07-12.

Julian Risch, Anke Stoll, Lena Wilms, and Michael Wiegand. 2021. 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 co-located with KONVENS*, pages 1–12.

Roy Schwartz, Jesse Dodge, Noah A. Smith, and Oren Etzioni. 2020. Green ai. *Commun. ACM*, 63(12):54–63.

Noam Shazeer and Mitchell Stern. 2018. Adafactor: Adaptive learning rates with sublinear memory cost. *CoRR*, abs/1804.04235.

Julia Maria Struß, Melanie Siegel, Josef Ruppenhofer, Michael Wiegand, and Manfred Klenner. 2019. Overview of germeval task 2, 2019 shared task on the identification of offensive language. In *Preliminary proceedings of the 15th Conference on Natural Language Processing (KONVENS 2019), October 9 11, 2019 at Friedrich-Alexander-Universität Erlangen-Nürnberg*, pages 352–363, München [u.a.]. German Society for Computational Linguistics & Language Technology und Friedrich-Alexander-Universität Erlangen-Nürnberg.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Michael Wiegand, Melanie Siegel, and Josef Ruppenhofer. 2019. Overview of the germeval 2018 shared task on the identification of offensive language. In *Proceedings of GermEval 2018, 14th Conference on Natural Language Processing (KONVENS 2018), Vienna, Austria September 21, 2018*, pages 1–10. Austrian Academy of Sciences, Vienna, Austria.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rmi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

DFKI SLT at GermEval 2021: Multilingual Pre-training and Data Augmentation for the Classification of Toxicity in Social Media Comments

Remi Calizzano
DFKI GmbH
Berlin, Germany
remi.calizzano@dfki.de

Malte Ostendorff
DFKI GmbH
Berlin, Germany
malte.ostendorff@dfki.de

Georg Rehm
DFKI GmbH
Berlin, Germany
georg.rehm@dfki.de

Abstract

We present our submission to the first subtask of GermEval 2021 (classification of German Facebook comments as toxic or not). Binary sequence classification is a standard NLP task with known state-of-the-art methods. Therefore, we focus on data preparation by using two different techniques: task-specific pre-training and data augmentation. First, we pre-train multilingual transformers (XLM-RoBERTa and MT5) on 12 hatespeech detection datasets in nine different languages. In terms of F1, we notice an improvement of 10% on average, using task-specific pre-training. Second, we perform data augmentation by labelling unlabelled comments, taken from Facebook, to increase the size of the training dataset by 79%. Models trained on the augmented training dataset obtain on average +0.0282 (+5%) F1 score compared to models trained on the original training dataset. Finally, the combination of the two techniques allows us to obtain an F1 score of 0.6899 with XLM-RoBERTa and 0.6859 with MT5. The code of the project is available at: https://github.com/airKlizz/germeval2021toxic.

1 Introduction

Toxicity classification, or, more generally, hatespeech detection, has become a highly important topic due to the explosion of social media use. The automation of this task is a challenge for the NLP field with an increasing amount of research on this subject (Schneider et al., 2018; Aluru et al., 2020; Corazza et al., 2020). The GermEval series has already looked into various aspects related to the detection of German language hatespeech with two shared tasks on offensive language identification (Wiegand et al., 2018; Struß et al., 2019). The first subtask of GermEval 2021 follows in these footsteps with the classification of toxic comments.

We want to take advantage of the proliferation of hatespeech datasets for various languages created in the last couple of years. Additionally, in the meantime, a number of multilingual language models have been published (Conneau et al., 2020; Xue et al., 2021; Liu et al., 2020; Lewis et al., 2020) with a high capacity for cross-lingual transfer. We use multilingual models and pre-train them on a multilingual dataset created out of 12 datasets for nine different languages on toxicity and hatespeech detection. We evaluate whether performing this type of pre-training on multilingual models can improve their performance. We assume that the cross-lingual transfer capacity of the multilingual models can be applied to task-specific pre-training and that this will improve final performance on the German-only dataset of the shared task.

Furthermore, we perform data augmentation by labelling unlabeled data, retrieved from Facebook, using one of the multilingual models pre-trained and fine-tuned on the toxicity classification task. As the dataset of the shared task contains only 3244 examples, we hope that extending the number of training examples can improve the overall performance of the models.

In summary, our main contributions are:

- Comparison of the performance of two multilingual models (XLM-RoBERTa and mT5) against a German-specific language model (GBERT) on a German binary classification task with and without task-specific pre-training for multilingual models.

- Evaluation of the models when using data augmentation to increase the size of the dataset used for fine-tuning.

The rest of this article is structured as follows. Section 2 presents our methodology for task-specific pre-training and data augmentation. Section 3 introduces the task as well as the dataset and describes the models and training scenarios. Sections 4 and 5 present and discuss the results obtained in these training scenarios. Concluding remarks are provided in Section 6.

2 Methodology

2.1 Task-specific pre-training

Toxicity or, more generally, hatespeech classification is an NLP task that is supported through multiple datasets in multiple languages. Although the specific task may differ from one dataset to another due to the type of content and annotations used (Bourgonje et al., 2018), the features used to classify sequences are similar.

Pre-training is a technique that often enables state-of-the-art performance in many NLP tasks (Sarlin et al., 2020). Task-specific pre-training has shown its efficiency to produce models that capture task-specific features and that, thus, exhibit better performance (Li et al., 2020).

We want to profit from the many existing hatespeech classification datasets by using these datasets to perform task-specific pre-training.

We adapt task-specific pre-training to toxicity classification by taking 12 toxicity or hatespeech classification datasets and training language models on these datasets before fine-tuning them on the dataset of the shared task (Table 1). Our task-specific pre-training dataset is composed of a total of 105,142 examples in nine different languages.

To take advantage of this task-specific multilingual pre-training, we work with multilingual models. Indeed, these models have already demonstrated their ability to transfer what they have learned in one language into other languages (Hu et al., 2020). In this work, the models will be fine-tuned on the dataset of the shared task which is in German only, however, we assume that the multilingual models can benefit from the task-specific pre-training.

2.2 Data augmentation

In addition to the task specific pre-training, we increase the size of the shared task dataset using data labelling. We use our best performing model and fine-tune on the toxicity classification task of the shared task to label unlabelled Facebook comments we collected from German political talk shows. In total, we collected 5563 Facebook comments added to posts from the pages of ZDF heute[1], Panorama[2], Maischberger[3], and hart aber fair[4]. mT5 is performing better than XLM-RoBERTa on the final toxic classification task when simply using task-specific pre-training and fine-tuning, therefore we use mT5 to compute the probability of a comment to be toxic or not. We only keep the comments classified as toxic or non-toxic with a probability larger than 0.8. Figure 1 shows examples of comments with their toxicity probabilities. This way we label 2044 comments, which we add to the original shared task dataset. Table 2 compares the original dataset with the one we created and also with the augmented dataset which corresponds to the combination of the original dataset and the one we created using data augmentation.

Dataset	Number of examples	Languages
Chung et al. (2019)	7,659	eng, fra, ita
Gao and Huang (2017)	1,528	eng
Wiegand et al. (2018)	5,009	deu
Mandl et al. (2019)	14,336	eng, deu, hin
Ousidhoum et al. (2019)	13,014	ara, eng, fra
de Gibert et al. (2018)	10,944	eng
Davidson et al. (2017)	24,783	eng
Alfina et al. (2017)	713	ind
Ross et al. (2016)	469	deu
Mulki et al. (2019)	5846	apc
Nascimento et al. (2019)	7,672	por
Ibrohim and Budi (2019)	13,169	ind

Table 1: List of all the datasets used for the task-specific pre-training with the number of examples and the languages (code ISO 639-3) for each dataset.

3 Experiments

3.1 Task and dataset

The first subtask of GermEval 2021 is the classification of Facebook comments from German political talk shows with regard to their toxicity. Figure 2 shows two examples. Risch et al. (2021) provide a detailed description of the dataset.

We split the original dataset into a train and an evaluation portion to be able to evaluate our models during training. We use 80% of the original dataset for training and 20% for the evaluation, for which we use precision, recall, and macro-average F1.

[1] https://www.facebook.com/ZDFheute/
[2] https://www.facebook.com/panorama.de
[3] https://www.facebook.com/maischberger
[4] https://www.facebook.com/hartaberfairARD

Comment	**Toxicity probability**
Hat vermutlich auch überhaupt nichts mit Merkels Desaströser Politik zu tun	0.8790
Frage: Wenn die Tage kürzer werden, das Gehalt aber gleich bleibt, reicht es dann länger?	0.0541
Die Hausärzte bekommen Astra nicht verimpft und die Impfzentren bleiben halb leer. Impfturbo?	0.5627
Na was sind die Bürger erst enttäuscht von euch allen samt dem Gremium....	0.6742

Figure 1: Samples of comments collected on Facebook posts from German political talk shows with their toxicity probability. We only keep the comments classified as toxic or non-toxic with a probability larger than 0.8

	Number of examples		**Toxic label**	**Number of words per comment**		
	train	evaluation	**ratio**	mean	30^{th} pctl	70^{th} pctl
Original GermEval 2021 dataset	2,596	648	0.35	28	11	30
Created dataset	2,044	0	0.49	36	17	39
Augmented dataset	4,640	648	0.40	31	13	34

Table 2: Comparison of the original shared task dataset, the dataset created using data augmentation, and the augmented dataset, i. e., the combination of the other two datasets.

3.2 Models

The task-specific pre-training is based on a multilingual dataset (Section 2.1). We picked two multilingual Transformer models, XLM-RoBERTa and mT5. In addition, we compare multilingual models with the German Transformer based language model GBERT that we evaluate with our data augmentation method.

GBERT GBERT (Chan et al., 2020) is a German language model using the same architecture as BERT (Devlin et al., 2019). GBERT is an encoder-only Transformer model. It was trained using masked language modeling with whole word masking which corresponds to masking all of the tokens corresponding to a word. The pre-training corpus consists of German texts from Wikipedia, Common Crawl (Ortiz Suárez et al., 2019), OPUS (Tiedemann, 2012), and Open Legal Data (Ostendorff et al., 2020). GBERT outperforms the state-of-the-art for the GermEval 2018 hatespeech detection task and the GermEval 2014 NER task (Chan et al., 2020). We use the GBERT Base version.

XLM-RoBERTa XLM-RoBERTa (Conneau et al., 2020) is the multilingual version of RoBERTa (Liu et al., 2019). It was trained on the Common Crawl corpus in 100 languages using masked language modeling. We choose XLM-RoBERTa instead of Multilingual BERT[5] because XLM-RoBERTa outperforms Multilingual BERT on a variety of cross-lingual benchmarks (Conneau et al., 2020). We use the Base version of XLM-RoBERTa.

mT5 mT5 (Xue et al., 2021) is a multilingual variant of T5 (Raffel et al., 2020) covering 101 languages. It uses the same architecture as T5, an encoder-decoder Transformer model. Being a text-to-text model, we transform the binary classification task into a text generation task where we train mT5 to generate "neutral" when the input label corresponds to a non-toxic comment and "toxic" when the input label is toxic. We also add the task prefix "speech review" at the beginning of each input sequence. As T5, mT5 exists in five sizes: Small, Base, Large, XL, XXL. The XXL version of mT5 performs better than other multilingual models such as XLM-RoBERTa on many multilingual benchmarks, however, due to computational limits, we use the mT5 Base version that produces results comparable to XLM-RoBERTa (Xue et al., 2021).

3.3 Training scenarios

To evaluate the benefit of the task-specific pre-training and data augmentation, we train the models in four different scenarios.

Fine-tuning only We first fine-tune the three models on the original dataset of the shared task. These models are used as baselines to evaluate the two methodologies we propose.

With task-specific pre-training In this scenario, we pre-train mT5 and XLM-RoBERTa on the task-specific pre-training dataset (Section 2.1). The task-specific pre-training consists of training the models with the same objective as the fine-tuning task

[5] https://github.com/google-research/bert/blob/master/multilingual.md

Comment	Toxicity
Die SPD, Verbrecher,die haben Angst vor den Wahlen in den neuen Bundesländern,weg mit Euch.	1
Ich schmeiß mich weg... 800 Euro sollen für ein ””vernünftiges”” Leben ausreichen?	0

Figure 2: Two comments from the original GermEval21 shared task dataset with their toxicity labels.

which is the classification of toxic comments. As the result of the combination of those datasets is not balanced, we randomly remove non-toxic samples to arrive at the same number of toxic and non-toxic samples. Afterwards, we fine-tune the task-specific pre-trained models as in the first scenario.

With data augmentation This scenario corresponds to the first one except we use the augmented dataset instead of the original shared task dataset. The augmented dataset combines the original and one additional dataset (Table 2).

With task-specific pre-training and data augmentation This scenario combines the second and third scenario. We fine-tune the task-specific pre-trained models on the augmented dataset.

We use the HuggingFace Transformers library (Wolf et al., 2020) to train the models. GBERT and XLM-RoBERTa are trained using the hyperparameter search method[6] with Optuna as the optimization framework[7], the maximization of the F1 metric as computing objective, and a number of trials equals to 10. As mT5 requires more training time, we do not use hyperparameter search for mT5 but fixed parameters that we found to be the best. We use a learning rate of 5^{-5}, a batch size of 16, and we train mT5 for 3 epochs. In the end we select the best model with regard to the F1 score.

To deal with the imbalanced training dataset, we use class weights for GBERT and XLM-RoBERTa and oversample the dataset for mT5.

4 Results

We evaluate the models on the test dataset provided by the organizers of the shared task after the training phase and the submissions (see Table 3).

First, adding task-specific pre-training and/or using data augmentation improves the results for both XLM-RoBERTa and mT5. Training with task-specific pre-training and data augmentation improves the F1 score by 0.0490 (+8%) for XLM-RoBERTa and by 0.0836 (+14%) for mT5. GBERT also produces slightly better results, the F1 score improves by 0.0066 (+1%), when using the augmented dataset for fine-tuning.

Model	F1	Precision	Recall
Fine-tuning only			
GBERT	0.6663	0.6437	0.6906
XLM-RoBERTa	0.6409	0.6373	0.6445
mT5	0.6023	0.5995	0.6052
With task-specific pre-training			
XLM-RoBERTa	0.6785	0.6851	0.6720
mT5	0.6799	0.6840	0.6759
With data augmentation			
GBERT*	0.6729	0.6724	0.6734
XLM-RoBERTa	0.6680	0.6720	0.6639
mT5	0.6533	0.6541	0.6526
With task-specific pre-training and data augmentation			
XLM-RoBERTa*	**0.6899**	**0.6900**	**0.6898**
mT5*	**0.6859**	**0.6899**	**0.6818**

Table 3: F1, recall and precision results of each model on the test dataset of the shared task for each training scenario. * *models used for our submissions. Results slightly differ from the submissions because we retrained all the models for the paper.*

Second, for the models fine-tuned only on the original dataset, mT5 obtains the worst results with an F1 score of 0.6023, followed by XLM-RoBERTa with 0.6409, and GBERT with 0.6663. The ranking is the same for the models fine-tuned on the augmented dataset but with a smaller gap between scores. F1 scores for mT5, XLM-RoBERTa and GBERT are 0.6533, 0.6680 and 0.6729.

Third, despite mT5 performing worse that XLM-RoBERTa by 0.0386 when fine-tuned on the original dataset, the results with task-specific pre-training and data augmentation of the two models are very similar with a difference between F1 scores lower than 0.1%. This correlates with the fact that the task-specific pre-training particularly improves the results of mT5 with an increase of 0.0776 (+13%) of the F1 score compared to an increase of 0.0376 (+6%) for XLM-RoBERTa.

Overall, XLM-RoBERTa and mT5 with task-specific pre-training and data augmentation are the

[6] https://huggingface.co/transformers/main_classes/trainer.html#transformers.Trainer.hyperparameter_search

[7] https://optuna.org

models that obtain the best F1 scores with 0.6899 and 0.6859, respectively.

5 Discussion

In the two scenarios where only German data is used (*Fine-tuning only* and *With data augmentation*), GBERT performs better than XLM-RoBERTa and mT5. This is easily explained by the fact that GBERT was pre-trained only on German data, in contrast to mT5 and XLM-RoBERTa. However, the small difference in F1 scores with the use of the augmented dataset (*With data augmentation*) implies that with more data, multilingual models can perform as well as monolingual models. Additionally, we see that the task-specific pre-training of multilingual models on a multilingual dataset compensates for the poorer performance of mT5 and XLM-RoBERTa when trained on a German only dataset compared to GBERT. It is interesting to note that the task-specific pre-training of mT5 and XLM-RoBERTa on a multilingual dataset allows them to perform better than GBERT. The fact that multilingual models can benefit from hate-speech classification datasets in other languages allows them to perform better than the German-only model. It is also important to notice that XLM-RoBERTa and mT5 use more recent architectures and/or pre-training methods than GBERT. It may also partly explain that GBERT's results are worse than those of XLM-RoBERTa and mT5.

Moreover, as noted in Section 4, XLM-RoBERTa does not benefit from the task-specific pre-training as much as mT5. Our hypothesis is that having less trainable parameters, XLM-RoBERTa (270M parameters) does not have as much capacity as mT5 (580M parameters) to benefit from all the examples on which the models are pre-trained. The number of parameters of the models is an important aspect to take into consideration when doing pre-training in general, and we observe this again in our experiments with task-specific pre-training.

6 Conclusion

We describe the methods used for our submissions to the GermEval 2021 toxic comment classification task. Specifically, we can benefit from hatespeech detection datasets in other languages to improve the performance of multilingual models through task-specific pre-training. With this method, multilingual models (XLM-RoBERTa and mT5) perform even better, +0.0576 (+10%) in average in terms of F1, than GBERT, a German-specific language model. We show that by increasing the shared task dataset by automatically labeling additional comments from Facebook, we are able to improve the results of the three models we evaluated (GBERT, XLM-RoBERTa, mT5) by 5% in average.

We have shown that multilingual models can perform as well or even better than monolingual models by performing task-specific multilingual pre-training. This particularly applies to tasks for which many datasets are available in languages different from the dataset used for fine-tuning and where the fine-tuning dataset is relatively small (less than 10,000 samples) as is the case of the German toxic comment classification task.

In addition, multilingual models have some other advantages. First, in a production setting, it might not be feasible to deploy multiple monolingual models due to resource constraints. Replacing multiple monolingual models with a single multilingual model can be a solution. Second, multilingual models, due to their cross-lingual transfer capacity, can be used in a language other than the language of the training dataset. This allows the creation of models for languages for which obtaining training data can be difficult.

Acknowledgments

The research presented in this paper is funded by the German Federal Ministry of Education and Research (BMBF) through the project QURATOR (http://qurator.ai) (Unternehmen Region, Wachstumskern, grant no. 03WKDA1A). In addition, the authors would like to thank Melina Plakidis for her contribution regarding the data labelling.

References

Ika Alfina, Rio Mulia, Mohamad Ivan Fanany, and Yudo Ekanata. 2017. Hate speech detection in the indonesian language: A dataset and preliminary study. In *2017 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, pages 233–238. IEEE.

Sai Saket Aluru, Binny Mathew, Punyajoy Saha, and Animesh Mukherjee. 2020. Deep learning models for multilingual hate speech detection. *ArXiv*, abs/2004.06465.

Peter Bourgonje, Julian Moreno Schneider, and Georg Rehm. 2018. Automatic Classification of Abusive Language and Personal Attacks in Various Forms of Online Communication. In *Language Technologies* *for the Challenges of the Digital Age: 27th International Conference, GSCL 2017, Berlin, Germany, September 13-14, 2017, Proceedings*, number 10713 in Lecture Notes in Artificial Intelligence (LNAI), pages 180–191, Cham, Switzerland. Gesellschaft fur Sprachtechnologie und Computerlinguistik e.V., Springer. 13/14 September 2017.

Branden Chan, Stefan Schweter, and Timo Möller. 2020. German's next language model.

Yi-Ling Chung, Elizaveta Kuzmenko, Serra Sinem Tekiroglu, and Marco Guerini. 2019. CONAN - COunter NArratives through nichesourcing: a multilingual dataset of responses to fight online hate speech. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2819–2829, Florence, Italy. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *ACL*.

Michele Corazza, Stefano Menini, Elena Cabrio, Sara Tonelli, and Serena Villata. 2020. A multilingual evaluation for online hate speech detection. *ACM Trans. Internet Technol.*, 20(2).

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *Proceedings of the 11th International AAAI Conference on Web and Social Media*, ICWSM '17, pages 512–515.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Lei Gao and Ruihong Huang. 2017. Detecting online hate speech using context aware models. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 260–266, Varna, Bulgaria. INCOMA Ltd.

Ona de Gibert, Naiara Perez, Aitor García-Pablos, and Montse Cuadros. 2018. Hate Speech Dataset from a White Supremacy Forum. In *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*, pages 11–20, Brussels, Belgium. Association for Computational Linguistics.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalization. *CoRR*, abs/2003.11080.

Muhammad Okky Ibrohim and Indra Budi. 2019. Multi-label hate speech and abusive language detection in Indonesian twitter. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 46–57, Florence, Italy. Association for Computational Linguistics.

Mike Lewis, Marjan Ghazvininejad, Gargi Ghosh, Armen Aghajanyan, Sida Wang, and Luke Zettlemoyer. 2020. Pre-training via paraphrasing.

Junlong Li, Zhuosheng Zhang, Hai Zhao, Xi Zhou, and Xiang Zhou. 2020. Task-specific objectives of pre-trained language models for dialogue adaptation. *ArXiv*, abs/2009.04984.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, M. Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pretraining for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv e-prints*, page arXiv:1907.11692.

Thomas Mandl, Sandip Modha, Prasenjit Majumder, Daksh Patel, Mohana Dave, Chintak Mandlia, and Aditya Patel. 2019. Overview of the hasoc track at fire 2019: Hate speech and offensive content identification in indo-european languages. In *Proceedings of the 11th Forum for Information Retrieval Evaluation*, FIRE '19, page 14–17, New York, NY, USA. Association for Computing Machinery.

Hala Mulki, Hatem Haddad, Chedi Bechikh Ali, and Halima Alshabani. 2019. L-hsab: A levantine twitter dataset for hate speech and abusive language. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 111–118.

Gabriel Nascimento, Flavio Carvalho, Alexandre Martins da Cunha, Carlos Roberto Viana, and Gustavo Paiva Guedes. 2019. Hate speech detection using brazilian imageboards. In *Proceedings of the 25th Brazillian Symposium on Multimedia and the Web*, WebMedia '19, page 325–328, New York, NY, USA. Association for Computing Machinery.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous Pipeline for Processing Huge Corpora on Medium to Low Resource Infrastructures. In *7th Workshop on the Challenges in the Management of Large Corpora (CMLC-7)*, Cardiff, United Kingdom. Leibniz-Institut für Deutsche Sprache.

Malte Ostendorff, Till Blume, and Saskia Ostendorff. 2020. Towards an open platform for legal information. In *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2020*, JCDL '20, page 385–388, New York, NY, USA. Association for Computing Machinery.

Nedjma Ousidhoum, Zizheng Lin, Hongming Zhang, Yangqiu Song, and Dit-Yan Yeung. 2019. Multilingual and multi-aspect hate speech analysis. In *Proceedings of EMNLP*. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Julian Risch, Anke Stoll, Lena Wilms, and Michael Wiegand. 2021. 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 co-located with KONVENS*, pages 1–12.

Björn Ross, Michael Rist, Guillermo Carbonell, Benjamin Cabrera, Nils Kurowsky, and Michael Wojatzki. 2016. Measuring the Reliability of Hate Speech Annotations: The Case of the European Refugee Crisis. In *Proceedings of NLP4CMC III: 3rd Workshop on Natural Language Processing for Computer-Mediated Communication*, volume 17 of *Bochumer Linguistische Arbeitsberichte*, pages 6–9, Bochum.

Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. 2020. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4938–4947.

Julian Moreno Schneider, Roland Roller, Peter Bourgonje, Stefanie Hegele, and Georg Rehm. 2018. Towards the Automatic Classification of Offensive Language and Related Phenomena in German Tweets. In *Proceedings of the GermEval Workshop 2018 – Shared Task on the Identification of Offensive Language*, pages 95–103, Vienna, Austria. 21 September 2018.

Julia Struß, Melanie Siegel, Josef Ruppenhofer, Michael Wiegand, and Manfred Klenner. 2019. Overview of germeval task 2, 2019 shared task on the identification of offensive language. In *Zurich Open Repository and Archive*.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pages 2214–2218, Istanbul, Turkey. European Languages Resources Association (ELRA).

Michael Wiegand, Melanie Siegel, and Josef Ruppenhofer. 2018. Overview of the germeval 2018 shared task on the identification of offensive language. In *14th Conference on Natural Language Processing KONVENS 2018*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

WLV-RIT at GermEval 2021: Multitask Learning with Transformers to Detect Toxic, Engaging, and Fact-Claiming Comments

Skye Morgan[1], **Tharindu Ranasinghe**[2], **Marcos Zampieri**[1]
[1]Rochester Institute of Technology, USA
[2]University of Wolverhampton, UK
sdm9815@rit.edu

Abstract

This paper addresses the identification of toxic, engaging, and fact-claiming comments on social media. We used the dataset made available by the organizers of the GermEval-2021 shared task containing over 3,000 manually annotated Facebook comments in German. Considering the relatedness of the three tasks, we approached the problem using large pre-trained transformer models and multitask learning. Our results indicate that multitask learning achieves performance superior to the more common single task learning approach in all three tasks. We submit our best systems to GermEval-2021 under the team name WLV-RIT.

1 Introduction

The popularity and accessibility associated with social media have greatly promoted user-generated content. At the same time, social media sites have increasingly become more prone to offensive content (Hada et al., 2021; Zhu and Bhat, 2021; Bucur et al., 2021). As such, identifying the toxic language in social media is a topic that has gained, and continues to gain traction. Research surrounding the problem of offensive content has centered around the application of computational models that can identify various forms of negative content such as hate speech (Malmasi and Zampieri, 2018; Nozza, 2021), abuse (Corazza et al., 2020), aggression (Kumar et al., 2018, 2020), and cyber-bullying (Rosa et al., 2019; Cheng et al., 2021; Salawu et al., 2021).

GermEval-2021 (Risch et al., 2021) focuses on identifying multiple types of comments in social media. This year's shared task is divided into three distinct classifications of comments: *i*) Toxic, *ii*) Engaging, and *iii*) Fact-Claiming. Like previous GermEval shared tasks (Struß et al., 2019), the detection of toxic content remains an integral part of GermEval-2021. Regarding engaging comments, there is an increasing desire from community managers as well as moderators to identify valuable user content (Kolhatkar and Taboada, 2017; Napoles et al., 2017). More particularly, rational comments that serve to encourage readers to engage in a discussion. In a similar light, identifying fact-claiming comments is equally important as platforms need to consistently review and verify user-generated content to uphold their responsibility as information distributors (Mihaylova et al., 2018; Shaar et al., 2020).

We pose that multitask learning (MTL) is a suitable approach for this year's GermEval as it enables what is learned from each task to aid in the learning of other tasks. The current state-of-the-art approach for offensive language identification is neural transformers modeled using single task learning (SLT) (Liu et al., 2019; Ranasinghe and Zampieri, 2020). It is well-known that training large neural transformer models often result in long processing times. As GermEval-2021 features three related tasks, from a performance standpoint, we pose that training a model jointly on three tasks is likely to be computationally more efficient than training three models in isolation. Moreover, as GermEval-2021 provides a single dataset for the three tasks, MTL can also be used to help improving performance across tasks. As such, we introduce multitask learning whereby one model can predict all three tasks as an alternative approach.

In this paper, we present the methods and results of the WLV-RIT submission to the GermEval-2021 shared task. We explore transformer architectures in two different environments, single task learning and multitask learning, and describe them in detail in Section 4. We perform several experiments using three transformer models that support German and evaluate their performance on the GermEval-2021 dataset.

2 Related Work

The identification of offensive language in online discussions is an extensive topic that has become popular over the past several years. The majority of the research related to this topic is centered on English data due to the availability of annotated datasets (Zampieri et al., 2019a; Rosenthal et al., 2021). Notwithstanding this, offensive language datasets are being annotated in other languages. Researchers have examined offensive content across multiple social media platforms and have both annotated and utilized data from different languages such as Greek (Pitenis et al., 2020), Marathi (Gaikwad et al., 2021), Italian (Chiril et al., 2019), Portuguese (Fortuna et al., 2019; Vargas et al., 2021), Arabic (Mubarak et al., 2021), Turkish (Çöltekin, 2020), and multiple languages of India (Ranasinghe and Zampieri, 2021a).

Past approaches to tackling the problem of offensive content on social media have relied on using a variety of computational models ranging from traditional machine learning classifiers such as Logistic Regression and SVMs (Malmasi and Zampieri, 2018), to various deep learning models (de Gibert et al., 2018). SemEval-2019 Task 5 (HatEval) (Basile et al., 2019) presented the challenge of detecting the presence of hate speech and identifying further features in hateful contents, which included two sub-tasks. For subtask A, which was the hate speech (HS) category, the best performance was achieved by training a support vector machine (SVM) model with a radial basis function (RBF) kernel. Several other high scoring teams used a convolutional neural network (CNN) which was traditionally the most popular approach to this topic (Hettiarachchi and Ranasinghe, 2019). For TRAC-1 (Kumar et al., 2018), the challenge was to develop a classifier that could discriminate between three levels of aggression in social media. The results showed that with careful consideration, classifiers like SVM and even random forest could perform at par with deep neural networks. However, in the end, more than half of the top 15 systems were trained on neural networks which demonstrates the approach's effectiveness.

The introduction of BERT (Devlin et al., 2019) spurred the use of pre-trained transformer models for classifying offensive speech (Ranasinghe and Zampieri, 2021b). As a result, neural transformer based language models have increasingly become more popular in offensive language identification. The use of pre-trained BERT models, as well as BERT-based models, was shown to be able to achieve competitive performance in popular competitions such as OffensEval (Zampieri et al., 2019b, 2020). Language-specific and multilingual models have also been introduced to assist NLP research in various languages such as GBERT for German (Chan et al., 2020), AraBERT for Arabic (Antoun et al., 2020), and the multilingual XLM-R (Conneau et al., 2019) that has been been applied to offensive language identification (Ranasinghe and Zampieri, 2020, 2021c).

3 Data

In the GermEval-2021 dataset, the focus has been extended beyond the identification of offensive comments to include two additional classes: engaging comments that can motivate readers to participate in conversations, and fact-claiming comments. The dataset for this iteration of GermEval comprises over 3,000 Facebook user comments that have been extracted from the page of a political talk show of a German television broadcaster. The training dataset has a total of 3,244 instances and comprises 1,074 instances without any toxic, engaging or fact claiming content. In Table 1, we present four different Facebook user comments along with their annotation.

Toxic	Engaging	Fact-Claiming	Training
0	0	0	1074
1	0	0	739
0	1	0	239
1	1	0	89
0	1	1	403
1	0	1	160
0	0	1	406
1	1	1	134
All			3244

Table 2: GermEval 2021 - Training Set User Comment Distribution

4 Methods

Considering the success that neural transformers have demonstrated across various natural language processing tasks (Uyangodage et al., 2021; Jauhiainen et al., 2021; Hettiarachchi and Ranasinghe, 2020a) including offensive language identification (Ranasinghe and Zampieri, 2020, 2021b; Dai et al., 2020) we used transformers to tackle this task too.

Comment	Sub1	Sub2	Sub3
”Die AfD sind genau so neoliberal und kapitalistische Zerstörer unserer Heimat, wie die CDU, CSU, FDP, SPD und Grüne auch.”	1	0	0
”Sarazin ist ein rechtsradikaler Mensch. Ein Menschenhasser. Sie kennen nur Zerstörung. Die Geschichte hat es gezeigt.”	1	0	1
”@USER, du hast das Thema im Kern nicht verstanden”	0	0	1
”Ich frage dich, verlassen Menschen gerne ihre Heimat?”	0	0	0

Table 1: Annotation examples of four different Facebook user comments. Sub1 represents toxic comments, Sub2 stands for engaging comments, and Sub3 stands for fact claiming.

Parameter	Value
learning rate‡	$1e^{-5}$
number of epochs‡	3
adam epsilon	$1e^{-8}$
warmup ratio	0.1
warmup steps	0
max grad norm	1.0
max seq. length	120
gradient accumulation steps	1

Table 3: Hyperparameter specifications. The optimised hyperparameters are marked with ‡ and their optimal values are reported. The rest of the hyperparameter values are kept as constants.

We explored transformer architectures in two different environments; single task learning and multi task learning.

Single Task Learning (STL) For the STL environment we trained three classification models based on transformers. By utilizing the hidden representation of the classification token (CLS) in the transformer model, we predict the target labels (toxic/non-toxic, engaging/non-engaging, fact-claiming, non-fact-claiming) by applying a linear transformation followed by the softmax activation (σ):

$$\hat{\mathbf{y}}_{task} = \sigma(\mathbf{W}_{[CLS]} \cdot \mathbf{h}_{[CLS]} + \mathbf{b}_{[CLS]}) \qquad (1)$$

where $\cdot$ denotes matrix multiplication, $\mathbf{W}_{[CLS]} \in \mathcal{R}^{D\times 3}$, $\mathbf{b}_{[CLS]} \in \mathcal{R}^{1\times 2}$, and D is the dimension of the input activation layer $\mathbf{h}$. $\hat{\mathbf{y}}_{task}$ is the predicted value of any of the three tasks.

We construct three separate classification models minimising the cross-entropy loss for each of the three tasks as defined in the Equation 2, where y_{toxic}, y_{engage} and y_{fact} represent ground truth labels of each task. These particular losses are:

$$\mathcal{L}_{toxic} = -\sum_{i=1}^{2} \Big(\mathbf{y}_{toxic} \otimes \log(\hat{\mathbf{y}}_{toxic})\Big)[i]$$

$$\mathcal{L}_{engage} = -\sum_{i=1}^{2} \Big(\mathbf{y}_{engage} \otimes \log(\hat{\mathbf{y}}_{engage})\Big)[i]$$

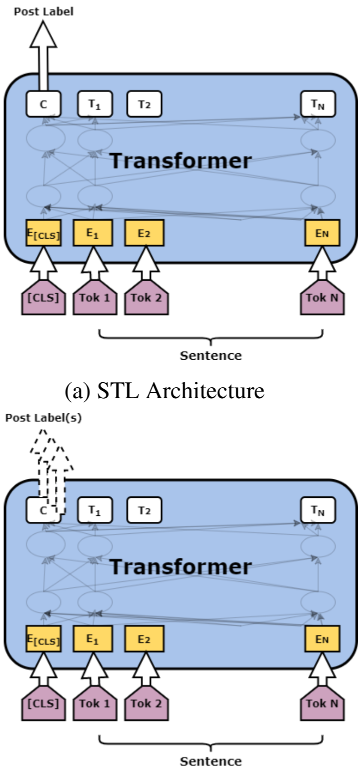
$$\mathcal{L}_{fact} = -\sum_{i=1}^{2} \Big(\mathbf{y}_{fact} \otimes \log(\hat{\mathbf{y}}_{fact})\Big)[i] \qquad (2)$$

where $\mathbf{v}[i]$ retrieves the ith item in a vector $\mathbf{v}$ and $\otimes$ indicates element-wise multiplication. The corresponding STL architecture is shown in Figure 1a.

Multi Task Learning (MTL) MTL was introduced as an approach to inductive transfer (Caruana, 1997). The main goal of which was to improve generalization performance on a current task after having learned a different but related concept on a previous task. MTL is quite efficient as one model can be utilized to predict multiple tasks so long as they are related. In hate speech and offensive language detection, MTL has been shown to outperform single-task environments as well as learn task efficiently with the presence of little labelled data per-task (Djandji et al., 2020). Despite this, MTL has not been used much in the context of offensive language detection. As such, we decided to use multitask learning to compare the performance within the two different environments using different transformer models. We used the transformer as the base model for our MTL approach. Our approach will learn the three tasks jointly, i.e., Toxic comment detection, Engaging comment detection and Fact-claiming comment detection. The implemented architecture shares the hidden layers between the tasks. The shared portion includes a transformer model that learns shared information across the tasks by minimizing a combined loss.

		Toxic			Engaging			Fact-Claiming		
Model	**Environment**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
mBERT	STL	0.4897	0.4421	0.4500	0.5421	0.5310	0.5380	0.5532	0.5093	0.5511
	LM + STL	0.4921	0.4432	0.4512	0.5436	0.5314	0.5398	0.5669	0.5101	0.5521
	MTL	0.5042	0.4449	0.4551	0.5472	0.5325	0.5401	0.5702	0.5113	0.5532
	LM + MTL	0.5063	0.4543	0.4665	0.5542	0.5341	0.5442	0.5732	0.5231	0.5555
gBERT	STL	0.6449	0.5801	0.6102	0.6449	0.6312	0.6342	0.6812	0.6752	0.6852
	LM + STL	0.6552	0.5841	0.6173	0.6254	0.6442	0.6354	0.6821	0.6779	0.6872
	MTL	0.7001	0.6321	0.6654	0.6777	0.6931	0.6841	0.7311	0.7211	0.7352
	LM + MTL‡	0.7124	0.6456	0.6796	0.6827	0.7027	0.6926	0.7450	0.7495	**0.7472**
gELECTRA	STL	0.6551	0.5991	0.6227	0.6391	0.6482	0.6431	0.6954	0.7002	0.7045
	LM + STL	0.6651	0.6078	0.6321	0.6422	0.6561	0.6555	0.7021	0.7102	0.7100
	MTL‡	0.7256	0.6603	0.6914	0.6895	0.6999	**0.6947**	0.7530	0.7407	0.7468
	LM + MTL‡	0.7542	0.6732	**0.7112**	0.6944	0.6924	0.6934	0.7354	0.7383	0.7369

Table 4: Results for the evaluation set in each task with Transformer models. For each model, Precision (P), Recall (R), and F1 are reported on all tasks. The best result for each task has been marked with bold considering F1. The experiments we submitted are marked with ‡



(a) STL Architecture

(b) MTL Architecture

Figure 1: The STL (top) and MTL (bottom) transformer-based architectures experimented with the GermEval-2021 dataset.

We assign equal importance to each task in our experiments. The full loss is:

$$\mathcal{L}_{multi} = \frac{\mathcal{L}_{toxic} + \mathcal{L}_{engage} + \mathcal{L}_{fact}}{3}. \qquad (3)$$

The task-specific classifiers receive input from the last hidden layer of the transformer language model and predict the output for the tasks. The corresponding MTL architecture is shown in Figure 1b

5 Experimental Setup

We performed experiments using three transformer models that support German; mBERT (Devlin et al., 2019), German BERT-large (gBERT) (Chan et al., 2020) and German Electra-large (gELECTRA) (Chan et al., 2020) transformer models available in the HuggingFace model repository (Wolf et al., 2020).

We used an Nvidia Tesla K80 GPU to train the models. We divided the input dataset into a training set and a validation set using 0.8:0.2 split. We predominantly fine-tuned the learning rate and the number of epochs of the classification model manually to obtain the best results for the validation set. We obtained $1e^-5$ as the best value for the learning rate and 3 as the best value for the number of epochs. We used a batch size of 8 for the training process and the model was evaluated after every 100 batches. We performed *early stopping* if the validation loss did not improve over 10 evaluation steps. The rest of the hyperparameters which we kept as constants are mentioned in the Table 3. For both STL and MTL we finetuned the considered transformer model on the GermEval 2021 training set using Masked Language Modeling (MLM) (Devlin et al., 2019) objective which we call as

		Toxic			Engaging			Fact-Claiming		
Model	**Environment**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
mBERT	STL	0.5081	0.4672	0.4781	0.5689	0.5561	0.5555	0.5763	0.5286	0.5761
	LM + STL	0.5162	0.4657	0.4782	0.5698	0.5561	0.5568	0.5871	0.5389	0.5780
	MTL	0.5284	0.4672	0.4781	0.5690	0.5571	0.5678	0.5901	0.5364	0.5782
	LM + MTL	0.5243	0.4763	0.4871	0.5762	0.5590	0.5601	0.5983	0.5482	0.5782
gBERT	STL	0.6692	0.6092	0.6354	0.6678	0.6572	0.6532	0.7095	0.6982	0.7011
	LM + STL	0.6752	0.6072	0.6342	0.6453	0.6683	0.6572	0.7063	0.6982	0.7041
	MTL	0.7223	0.6532	0.6842	0.6954	0.7132	0.7041	0.7553	0.7493	0.7562
	LM + MTL‡	0.7321	0.6654	0.6941	0.7041	0.7298	0.7145	0.7653	0.7602	0.7652
gELECTRA	STL	0.6752	0.6111	0.6498	0.6531	0.6679	0.6609	0.7178	0.7285	0.7265
	LM + STL	0.6874	0.6231	0.6562	0.6666	0.6742	0.6731	0.7231	0.7303	0.7367
	MTL‡	0.7456	0.6802	0.7132	0.7001	0.7101	**0.7198**	0.7754	0.7652	**0.7653**
	LM + MTL‡	0.7853	0.6997	**0.7342**	0.7132	0.7156	0.7190	0.7542	0.7563	0.7590

Table 5: Results for the test set in each task with Transformer models. For each model, Precision (P), Recall (R), and F1 are reported on all tasks. The best result for each task has been marked with bold considering F1. The experiments we submitted are marked with ‡

Language Modeling (LM). When performing training, we trained five models with different random seeds and considered the majority-class self ensemble mentioned in Hettiarachchi and Ranasinghe (2020b) to get the final predictions.

6 Results

We show the results for the evaluation set in Table 4. In all the experimented transformer models, the MTL approach outperformed the STL approach. Furthermore in most scenarios, the systems that included a LM component outperformed those without the LM component. This corroborates the findings of previous research in offensive language identification (Ranasinghe et al., 2019). gBERT and gELECTRA models clearly outperformed mBERT in all the tasks. For the Task 1, gELECTRA model with LM and MTL achieved the best result with 0.7342 F1 score, for the Task 2 gELECTRA model with MTL, without LM achieved the best result with 0.7198 F1 score and for the Task 3 too, the same model achieved the best result with 0.7653 F1 score. Considering the overall performance we selected three best models for the submission; gELECTRA with LM+MTL, gELECTRA with MTL and gBERT with LM+MTL.

The official leaderboard of the competition was not yet released at the time of writing this paper, therefore, after the organizers released the gold labels for the test set, we calculated the Precision, Recall, and F1 values for the test set. The results are shown in Table 5. As shown in the results, the three models we selected provided the top three results for the test set too. MTL consistently outperformed STL in all the tasks with all the transformer models we experimented.

7 Conclusion and Future Work

In this paper, we presented the WLV-RIT entry to GermEval-2021. GermEval-2021 provided participants with the opportunity of testing computational models to identify toxic, engaging, and fact claiming comments. We experimented with neural transformer models in STL environment and MTL environment. MTL environment consistently outperformed STL suggesting that the use of shared learning methods improves the performance of individual tasks. Furthermore, we observed that pre-trained language-specific transformer models trained for German such as gBERT and gElectra outperform mBERT. Finally, in addition to the transformer-based MTL approach, we could observe that the use of language modelling led performance improvement in some of the tasks.

In the future, we would like to carry out an error analysis on the output of our systems to better understand the impact and limitations of MTL for these three tasks. Finally, we would like to experiment with multi-task learning in other languages, particularly low-resource languages for which only limited language resources are available.

Acknowledgments

The authors would like to thank the GermEval-2021 organizers for organizing this interesting shared task and for making the dataset available.

References

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. AraBERT: Transformer-based model for Arabic language understanding. In *Proceedings of LREC*.

Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. 2019. Semeval-2019 task 5: Multilingual detection of hate speech against immigrants and women in twitter. In *Proceedings of SemEval*.

Ana-Maria Bucur, Marcos Zampieri, and Liviu P. Dinu. 2021. An exploratory analysis of the relation between offensive language and mental health. In *Findings of the ACL*.

Rich Caruana. 1997. Multitask Learning. *Machine Learning*, 28:41–75.

Çağrı Çöltekin. 2020. A Corpus of Turkish Offensive Language on Social Media. In *Proceedings of LREC*.

Branden Chan, Stefan Schweter, and Timo Möller. 2020. German's next language model. In *Proceedings of COLING*.

Lu Cheng, Ahmadreza Mosallanezhad, Yasin Silva, Deborah Hall, and Huan Liu. 2021. Mitigating bias in session-based cyberbullying detection: A non-compromising approach. In *Proceedings of ACL*.

Patricia Chiril, Farah Benamara Zitoune, Véronique Moriceau, Marlène Coulomb-Gully, and Abhishek Kumar. 2019. Multilingual and multitarget hate speech detection in tweets. In *Proceedings of TALN*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. In *Proceedings of ACL*.

Michele Corazza, Stefano Menini, Elena Cabrio, Sara Tonelli, and Serena Villata. 2020. Hybrid emoji-based masked language models for zero-shot abusive language detection. In *Findings of the ACL*.

Wenliang Dai, Tiezheng Yu, Zihan Liu, and Pascale Fung. 2020. Kungfupanda at SemEval-2020 task 12: BERT-based multi-TaskLearning for offensive language detection. In *Proceedings of SemEval*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of NAACL*.

Marc Djandji, Fady Baly, Wissam Antoun, and Hazem Hajj. 2020. Multi-task learning using AraBert for offensive language detection. In *Proceedings of OSCAT*.

Paula Fortuna, Joao Rocha da Silva, Leo Wanner, Sérgio Nunes, et al. 2019. A Hierarchically-labeled Portuguese Hate Speech Dataset. In *Proceedings of ALW*.

Saurabh Gaikwad, Tharindu Ranasinghe, Marcos Zampieri, and Christopher Homan. 2021. Cross-lingual offensive language identification for low resource languages: The case of Marathi. In *Proceedings of RANLP*.

Ona de Gibert, Naiara Perez, Aitor García-Pablos, and Montse Cuadros. 2018. Hate speech dataset from a white supremacy forum. In *Proceedings of ALW*.

Rishav Hada, Sohi Sudhir, Pushkar Mishra, Helen Yannakoudakis, Saif M. Mohammad, and Ekaterina Shutova. 2021. Ruddit: Norms of offensiveness for English Reddit comments. In *Proceedings of ACL*.

Hansi Hettiarachchi and Tharindu Ranasinghe. 2019. Emoji Powered Capsule Network to Detect Type and Target of Offensive Posts in Social Media. In *Proceedings of RANLP*.

Hansi Hettiarachchi and Tharindu Ranasinghe. 2020a. BRUMS at SemEval-2020 task 3: Contextualised embeddings for predicting the (graded) effect of context in word similarity. In *Proceedings of SemEval*.

Hansi Hettiarachchi and Tharindu Ranasinghe. 2020b. InfoMiner at WNUT-2020 task 2: Transformer-based covid-19 informative tweet extraction. In *Proceedings of W-NUT*.

Tommi Jauhiainen, Tharindu Ranasinghe, and Marcos Zampieri. 2021. Comparing approaches to Dravidian language identification. In *Proceedings of VarDial*.

Varada Kolhatkar and Maite Taboada. 2017. Using New York Times Picks to Identify Constructive Comments. In *Proceedings of NLPJ*.

Ritesh Kumar, Atul Kr Ojha, Shervin Malmasi, and Marcos Zampieri. 2018. Benchmarking Aggression Identification in Social Media. In *Proceedings of TRAC*.

Ritesh Kumar, Atul Kr. Ojha, Shervin Malmasi, and Marcos Zampieri. 2020. Evaluating aggression identification in social media. In *Proceedings of TRAC*.

Ping Liu, Wen Li, and Liang Zou. 2019. NULI at SemEval-2019 task 6: Transfer learning for offensive language detection using bidirectional transformers. In *Proceedings of SemEval*.

Shervin Malmasi and Marcos Zampieri. 2018. Challenges in Discriminating Profanity from Hate Speech. *Journal of Experimental & Theoretical Artificial Intelligence*, 30:1–16.

Tsvetomila Mihaylova, Preslav Nakov, Lluís Màrquez, Alberto Barrón-Cedeño, Mitra Mohtarami, Georgi Karadzhov, and James Glass. 2018. Fact Checking in Community Forums. In *Proceedings of AAAI*.

Hamdy Mubarak, Ammar Rashed, Kareem Darwish, Younes Samih, and Ahmed Abdelali. 2021. Arabic Offensive Language on Twitter: Analysis and Experiments. In *Proceedings of WANLP*.

Courtney Napoles, Joel Tetreault, Aasish Pappu, Enrica Rosato, and Brian Provenzale. 2017. Finding Good Conversations Online: The Yahoo news annotated comments corpus. In *Proceedings of LAW*.

Debora Nozza. 2021. Exposing the limits of zero-shot cross-lingual hate speech detection. In *Proceedings of ACL*.

Zeses Pitenis, Marcos Zampieri, and Tharindu Ranasinghe. 2020. Offensive Language Identification in Greek. In *Proceedings of LREC*.

Tharindu Ranasinghe and Marcos Zampieri. 2020. Multilingual Offensive Language Identification with Cross-lingual Embeddings. In *Proceedings of EMNLP*.

Tharindu Ranasinghe and Marcos Zampieri. 2021a. An Evaluation of Multilingual Offensive Language Identification Methods for the Languages of India. *Information*, 12(8).

Tharindu Ranasinghe and Marcos Zampieri. 2021b. MUDES: Multilingual Detection of Offensive Spans. In *Proceedings of NAACL*.

Tharindu Ranasinghe and Marcos Zampieri. 2021c. Multilingual Offensive Language Identification for Low-resource Languages. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*.

Tharindu Ranasinghe, Marcos Zampieri, and Hansi Hettiarachchi. 2019. BRUMS at HASOC 2019: Deep Learning Models for Multilingual Hate Speech and Offensive Language Identification. In *Proceedings of FIRE*.

Julian Risch, Anke Stoll, Lena Wilms, and Michael Wiegand. 2021. Overview of the GermEval 2021 shared task on the identification of toxic, engaging, and fact-claiming comments. In *Proceedings of GermEval*.

Hugo Rosa, N Pereira, Ricardo Ribeiro, Paula Costa Ferreira, Joao Paulo Carvalho, S Oliveira, Luísa Coheur, Paula Paulino, AM Veiga Simão, and Isabel Trancoso. 2019. Automatic cyberbullying detection: A systematic review. *Computers in Human Behavior*, 93:333–345.

Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Marcos Zampieri, and Preslav Nakov. 2021. A Large-Scale Weakly Supervised Dataset for Offensive Language Identification. In *Findings of the ACL*.

Semiu Salawu, Jo Lumsden, and Yulan He. 2021. A large-scale English multi-label Twitter dataset for cyberbullying and online abuse detection. In *Proceedings of WOAH*.

Shaden Shaar, Alex Nikolov, Nikolay Babulkov, Firoj Alam, Alberto Barrón-Cedeno, Tamer Elsayed, Maram Hasanain, Reem Suwaileh, Fatima Haouari, Giovanni Da San Martino, et al. 2020. Overview of CheckThat! 2020 English: Automatic identification and verification of claims in social media. In *Proceedings of CLEF*.

Julia Maria Struß, Melanie Siegel, Josep Ruppenhofer, Michael Wiegand, and Manfred Klenner. 2019. Overview of germeval task 2, 2019 shared task on the identification of offensive language. In *Proceedings GermEval*.

Lasitha Uyangodage, Tharindu Ranasinghe, and Hansi Hettiarachchi. 2021. Can Multilingual Transformers Fight the COVID-19 Infodemic? In *Proceedings of RANLP*.

Francielle Alves Vargas, Fabiana Rodrigues de Góes, Isabelle Carvalho, Fabrício Benevenuto, and Thiago Alexandre Salgueiro Pardo. 2021. Contextual lexicon-based approach for hate speech and offensive language detection. *arXiv preprint arXiv:2104.12265*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of EMNLP*.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019a. Predicting the Type and Target of Offensive Posts in Social Media. In *Proceedings of NAACL*.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019b. SemEval-2019 Task 6: Identifying and Categorizing Offensive Language in Social Media (OffensEval). In *Proceedings of SemEval*.

Marcos Zampieri, Preslav Nakov, Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Hamdy Mubarak, Leon Derczynski, Zeses Pitenis, and Çağrı Çöltekin. 2020. SemEval-2020 Task 12: Multilingual Offensive Language Identification in Social Media (OffensEval 2020). In *Proceedings of SemEval*.

Wanzheng Zhu and Suma Bhat. 2021. Generate, prune, select: A pipeline for counterspeech generation against online hate speech. In *Findings of the ACL*.

Precog-LTRC-IIITH at GermEval 2021: Ensembling Pre-Trained Language Models with Feature Engineering

T. H. Arjun and **Arvindh A.** and **Ponnurangam Kumaraguru**
Precog-LTRC, International Institute of Information Technology, Hyderabad, India
{arjun.thekoot,arvindh.a}@research.iiit.ac.in
pk.guru@iiit.ac.in

Abstract

We describe our participation in all the subtasks of the Germeval 2021 shared task on the identification of Toxic, Engaging, and Fact-Claiming Comments. Our system is an ensemble of state-of-the-art pre-trained models finetuned with carefully engineered features. We show that feature engineering and data augmentation can be helpful when the training data is sparse. We achieve an F1 score of 66.87, 68.93, and 73.91 in Toxic, Engaging, and Fact-Claiming comment identification subtasks.

1 Introduction

Facebook quickly rose in popularity around 2008, taking the world by storm single-handedly creating the initial social media buzz. Its user base is steadily increasing ever since and has held its position as the most used platform ever since the early 2010s.[1] It has around 2.38 billion users, and the increase hasn't flattened yet. The initial purpose of such social media platforms was to establish a bridge for fruitful information exchange, which is currently inhibited by offensive language and misinformation spread. Given the number of comments exchanged each day, it's impossible to manually classify and mitigate such behavior.

GermEval is a series of shared task evaluation campaigns that focus on natural language processing for the German language. GermEval 2021 tasks are intended to classify comments on Facebook into three categories of Toxic, Engaging, and Fact-Claiming comments. Subtask A focuses on the identification of offensive language which could be used to ban/timeout these users. Subtask B on Fact-claiming can further be classified as misinformation, and Subtask C on engaging comments promoting cleaner information exchange. The outline of this paper is as follows: We give a short overview of related work in Section 2. We then describe the dataset provided in Section 3 and the preprocessing techniques we use in Section 4, explain the features we engineered in Section 5, and the architecture of our solution in Section 6. We then move onto the evaluation of our solution in Sections 7-9 and conclude in Section 10.

2 Related Work

2.1 Toxic Comment Classification

There have been various shared tasks and competitions in this task such as: GermEval Task 2, 2019 (Struß et al., 2019), GermEval 2018 (Wiegand et al., 2018), SemEval 2019 - Task 5 (Basile et al., 2019), SemEval 2019 - Task (OffensEval 2019) (Zampieri et al., 2019), SemEval 2020 (Zampieri et al., 2020), Kaggle's Toxic Comment Classification Challenges.[2]

Wu et al. (2019) use the BERT model to detect and classify offensive language in English tweets and obtain good results. Risch and Krestel (2020b) discuss toxic comments in online news discussions and describe subclasses of toxicity, present various deep learning approaches, and propose to augment training data by using transfer learning when the training data is sparse.

2.2 Engaging Comment Classification

Risch and Krestel (2020a) analyze user engagement in the form of the upvotes and replies that the comments receive. They train a model to classify based on text and achieve excellent results with RNN and CNN models. They also analyze what makes each comment engaging. Ambroselli et al. (2018) use a Logistic Regression Model with metadata, along with extracted semantic and linguistic

[1] Statistics https://bit.ly/3AZdQtj

[2] kaggle-challenge https://bit.ly/3hZMYAx

features. Napoles et al. (2017) use a CNN with word embeddings to classify engaging threads.

2.3 Fact-Claiming Comment Classification

Chatterjee et al. (2018) propose combining BOW and manually engineered features for classifying facts and opinions on Twitter and show that handcrafted textual features could help in the task. Hassan et al. (2015) propose a feature-based method in which sentiment, TF-IDF, part-of-speech, and other descriptive features are fed into classical models, such as SVMs. There have been various other deep learning-based attempts as well (Atanasova et al., 2018). Meng et al. (2020) identify fact-claiming text using a Bert Model and use adversarial training to avoid overfitting.

All previous attempts at these tasks show how feature engineering and deep learning approaches can be helpful in these tasks.

3 Dataset

The dataset provided for the shared task (Risch et al., 2021) is an annotated dataset of Facebook user comments that four trained annotators have labeled. The dataset was collected from the Facebook page of a political talk show of a German television broadcaster (information about which was not revealed to the participants), consisting of user discussions from February till July, 2019. The dataset provided is anonymized. Links to users are replaced by @USER, likewise links to the show replaced by @MEDIUM, and the links to the show's moderator replaced by @MODERATOR. Each comment of the dataset is annotated into three categories - Toxic, Fact-Claiming, and Engaging. The test set contains 944 comments extracted from different shows other than the one in the training data. This way, the participants were provided with a realistic use case and could possibly test a possible bias caused by topics of discussion. There is an imbalance in the distribution of classes in the given dataset. Still, we let the models be biased with this class imbalance as we believe it provides our models a fair understanding of these distributions from the real world.

4 Data Preprocessing

The corpora is abundant in emojis. We transcribe all emojis into German text instead of removing them while cleaning the text as not to lose information present, such as emotions. For this, we use a transliteration mapping for emojis.[3] We use the googletrans library[4] to translate these to German. We remove hyperlinks, mentions, lower-case the text, remove punctuations except for apostrophes and periods, and perform Unicode normalization. Due to the limit on the number of tokens (512 tokens) for transformer-based models, we cut the text in the middle of the sentence i.e. we take the first 500 and last 12 tokens.

5 Feature Engineering

We look at the linguistic features of the text and explore their correlation with each categorical prediction.

5.1 Stylistic Features

Features include total length, number of unique words, words, exclamation, question marks, all capital words, the percentage of unique words, other punctuations, URLs, distribution of emojis, etc.

5.2 Linguistic Features

We use a list of German stopwords from the nltk library (Loper and Bird, 2002) and use their distribution as a feature. We use SentiWS Dataset (Remus et al., 2010), which provides negative and positive sentiment scores for words. We use this to get the percentage of negative and positive sentiment scores. We use Language Tool[5] which can detect a variety of linguistic anomalies, including grammatical errors, missing punctuation, or wrong capitalization. We note these errors and we propose the distribution of them as a feature. FTR Classifier[6] is a natural language classifier that uses keyword-based methods to identify future-referring sentences and whether they use the present tense, future tense, or express epistemic certainty or uncertainty. It also has a list of German past, future, uncertain, certain words which we use. German grammatical features such as Partizip (Participle), Partizip II (Past Participle), Präteritum (Preterite) are taken from the German Verbs Database.[7] We note down the distribution of the 10 verb categories mentioned in the database. We also use the Dale Chall Readability Index Calculation (Dale and Chall, 1948) to find the

[3] Emoji-list https://bit.ly/3wtom8E

[4] googletrans https://bit.ly/3yNW0Y5

[5] language-tool-python https://bit.ly/3yKb6Og

[6] FTR Classifier https://bit.ly/3xER8Vk

[7] German-Verbs-Database https://bit.ly/3i1BDzZ

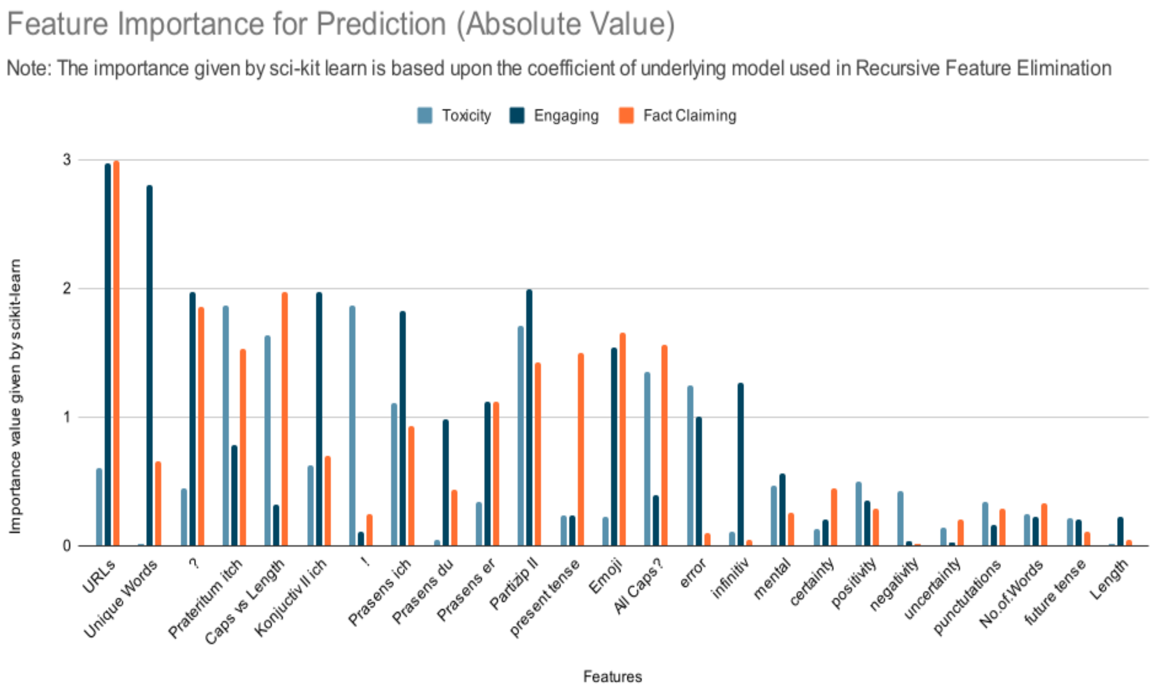


Figure 1: We look at the importance of features using scikit-learn feature selection and choose 20 important features. The plot shows the importance marked by feature selection (absolute values).

readability of the text. For applying this to the German text, we use the python library readability.[8]

We perform feature selection on these hand-crafted features using two filters, Pearson correlation and scikit-learn library's[9] feature selection. We choose the top 20 that we presume to be essential for our models.

Figure 1 shows the importance (absolute values) marked by the scikit-learn library's feature selection. After feature selection, we select the following 20 features:

- Readability
- Number of '!', '?', words, URLs
- Percentage of all Capital Words, Partizip II, Präteritum, Punctuations, Linguistic Errors, Präsens ich, words in present, and future tense, unique words, "certainty" and "uncertainty" words.
- Positive and Negative Sentiment score
- Moderator mentions
- Distribution of emojis

6 Model Architectures

We formulate the tasks as a Multi-Label Classification Problem as we are trying to address all 3 tasks. To learn the correlation between these classes, all our models are three-headed which output probabilities for 3 classes corresponding to each subtask. Devlin et al. (2019) achieve the best performance when they concatenate the last 4 hidden layers of the pre-trained network for sentence-level tasks, so in all our models, we use the same approach and concatenate the last 4 hidden layers. We experiment with the following models and techniques (after freezing the pre-trained weights):

6.1 Models

- **Pretrained Transformer Based Models with CNN head**: In this approach, we freeze the pre-trained layers and pass the embedding (concatenated last 4 hidden layers) to a CNN. Kim (2014) report state-of-the-art performance on sentence-level classification after max-pooling convolution layers of various widths to a fully connected layer with dropout. We follow a similar approach where we pass the concatenated last 4 hidden layers of the pre-trained model to convolution lay-

[8] readability https://bit.ly/2U1XKym
[9] scikit-learn https://bit.ly/3i6j8ut

ers of filter sizes 2,3,4,5, on which we apply Max Pooling of pool size 3. We concatenate these outputs with a dropout of 0.5, which is then passed onto a Dense Layer of size 128 with ReLu activation succeeded by a dropout of 0.5. We concatenate this 128 dimensional vector with our 20 dimensional hand-crafted feature vector. We pass this output onto a dense output layer of dimension 3 with sigmoid activation.

- **Pretrained Transformer Based Models with Capsule Net head**: In the image classification domain, capsule networks (Hinton et al., 2011; Sabour et al., 2017) prove to be effective at understanding spatial relationships. Kim et al. (2018) apply this network structure to the classification of text and show its advantage. They argue that CNNs could extract features, but CNNs cannot understand the spatial and proportional relationships between objects in the images or words. Capsule networks address this problem by learning the spatial relationships between words (in text) using additional encoded information. We apply this network architecture with pre-trained embeddings. We pass the pre-trained embeddings through a Bi-Directional GRU Layer of dimension 128 with ReLu activation and dropout of 0.25. We pass this through a Capsule Network of 5 Capsules, 4 routings, and squash activation. This is followed by a dropout of 0.25 and concatenation with our hand-crafted feature vector. This is then passed onto a 3-dimensional dense output layer with sigmoid activation.

- **Fastext and Glove Embeddings with RNN-GRU head**: Along with the transformer-based models, we train word embedding-based models with a RNN head. Unlike transformer-based models, which use sub-word tokenization, the word embedding models could face Out Of Vocabulary (OOV) words. Therefore, we add an extra data cleaning step to reduce the number of OOV words. We deploy a spell checker and correct spellings if possible. For the embeddings layer, we concatenate German fastText (Grave et al., 2018) and German Glove Embeddings (Pennington et al., 2014).[10] We then pass the embeddings through a dropout of 0.5 followed by Bi-Directional LSTM of kernel size 40. This is then passed through a Bi-Directional GRU of the same kernel size. We concatenate the average pool, maximum pool, and the last layer output with our hand-crafted feature vector. This is then passed onto a dense output layer of size 3 with sigmoid activation.

6.2 Ensembling

Our approach uses two levels of Ensembling:

- **Fold Level Ensembling**: We implement early stopping and save the best checkpoint during k-Fold validation for each proposed model. We make a prediction on the test set for each best checkpoint, which we average out to get the best prediction over the k-folds.

- **Model-Level Ensembling**: The predictions of each of the proposed models for each of the pre-trained language models are averaged.

7 Experiments

7.1 Training Data Augmentation

Since the training data is sparse, we follow the approach by Risch and Krestel (2018) where we augment the training set by translating the text to English and then back again to German. We reuse googletrans library for this. This can give us different forms of the same text. Thanks to the accuracy of Google Translate and assuming the meaning remains the same, we can also assume that the labels remain the same. We randomly pick 600 comments for training from this augmented dataset and concatenate them with the given training set.

The models output probabilities for each class. When the value of an output unit is above a given threshold, the corresponding label is predicted. The optimum was found by varying the threshold for the validation set during k-Fold Validation.

7.2 Baseline

We train a Bert finetuned baseline to compare our models against. The Bert model is finetuned for 7 epochs with early stopping and 10-Fold Cross-Validation. This has a classification head on top of the concatenated last 4 hidden layer CLS Token for sentence classification. We consider this a solid

[10] German glove embeddings by deepset.ai https://bit.ly/3xwE58a

baseline as it is an ensemble across 10-Fold Cross-Validation of the state-of-the-art Pretrained Language Model, which has proven to be very strong in most cases.

7.3 Experimental Setting

All the above approaches were run on four pre-trained models from huggingface hub[11] namely electra-base-german-uncased[12], German convbert[13], bert-base-german-uncased[14], and a multilingual model xlm-roberta-large.[15] We train the models on each of these pre-trained embeddings and we average the predictions of these models resulting in an ensemble.

We train the models with 10-Fold cross-validation. We use Adam optimizer (Loshchilov and Hutter, 2019) with a learning rate of 1e-3 and a batch size of 32. We train the model for 20 epochs with early stopping with a patience of 3. We didn't experiment with the hyperparameters. The models were implemented using Tensorflow,[16] Keras,[17] and Huggingface Transformers Library.[18] We train on the given dataset with augmentation.

8 Results

Experimenting with the models, we achieve the best performance with an ensemble of models mentioned in the architecture section, which is also our submission. The participants were provided with the gold labels for the test set to evaluate the models. In Table 1, we compare our models on the gold labels and with the baseline model. It is also worth noting that we submitted two system runs. The first one was ensemble of all the individual models listed in Table 1, except models with Capsule Net head. In the second system run, we incorporated models with Capsule Net head into the ensemble (ensemble of all individual models in Table 1). The second system run performed better; hence we centered the analysis around it.

9 Analysis

We carry out an analysis of the test set gold labels to find where our models failed. We find that many misclassified comments were very long ones with more than 512 tokens truncated in the middle part. We truncated in the middle as most of the emotions must be concentrated at the two ends. A possible solution could be is to use hierarchical LSTMs with chunking of 512 token chunks of these texts and feeding them to the models or using longformer based models (Beltagy et al., 2020). We analyze some of the misclassified texts by our model below. They were translated by a native German, two non-native speakers, and google translate. (Note: The translations given below are the ones by the native speaker)

1. "Großen Respekt wie Herr Hallervorden mit der Situation und seinen Mitarbeitern umgeht. Wenn es nach Herrn Lauterbach gehen würde ,würden sie es im stillen Kämmerlein aussitzen."
   translates to
   "I pay a lot of respect to how Mr. Hallervorden is dealing with the situation and his co-workers. If it were up to Mr. Lauterbach, they would keep it under the table."

2. "@USER Wissen sie was oder reden Sie einfach auch völlig unfundiert daher? Wenn sie was wissen lassen sie uns an ihrem Wissen teilhaben!"
   translates to
   "@USER Do you know something or are you also speaking fully in unfounded terms? If you know something, let us know about the knowledge you have!"

3. "@USER weil er es kann."
   translates to
   "@USER because he can."

4. "@USER dem kann ich nur zustimmen. Was nützt dem Klima eine CO2 Bepreisung? Finde den Fehler. Aber so generiert man unter dem Deckmantel Klimaschutz neue Abgaben, wir alle werden noch mehr zahlen müssen ohne das sich etwas ändert. Bewährtes Verfahren. Immer mit dem Finger auf die anderen zeigen ist ja so einfach"
   translates to
   "@USER I can agree with that. How could a CO2-tax be useful for climate? Find the mistake. But with that you can implement new taxes under the disguise of climate protection.

[11] huggingface-hub https://huggingface.co/models

[12] electra https://bit.ly/3e8zX6w

[13] convbert https://bit.ly/3wB6Qzt

[14] bert https://bit.ly/3yQXqB8

[15] xlm-roberta https://bit.ly/2TUkzEh

[16] tensorflow https://tensorflow.org/

[17] keras https://keras.io/

[18] Huggingface https://huggingface.co/

	SubTask A			SubTask B			SubTask C		
Model	T1 F1	T1 P	T2 R	T2 F1	T2 P	T2 R	T3 F1	T3 P	T3 R
Baseline	**59.38**	**60.54**	**58.27**	**65.27**	**65.92**	**64.64**	**67.19**	**67.47**	**66.9**
FastText Glove RNN	63.59	67.18	60.37	68.71	68.86	68.67	70.24	71.57	68.97
Bert CNN	62.76	65.67	60.10	67.09	69.13	65.17	73.69	76.19	71.35
Bert BertCapsule Net	64.56	66.28	62.93	67.18	68.28	66.13	73.99	75.41	72.63
Electra CNN	64.82	66.56	63.16	67.00	68.37	65.69	73.49	74.20	72.69
Electra Capsule Net	67.80	72.99	63.31	66.52	66.96	66.07	72.72	72.89	72.55
ConvBert CNN	58.94	60.72	57.27	66.06	67.17	64.99	70.32	71.74	68.97
ConvBert Capsule Net	64.17	67.70	61.00	67.28	69.30	65.37	71.94	73.56	70.40
XLM-Roberta CNN	62.01	68.01	56.98	67.65	68.75	66.59	71.87	72.90	70.88
XLM-Roberta Capsule Net	65.05	67.25	63.00	70.26	70.93	69.60	73.95	76.48	71.59
Ensemble Submission	**66.87**	**67.42**	**66.33**	**68.93**	**68.37**	**69.50**	**73.91**	**73.44**	**74.39**

Table 1: Comparison of various models, including baseline across the three tasks in which the ensemble submission incorporates Capsule Net.

> We all will have to pay more without any improvement. Best practice. It is always easy to point finger at others."

In Comment 1, the gold label is toxic. Without context, it could also be classified as non-toxic, since it is congratulatory in the first part. Comments 2 and 3 were classified as toxic but are non-toxic. One could note that both are in a rude tone. This could be because of the context of the comment and to what it is referring to.

For engaging comments, some misclassified comments in our analysis were both toxic and engaging, which is strange without context. For subtask 3, comment 4 was classified as Fact-claiming by the model, but the comment seems to be claiming a practice.

We find in our testing that hand-crafted features could be crucial in improving the performance of pre-trained finetuning for low-resource tasks. We also notice no discrepancy between precision and recall even though there was a class imbalance in the training set. Hence, our hypothesis that the model benefits from learning the class distributions and their correlations in the real world is validated.

10 Conclusion

Participating in all three shared tasks, we submit predictions from a model ensemble. We perform feature engineering and dataset augmentation and show how this can help train neural networks in low-resource tasks. Our model ensemble with hand-crafted features performs better than the baseline Fine-Tuned Bert Model. We also analyze the errors made by our model against the gold label to understand the flaws in the model. We have also made the source code public[19] for reference.

11 Acknowledgments

We want to thank Google Colab for free access to GPUs and TPUs for quick and small experimentation in teams and visualization help, IIIT-H for providing us with access to HPC Cluster with GPUs for training our deep learning models and also to our colleagues at Precog[20] for constant support and feedback.

References

Carl Ambroselli, Julian Risch, Ralf Krestel, and Andreas Loos. 2018. Prediction for the newsroom: Which articles will get the most comments? In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 3 (Industry Papers)*, pages 193–199, New Orleans - Louisiana. Association for Computational Linguistics.

[19] Source code `https://bit.ly/36zOjJg`

[20] Precog Lab `https://precog.iiitd.edu.in/`

Pepa Atanasova, Alberto Barron-Cedeno, Tamer Elsayed, Reem Suwaileh, Wajdi Zaghouani, Spas Kyuchukov, Giovanni Da San Martino, and Preslav Nakov. 2018. Overview of the clef-2018 checkthat! lab on automatic identification and verification of political claims. task 1: Check-worthiness.

Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. 2019. SemEval-2019 task 5: Multilingual detection of hate speech against immigrants and women in twitter. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 54–63, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer.

Swayambhu Chatterjee, Shuyuan Deng, Jun Liu, Ronghua Shan, and Wu Jiao. 2018. Classifying facts and opinions in twitter messages: a deep learning-based approach. *Journal of Business Analytics*, 1(1):29–39.

Edgar Dale and Jeanne S. Chall. 1948. A formula for predicting readability. *Educational Research Bulletin*, 27(1):11–28.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*, pages 4171–4186. Association for Computational Linguistics.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

Naeemul Hassan, Chengkai Li, and Mark Tremayne. 2015. Detecting check-worthy factual claims in presidential debates. *Proceedings of the 24th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 1835–1838.

Geoffrey E. Hinton, Alex Krizhevsky, and Sida D. Wang. 2011. Transforming auto-encoders. In *Proceedings of the 21th International Conference on Artificial Neural Networks - Volume Part I*, page 44–51. Springer-Verlag.

Jaeyoung Kim, Sion Jang, Sungchul Choi, and Eunjeong Park. 2018. Text classification using capsules.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751. Association for Computational Linguistics.

Edward Loper and Steven Bird. 2002. Nltk: The natural language toolkit. In *In Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics. Philadelphia: Association for Computational Linguistics*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization.

Kevin Meng, Damian Jimenez, Fatma Arslan, Jacob Daniel Devasier, Daniel Obembe, and Chengkai Li. 2020. Gradient-based adversarial training on transformer networks for detecting check-worthy factual claims.

Courtney Napoles, Aasish Pappu, and Joel R. Tetreault. 2017. Automatically identifying good conversations online (yes, they do exist!). In *ICWSM*.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Robert Remus, Uwe Quasthoff, and Gerhard Heyer. 2010. SentiWS - a publicly available German-language resource for sentiment analysis. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*. European Language Resources Association (ELRA).

Julian Risch and Ralf Krestel. 2018. Aggression identification using deep learning and data augmentation. In *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018)*, pages 150–158, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Julian Risch and Ralf Krestel. 2020a. Top comment or flop comment? predicting and explaining user engagement in online news discussions. In *Proceedings of the International Conference on Web and Social Media (ICWSM)*, pages 579–589.

Julian Risch and Ralf Krestel. 2020b. *Toxic Comment Detection in Online Discussions*, pages 85–109.

Julian Risch, Anke Stoll, Lena Wilms, and Michael Wiegand. 2021. 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 colocated with KONVENS*, pages 1–12.

Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. 2017. Dynamic routing between capsules.

Julia Struß, Melanie Siegel, Josef Ruppenhofer, Michael Wiegand, and Manfred Klenner. 2019. Overview of germeval task 2, 2019 shared task on the identification of offensive language.

Michael Wiegand, Melanie Siegel, and Josef Ruppenhofer. 2018. Overview of the germeval 2018 shared task on the identification of offensive language.

Zhenghao Wu, Hao Zheng, Jianming Wang, Weifeng Su, and Jefferson Fong. 2019. BNU-HKBU UIC NLP team 2 at SemEval-2019 task 6: Detecting offensive language using BERT model. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 551–555, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. Semeval-2019 task 6: Identifying and categorizing offensive language in social media (offenseval). In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 75–86.

Marcos Zampieri, Preslav Nakov, Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Hamdy Mubarak, Leon Derczynski, Zeses Pitenis, and Çağrı Çöltekin. 2020. SemEval-2020 task 12: Multilingual offensive language identification in social media (OffensEval 2020). In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1425–1447. International Committee for Computational Linguistics.

IRCologne at GermEval 2021: Toxicity Classification

Fabian Haak **Björn Engelmann**
TH Köln
Gustav-Heinemann-Ufer 54
50968 Köln
fabian.haak@th-koeln.de, bjoern.engelmann@th-koeln.de

Abstract

In this paper, we describe the TH Köln's submission for the "*Shared Task on the Identification of Toxic Comments*" at GermEval 2021. Toxicity is a severe and latent problem in comments in online discussions. Complex language model based methods have shown the most success in identifying toxicity. However, these approaches lack explainability and might be insensitive to domain-specific renditions of toxicity. In the scope of the GermEval 2021 toxic comment classification task (Risch et al., 2021), we employed a simple but promising combination of term-frequency-based classification and rule-based labeling to produce effective but to no lesser degree explainable toxicity predictions.

1 Introduction

Toxic language in online comments and discussions is an increasingly relevant problem (Mathew et al., 2019). However, toxicity classification is a challenging task (Paasch-Colberg et al., 2021). There is no universally agreed on complete definition of toxicity. Instead, toxicity is an umbrella term for a variety of problematic, negative phenomena (Malik et al., 2021). Since, even for human annotators, it often is hard to explain why precisely a comment is or is not toxic, language models trained on an extensive dataset of labeled comments usually performed best in the task of identifying toxicity (Zhao et al., 2021). However, these approaches lack explainability, and when the systems are employed to filter user-composed comments, the system should be able to indicate what aspect(s) of the comment lead to it being declared as toxic. More traditional approaches like support vector machines, linear models, or naive Bayes classification do not perform as well, generally. Nevertheless, TFIDF-based classification has clear advantages in terms of ease of domain adaptability and explainability. Our approach employed in the GermEval 2021 toxicity classification task (Risch et al., 2021) combines these traditional classification methods with the Snorkel framework (Ratner et al., 2017). Using labeling functions (LFs), we identify indicators for aspects of toxicity to enable explainable toxicity judgments and improve classification performance.

2 Toxicity: Definition, Aspects, Classification.

Depending on the definition, a wide range of aspects defines what constitutes toxic comments (Wulczyn et al., 2017). As described by Georgakopoulos et al. (2018), a toxic comment, aside from exhibiting verbal violence, can also be "a comment that is rude, disrespectful or otherwise likely to make someone leave a discussion". This definition is consistent with the definition of toxicity given in the overview paper, where toxic comments are deemed problematic because they discourage and hamper participation in discussions. (Risch et al., 2021). To better tackle the issue of toxic language in comments, we categorized the aspects of toxicity in comments into three categories. The first category of toxicity-defining aspects can be characterized as language aspects. These describe features like particular vocabularies or attributions that carry a toxic tonality. They range from different forms of hate speech such as racism (Kwok and Wang, 2013), sexism (Jha and Mamidi, 2017), fanaticism and identity hate, to profane, offensive and aggressive language and incivility (Risch et al., 2019, 2021). Of the toxic features listed in the task description, vulgar language, screaming, and insults are listed as toxic comment features (Risch et al., 2021). The second category can be described as toxic behavioral aspects, defined by their communicative intentions. They are composed of cyberbullying (Chavan and Shylaja, 2015), sexual predation (McGhee et al., 2011), threats, and so-called spam messages (Founta et al., 2018). Sarcasm and zynism, discrimination, discrediting, accusations, and threats are the aspects of toxic comments mentioned in Risch et al. (2021) that best fit this category. Finally, the most latent type of toxic aspects can be grouped as inappropriate language. These highly context- or domain-specific aspects span from age-inappropriateness (Alshamrani et al., 2021) to general undesirable topics or off-topic messages.

In one of the rare approaches to more explainable toxicity classification, Xiang et al. (2021) tried to address the issue of poor explainability of language model classification techniques. However, their approach is decidedly different from ours. By assuming that a text is at least as toxic as its most toxic part, they focused their work on increasing the explainability of transformer-based classification. As previously mentioned, this is a rare exception since most recent approaches utilize deep learning and attention-based language models like BERT (Devlin et al., 2018).

3 Methodological Approach

Our approach applies binary classification (BC) and data programming on a preprocessed version of the provided corpus. In this section, we give a brief overview of all aspects of this approach.

3.1 Preprocessing

Before any classifier is trained or other potential biased patterns in the text are addressed, the comment texts are preprocessed. Since the original texts are needed for labeling functions, the cleaned texts are saved separately. The preprocessing consists of the following steps:

1. Removing single-character-words
2. Deleting any html snippets
3. Discard all characters that are not European ASCII characters (f.e. digits)
4. Removing any white space characters
5. Tokenization using the TweetTokenizer provided by the NLTK (Bird et al., 2009)
6. Excluding all tokens from NLTK's list of German stopwords
7. Stemming using the Cistem German Stemmer (Weißweiler, 2017)

3.2 Binary Classification

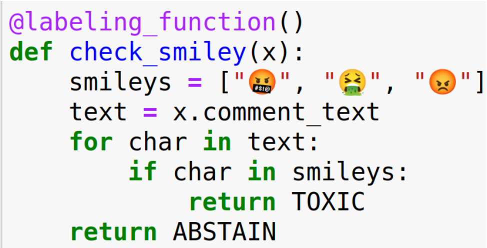
Especially deep learning models with a large number of training parameters require an extensive data set (Feng et al., 2021). Although datasets for toxic comments for pre-training would have existed for the classification of English texts, for German texts, these do not exist. We decided that translating texts from English to German or using datasets with labels for toxic aspects such as sexism or hate speech would induce too much noise. Therefore, we chose to base our model solely on the training data provided by GermEval. Since the training dataset is relatively small with 3244 labeled comments, we considered four different simple model types (see subsection 4.1). This model then serves as a baseline to ensure complete coverage across all comments.

3.3 Data Programming

Ratner et al. have presented a data programming framework (Snorkel) that produces noisy labels using user-defined labeling functions (Ratner et al., 2017). These labeling functions can express simple rules such as regular expressions or more complex heuristics that use external resources. Snorkel is typically used to solve tasks where no labeled dataset is available by combining these labeling functions to produce provisional labels to train a discriminative model. Snorkel has been successfully used for various NLP tasks, such as named entity recognition (Lison et al., 2020), fake news detection (Shu et al., 2020), and spam classification (Maheshwari et al., 2020).

3.4 Labeling Functions

Labeling functions express simple heuristics that assign either a label, in our case Toxic, OK, or abstain, to label an input comment. An example is shown in Figure 1. Each labeling function should represent either a toxic or normal aspect for a comment as part of a larger set of labeling functions to move the classification in one direction. Our approach to producing labeling functions is to examine incorrect labels from the output of our classification model, namely false positives and false negatives. We only examine the incorrect labels from the training data to prevent overfitting. In this human-in-the-loop approach, new labeling functions can be defined iteratively after every evaluation step to improve the final classification performance (Wu et al., 2018).



```
@labeling_function()
def check_smiley(x):
    smileys = ["🤬", "🤮", "😡"]
    text = x.comment_text
    for char in text:
        if char in smileys:
            return TOXIC
    return ABSTAIN
```

Figure 1: Labeling function for toxic emojis.

We can then use Snorkel to examine the following properties for a set of labeling functions (Ratner et al., 2017):

- **Coverage**: Is a measure of how large the proportion of data is for which this labeling function has not been abstained from.
- **Overlap**: Proportion of data for which at least one other labeling function has also assigned a label.
- **Conflicts**: The fraction of the dataset where this LF and at least one other LF label and disagree.
- **Empirical Accuracy**: The empirical accuracy of this LF (if gold labels are provided).

With these properties, the LFs can be evaluated manually and adjusted if necessary. Snorkel can learn a generative model based on the correlations and accuracies among the LFs. This generative model serves as our final classification model.

4 Results

4.1 Classification Models

We randomly split the provided data into 80% train data and 20% for test data. We performed a parameter search for the following classifiers using scikit-learn's Grid Search (Pedregosa et al., 2011).

- Logistic Regression (LR)
- Support Vector classifier (SVC)
- Linear SVC
- Multi-layer Perceptron classifier(MLP)

For LR, SVC and Linear SVC we evaluated the inverse regulation strength $C \in \{0.8, 1, 1.2\}$. For the MLP, we tested the following sizes for the hidden states $(h_1, h_2, h_3) \in \{(10, 10, 10), (20, 20, 20), (40, 40, 40), (80, 80, 80)\}$.

Method	Accuracy	F1-Score (Macro)
Logistic Regression	67.8	51.78
Linear SVC	65.49	57.38
SVC	57.16	54.28
MLP	62.1	57.62

Table 1: Best Results for different classifiers after grid search parameter optimization.

All classifiers were also evaluated with the following tf-idf parameters for the sklearn-learn TFidfVectorizer:

- word n-gram range $\in \{(1, 1), (1, 2), (1, 3)\}$
- minimum document frequency threshold $min_{df} \in \{0, 0.02, 0.04, 0.06\}$
- maximum document frequency threshold $max_{df} \in \{0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$

The results of the classifiers can be seen in Table 1. The evaluation metrics in our case are accuracy and F1 Score, as the classes in the training dataset are not balanced. We have chosen to use the Linear SVC model as it has the best balance between accuracy and F1 score.

4.2 Labeling Functions

van Aken et al. (2018) have analyzed the results of machine learning classification for toxicity prediction and identified aspects of toxicity that are hard to detect by these models. Similarly, labeling functions for improving the overall classification results are derived from the false positive and false negative classifications that the linear SVC model produced. We hoped to find indicators in the form of patterns that match aspects for toxic comments we established in 2. The potential patterns we identified are:

- **Quotations**: In some false positives, potentially toxic text is presented by the author in quotation marks to show that it is not their thoughts or opinions but something they comment on. For example: ”*Sag mal ...willst du eine Menschen mit einer ”Rostlaube” vergleichen?? Wie impertinent !!*”. We decided to implement this feature in a second submission, where we included the deletion of quotations as an additional preprocessing step.

- **Sentiment**: Negative sentiment could be a universal indicator for toxic comments. As mentioned by van Aken et al. (2018), this could especially be the case if the toxicity is latent and topic-dependent. In the false negatives produced by our binary classification model, comments such as ”*100% der AfD scheißt auf die Menschenrechte*” or ”*Was eine dämliche Diskussion . Sind wir jetzt völlig verblödet. Ich muss abschalten .Ich bekomme Kopfschmerzen.*” also show negative sentiment. We implemented labeling functions with different thresholds for BAWL-R (Võ et al., 2009), SentiWS (Goldhahn et al., 2012) and German Polarity Cues (Waltinger, 2010) sentiment lexica, as well as German Sentiment (Guhr et al., 2020), a BERT-based sentiment classification approach. Although there were some indications that negative sentiment could be a signal for toxicity, we could not identify any obvious differences between toxic and non-toxic texts. Since this was supported by poor-performing LFs, we did not include any sentiment LFs into our classification.

- **Capitalization**: Capitalizations of words could indicate aggressive language, for example in ”*@MEDIUM Wenn Sie als objektive Presse, die sich an Fakten zu halten haben, da Sie auch einen BILDUNGSAUFTRAG haben, sich Pro-Organspende aussprechen sollten, muss man sie entweder der Organ- und/oder PharmaMafia zuordnen oder aber als Lügenpresse bezeichnen bzw erkennen, dass Sie Ihrem Auftrag nicht gerecht werden können. Ihnen würde dann Unfähigkeit attestiert. Alles nicht wirklich nett!*”. We quickly realized, that in many cases, 3-character words that are capitalized are abbreviations. Therefore, the labeling function only checks for words longer than three characters.

- **Sarcasm and Ridiculing**: Some text elements like certain emojis at the end of comments (🤣, 😂, and 👍, especially when used in multiples) and the term ”*haha*” in various variations and lengths could indicate toxic language.

- **Punctuation**: The use of multiple exclamation points or question marks at the end of sentences is common amongst the falsely negative classified comments. As implemented in the VADER sentiment analysis tool developed by Hutto and Gilbert (2014), multiple punctuation marks at the end of sentences enforce the sentiment of the sentence. Since we expect this to be the case for German texts and what we find in our dataset matches this assumption, we employ the phenomenon as a toxicity indicator as a labeling function. In addition, some falsely classified toxic comments show multiple ellipses, possibly indicating annoyance, such as in ”*Und überhaupt...wenn ich Spahn schon sehe...*🤢”.

LF	Coverage	Overlap	Acc.
question	0.011	0.005	0.897
exclamation	0.018	0.009	0.957
emojis	0.009	0.003	0.826
caps	0.04	0.011	0.625
haha	0.078	0.015	0.379
short_sens	0.057	0.013	0.265
ellipses	0.055	0.02	0.464

Table 2: Properties of the LFs. Those marked in green are part of the final classification and those marked in red have been discarded. None of the LFs showed any conflicts with other functions.

- **Toxic emojis**: Certain emojis that seem to be used to indicate disgust or anger appear almost exclusively in toxic comments and frequently appear in the list of false negatives. Therefore, we introduced a labeling function that checks for appearances of these emojis (c.f. Figure 1)

- **Insults**: We found a lot of insults in the false negatives, such as ”*Moralapostel*”, ”*Trendlemminge*”, or ”*Menschenhasser*”. A labeling function was created that labels texts based on the appearance of insults from a list of German insults from insult.wiki. However, the LF coverage and accuracy were low, probably due to the complexity and context-relatedness of German insults.

Table 2 shows the properties of some of the labeling functions, including those used in our classification. The first two LFs check whether there are three question marks or exclamation marks in a row in the comment. The high accuracy scores of the LFs indicate that multiple exclamation points and question marks indicate toxicity. The rela-

Method	Accuracy	F1-Score (Macro)
Linear SVC	65.49	57.38
Linear SVC + LF	**67.95**	**62.31**

Table 3: Comparison of the final classification with the baseline model performed on the split-produced test data.

tively higher accuracy for exclamation points could be explained by the fact that German exclamation points are used to signal imperative sentences, which could be perceived as toxic in the context of a discussion. As previously described, capitalized words and the use of particular emojis are also a sign of toxicity. This is also confirmed by our exploration and empirical accuracy of the emojis-LF (c.f. Figure 1). We discarded the remaining LFs marked in red because of the low accuracy since these patterns do not indicate toxicity for texts of the given corpus. The discarded LFs check whether the phrase "*haha*" is included, whether a comment contains at least one ellipsis, and short_sens checks, whether the comment consists of sentences with an average length of two or less words. Of the three categories of toxic aspects of comments described in section 2, mostly language aspects are effectively covered by the LFs. Covering the more latent behavioral aspects like discrimination, sarcasm, or threats indirectly by negative emotions was ineffective. Since inappropriateness is context-dependent and the context of the dataset is unknown, the LFs do not cover inappropriateness aspects of toxic comments.

Table 3 shows that the use of LFs led to an increase in Accuracy of 2.46 points and an increase in the Macro f1 score of 4.93.

4.3 Classification Results

Finally, we used our approach for classifying the provided evaluation test dataset. We produced two almost identical classification runs. The only difference is that text in quotation marks was removed from the training and test data in the second run. As described in subsection 4.2, this aimed at ignoring references to other potentially toxic comments. However, as Table 4 shows, with a F1 score of 0.576 (P: 0.582, R:0.57), the run with no filtering of quotes performed slightly better. Risch et al. (2021) compares the results of all submitted systems.

ID	F1	P	R
1: BC + LFs	0.576	0.582	0.570
2: BC + LFs + Quot.	0.574	0.580	0.568

Table 4: Classification results of the provided test data. Run 1 results were produced only using binary classification and data programming. For run 2 the same methods were used, but as described in subsection 4.2, text in quotation marks was removed from training and test data.

5 Conclusion

To overcome the difficulties posed by GermEval 2021's toxicity classification task, we combined traditional linear SVC classification with labeling functions based on false negative and false positive classifications of the model. This combined approach is able to deliver explainable results and adaptability. Despite the total coverage of the LFs in the training data of about 5% and although we discarded most of the developed labeling functions due to bad classification performance, data programming increased the classification's performance significantly. Including the four best performing labeling functions, our final classification model increased the F1-score by almost 5 points. This increase indicates that the toxic attributes covered by the LFs have not been taken into account by the linear SVC classifier. On the evaluation dataset, our approach reached an F1-score of 0,576. Overall, the approach was a success. However, a more extensive dataset might have benefited our linear SVC model's and our labeling functions' performance.

References

Betty van Aken, Julian Risch, Ralf Krestel, and Alexander Löser. 2018. Challenges for toxic comment classification: An in-depth error analysis. *CoRR*, abs/1809.07572.

Sultan Alshamrani, Ahmed Abusnaina, Mohammed Abuhamad, Daehun Nyang, and David Mohaisen. 2021. Hate, obscenity, and insults: Measuring the exposure of children to inappropriate comments in youtube.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python*, 1st edition. O'Reilly Media, Inc.

Vikas S Chavan and S S Shylaja. 2015. Machine learning approach for detection of cyber-aggressive comments by peers on social media network. In *2015 International Conference on Advances in Computing,* *Communications and Informatics (ICACCI)*, pages 2354–2358.

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

Steven Y. Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard H. Hovy. 2021. A survey of data augmentation approaches for NLP. *CoRR*, abs/2105.03075.

Antigoni-Maria Founta, Constantinos Djouvas, Despoina Chatzakou, Ilias Leontiadis, Jeremy Blackburn, Gianluca Stringhini, Athena Vakali, Michael Sirivianos, and Nicolas Kourtellis. 2018. Large scale crowdsourcing and characterization of twitter abusive behavior.

Spiros V. Georgakopoulos, Sotiris K. Tasoulis, Aristidis G. Vrahatis, and Vassilis P. Plagianakos. 2018. Convolutional neural networks for toxic comment classification.

Dirk Goldhahn, Thomas Eckart, and Uwe Quasthoff. 2012. Building large monolingual dictionaries at the leipzig corpora collection: From 100 to 200 languages. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pages 759–765, Istanbul, Turkey. European Languages Resources Association (ELRA).

Oliver Guhr, Anne-Kathrin Schumann, Frank Bahrmann, and Hans Joachim Böhme. 2020. Training a broad-coverage german sentiment classification model for dialog systems. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 1620–1625, Marseille, France. European Language Resources Association.

C. Hutto and Eric Gilbert. 2014. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *ICWSM*.

insult.wiki. Liste der deutschen schimpfwörter.

Akshita Jha and Radhika Mamidi. 2017. When does a compliment become sexist? analysis and classification of ambivalent sexism using twitter data. In *Proceedings of the Second Workshop on NLP and Computational Social Science*, pages 7–16, Vancouver, Canada. Association for Computational Linguistics.

Irene Kwok and Yuzhou Wang. 2013. Locate the hate: Detecting tweets against blacks. In *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence*, AAAI'13, page 1621–1622. AAAI Press.

Pierre Lison, Aliaksandr Hubin, Jeremy Barnes, and Samia Touileb. 2020. Named entity recognition without labelled data: A weak supervision approach. *CoRR*, abs/2004.14723.

Ayush Maheshwari, Oishik Chatterjee, KrishnaTeja Killamsetty, Rishabh K. Iyer, and Ganesh Ramakrishnan. 2020. Data programming using semi-supervision and subset selection. *CoRR*, abs/2008.09887.

Pranav Malik, Aditi Aggrawal, and Dinesh K. Vishwakarma. 2021. Toxic speech detection using traditional machine learning models and bert and fasttext embedding with deep neural networks. In *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*, pages 1254–1259.

Binny Mathew, Ritam Dutt, Pawan Goyal, and Animesh Mukherjee. 2019. Spread of hate speech in online social media. In *Proceedings of the 10th ACM Conference on Web Science*, WebSci '19, page 173–182, New York, NY, USA. Association for Computing Machinery.

India McGhee, Jennifer Bayzick, April Kontostathis, Lynne Edwards, Alexandra McBride, and Emma Jakubowski. 2011. Learning to identify internet sexual predation. *International Journal of Electronic Commerce*, 15(3):103–122.

Sünje Paasch-Colberg, Christian Strippel, Joachim Trebbe, and Martin Emmer. 2021. From insult to hate speech: Mapping offensive language in german user comments on immigration. *Media and Communication*, 9:171–180.

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

Alexander Ratner, Stephen H. Bach, Henry Ehrenberg, Jason Fries, Sen Wu, and Christopher Ré. 2017. Snorkel: Rapid training data creation with weak supervision. *Proc. VLDB Endow.*, 11(3):269–282.

Julian Risch, Anke Stoll, Lena Wilms, and Michael Wiegand. 2021. Overview of the GermEval 2021 shared task on the identification of toxic, engaging, and fact-claiming comments. In *Proceedings of the GermEval 2021 SharedTask on the Identification of Toxic, Engaging, and Fact-Claiming Comments co-located with KONVENS*, pages 1–12.

Julian Risch, Anke Stoll, Marc Ziegele, and Ralf Krestel. 2019. hpidedis at germeval 2019: Offensive language identification using a german bert model. In *KONVENS*.

Kai Shu, Guoqing Zheng, Yichuan Li, Subhabrata Mukherjee, Ahmed Hassan Awadallah, Scott Ruston, and Huan Liu. 2020. Leveraging multi-source weak social supervision for early detection of fake news. *CoRR*, abs/2004.01732.

Melissa Võ, Markus Conrad, Lars Kuchinke, Karolina Urton, Markus Hofmann, and Arthur Jacobs. 2009. The berlin affective word list reloaded (bawl-r). *Behavior research methods*, 41:534–8.

Ulli Waltinger. 2010. Germanpolarityclues: A lexical resource for german sentiment analysis. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC)*, Valletta, Malta. electronic proceedings.

Leonie Weißweiler. 2017. Developing a stemmer for german based on a comparative analysis of publicly available stemmers. In *Proceedings of the International Conference of the German Society for Computational Linguistics and Language Technology*, Berlin, Germany. German Society for Computational Linguistics and Language Technology.

Sen Wu, Luke Hsiao, X. Cheng, Braden Hancock, Theodoros Rekatsinas, P. Levis, and C. Ré. 2018. Fonduer: Knowledge base construction from richly formatted data. *Proceedings of the 2018 International Conference on Management of Data*.

Ellery Wulczyn, Nithum Thain, and Lucas Dixon. 2017. Ex machina: Personal attacks seen at scale.

Tong Xiang, Sean MacAvaney, Eugene Yang, and Nazli Goharian. 2021. ToxCCIn: Toxic content classification with interpretability. In *Proceedings of the Eleventh Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 1–12, Online. Association for Computational Linguistics.

Zhixue Zhao, Ziqi Zhang, and Frank Hopfgartner. 2021. *A Comparative Study of Using Pre-Trained Language Models for Toxic Comment Classification*, page 500–507. Association for Computing Machinery, New York, NY, USA.

DeTox at GermEval 2021: Toxic Comment Classification

Mina Schütz[1], Christoph Demus[2], Jonas Pitz[1], Nadine Probol[1], Melanie Siegel[1], Dirk Labudde[2]

[1] Darmstadt University of Applied Sciences
Max-Planck-Straße 2, 64807 Dieburg
{mina.schuetz, melanie.siegel}@h-da.de
{jonas.pitz, nadine.probol}@stud.h-da.de

[2] Fraunhofer Institute for Secure Information Technology
Rheinstraße 75, 64295 Darmstadt
{christoph.demus, dirk.labudde}@sit.fraunhofer.de

Abstract

In this work, we present our approaches on the toxic comment classification task (subtask 1) of the GermEval 2021 Shared Task. For this binary task, we propose three models: a German BERT transformer model; a multilayer perceptron, which was first trained in parallel on textual input and 14 additional linguistic features and then concatenated in an additional layer; and a multilayer perceptron with both feature types as input. We enhanced our pre-trained transformer model by re-training it with over 1 million tweets and fine-tuned it on two additional German datasets of similar tasks. The embeddings of the final fine-tuned German BERT were taken as the textual input features for our neural networks. Our best models on the validation data were both neural networks, however our enhanced German BERT gained with a F1-score = 0.5895 a higher prediction on the test data.

1 Introduction

In recent years social media platforms became a popular medium to discuss all kinds of topics with people around the world. Also shops, companies, TV-shows and many more use social media to present their content to followers and discuss it with them. As it is possible to interact almost anonymously on the internet, such social media pages are often confronted with the problem of hate speech and toxic comments targeting single persons or whole groups (Watanabe et al., 2018). Although hate speech detection has been a top research topic for several years, there exists no satisfactory solution yet (Struß et al., 2019). The GermEval Shared Task 2021 (Risch et al., 2021) addresses this topic - especially the side of social media moderators that are responsible to filter such comments - in this years challenge with the following three tasks, where we participate in subtask 1:

- Subtask 1: toxic comment classification
- Subtask 2: engaging comment classification
- Subtask 3: fact-claiming

Over the last years transformer (Vaswani et al., 2017) models like BERT (Bidirectional Encoder Representations with Transformers) (Devlin et al., 2019) became state-of-the-art for many natural language processing (NLP) tasks and regularly outperformed traditional machine learning models and neural networks (Zampieri et al., 2020; Kumar et al., 2020). Nevertheless, the GermEval Shared Task 2019 showed that traditional machine learning methods can still achieve comparable results to the transformer models if the features are well chosen (Struß et al., 2019).

Therefore, we decided to experiment with standard supervised machine learning models and neural networks, different word embeddings, and pre-trained transformer models. We then chose our best performing transformer model, enhanced it with re-training on extracted tweets in German, and fine-tuned it with additional datasets. The extracted word embeddings by our transformer model were used as an textual input for our neural network architectures besides additional features.

Our presented work is structured as follows: Section 2 gives an overview of related work. In Section 3 we describe the GermEval 2021 data and the additional data we used for our final models. In Section 4 the feature extraction, the baseline and the final models are described. In Section 5, we show our final results and discuss our models.

2 Related Work

Toxic speech can be defined as a combination of hate speech and offensive language (D'Sa et al., 2020) or a type of aggressive writing style (Maslej-Krešňáková et al., 2020). Many recent research uses deep neural networks for such detection tasks in social media content (Georgakopoulos et al., 2018; van Aken et al., 2018). There has also been some research with transformer models, especially for English social media content. Maslej-Krešňáková et al. (2020) compared multiple transformers and neural networks for the classification of toxic content with different types of preprocessing steps, focussing on word embeddings. However, some related work to our modelling approach has been done by researchers in similar content detection tasks on social media.

Sohn and Lee (2019) used, in their study on hate speech detection with transformer models, a similar approach to our proposed models, after they fine-tuned a multi-channel BERT model: they applied a dropout on the [CLS] token of BERT and added a feed forward layer before the softmax output and calculated the weighted sum of three transformers instead of only one. The [CLS] token is the final hidden vector of BERT used for classification, however it can also be extracted for the models embeddings (Devlin et al., 2019). This was also done in (Rodríguez-Sánchez et al., 2020) for the task of automatic sexism classification, where the authors added features with a feed forward layer on top, however this did not improve their results. They also - in comparison to our concatenation strategy for our multilayer perceptron - created a Bi-LSTM (Bidirectional Long-Short-Term-Memory), where they concatenated the additional extracted features (in this case user and network information) after going through several layers of the neural network with only using textual input. Their work showed that using pre-trained embeddings for neural networks pushes the final classification by 3% (Rodríguez-Sánchez et al., 2020).

The study of Zhao et al. (2021) found that using pre-trained models as an input for neural networks leads to better results than using complex deep neural networks or transformers as a stand-alone architecture. Comparingly, another approach by D'Sa et al. (2020) on hate speech detection analyzed FastText (Bojanowski et al., 2017) and BERT embeddings and used them as the input for deep neural networks without any additional feature extraction. They found that fine-tuning transformers without a neural network layer performs better.

	Toxic	Not Toxic	Total
Train	1122 (35.6%)	2122 (64.4%)	3244
Test	350 (37.1%)	594 (62.9%)	944
Total	1472	2716	4188

Table 1: Class distribution for subtask 1 of the GermEval 2021 dataset. Percentages show the proportion of toxic and non-toxic comments in the training and test set.

Those studies show that combining transformers that are fine-tuned for a specific NLP task with neural networks is a promising approach to create better models for predicting toxic comments. Since transformers are usually only used for training on the textual input, the feed forward layers can be concatenated with more extracted features.

3 Data

In this section we describe the GermEval 2021 Shared Task dataset as well as the supplementary datasets that we used for fine-tuning our model.

3.1 GermEval 2021 Data

The dataset for the GermEval 2021 Shared Task contains 3244 user comments from the Facebook discussion page of a German news broadcast within the first half of 2019. The comments were anonymized and cleared of any references to the show, moderators and users. The dataset was provided with manual annotated labels for each of the subtasks. Table 1 shows that 35.6% of all comments are labeled as *Toxic* for subtask 1 while 64.4% are labeled as *Not Toxic*.

3.2 Additional Datasets

Augmentation allows a transformer model to be fine-tuned with additional labeled data (Schütz et al., 2021). In order to augment the GermEval 2021 training data we identified two German datasets that were labeled for hateful or offensive comment classification and shared a similar domain. We assumed that the tasks of identifying hateful and offensive comments should be similar to the task of identifying toxic comments.

- **GermEval 2019:** Task 2 of GermEval 2019 was a shared task on the identification and categorization of offensive language (Struß et al., 2019). For subtask 1 of this shared task a total of 7025 tweets were collected and labeled as either *OFFENSE* or *OTHER* with 32.1% of the tweets being labeled the former. The label *OFFENSE* was given to any comment that was deemed abusive, insulting and/or profane. Comparably to what we would expect from comments about a daily talk show the tweets in this dataset were chosen to cover a broad range of topics.

- **HASOC 2019:** HASOC (Hate Speech and Offensive Content Identification in Indo-European Languages) 2019 was a shared task comparable to GermEval Task 2 but with the addition of providing 3 separate datasets for German, English and Hindi (Mandl et al., 2019). The German dataset contains a total of 4669 tweets and Facebook posts collected by searching for offensive keywords and hashtags. 11.6% of the entries for subtask 1 are labeled as *HOF* while the rest is labeled as *NOT*. The categories *HOF* and *NOT* directly correspond to the categories *OFFENSE* and *OTHER* from Task 2 of GermEval 2019.

3.3 German Tweet Corpus:

For several unsupervised training steps in our experiments we also collected a total of unlabeled 1,156,458 German tweets of the first half year of 2019 via the Twitter API. Mainly, we focused on general tweets in German, as well as tweets from the Twitter pages of German talk shows and other socially critical TV-formats: ”Hart aber Fair”, ”Maybrit Illner”, ”Anne Will”, ”Markus Lanz”, ”ZDF heute-show” and ”Maischberger”. With this extra data we expected to enhance the predictions of our models, since the dataset hopefully contains tweets with a similar writing style and domain-specific politically discussed content by that time period.

4 Methodology

In this section the feature extraction methods as well as the baseline we used for comparison, the conducted preprocessing steps, and final models are described. Our baseline models include different combinations.

4.1 Feature Extraction

For training some of our models, we used several features as listed in Table 2. It has been shown that adding more specific features about the writing style of social media entries helps to improve the results of similar NLP tasks, such as hate speech and disinformation detection (Robinson et al., 2018; Volkova and Jang, 2018). For toxic comment classification we considered the word count for each input and extracted the number of punctuation, exclamation, and question marks and their relation to the total number of words per comment. For some features we used additional non-public word lists and libraries and cross-checked them for each entry in the dataset:

Feature	Toxic	Not Toxic
word count	201	179
punctuation count	7.41	6.84
exclamation count	0.69	0.31
question mark count	0.48	0.36
word punctuation ratio	0.0111	0.0138
word exclamation ratio	0.0027	0.0021
word question mark ratio	0.0020	0.0030
hate word count	0.32	0.24
hate word count ratio	0.0017	0.0014
character capslock ratio	0.0306	0.0168
sentiment	-0.0147	-0.0080
emoji count	0.49	0.13
emoji sentiment	0.0424	0.0191
word emoji ratio	0.0457	0.0227

Table 2: Extracted features and their mean values in toxic and non-toxic comments.

- ”Sentiment” features: list of 9,382 words and their sentiment values
- ”Hate” features: list of 3,550 words

Lastly, we counted the number of emojis per comment, determined the emoji word ratio and used the emosent[1] library to compute the average sentiment over all emojis in a comment.

We computed the mean values of each feature for both classes and found some significant differences between both categories: for example toxic comments are 22 words longer on average. Besides the length, there is a notable difference in the number of exclamation marks and emojis between toxic and not toxic comments. Contrary to the expectations the sentiment of the comments is in both cases slightly negative and does only differ by 0.0067 on a scale from -1 (most negative) to +1

[1] https://pypi.org/project/emosent-py/

(most positive). Nevertheless, we used all of the extracted features for our experiments.

4.2 Baseline

For our baseline we used a Support Vector Machine (SVM) and a sequential neural network (multilayer perceptron, MLP). Additionally, a Robust Soft Learning Vector Quantization (RSLVQ) model was trained and evaluated. RSLVQ is an adaption of the LVQ Model introduced by Kohonen (1997). In these models, class regions are defined by prototype vectors in the vector space, where each class has one or more prototype vectors. In contrast to the basic LVQ, which is a heuristic, RSLVQ can be mathematically verified (Schneider et al., 2009). Additionally, we tested three pre-trained transformer (Vaswani et al., 2017) models with only using the provided training set by the GermEval 21.

4.2.1 Preprocessing

Multiple preprocessing steps were applied to the SVM and RSLVQ, and the comments were vectorized. The steps included tokenization, stop word and punctuation removal and lemmatization. Hashtags and mentions were preserved in the data, only the characters "#" and "@" were removed. Afterwards 200-dimensional FastText word embeddings were trained on the preprocessed training dataset, on our self collected German Tweet corpus, and on the additional data. For the word embeddings, a skip-gram model with a window-size of 5 and a minimum word occurrence of 3 was used. All the word-vectors of every comment were averaged to receive a document vector.

Additionally, a feature vector for every comment, including the features mentioned in Table 2, was created from the original (not preprocessed) data and concatenated with the document vector.

In contrast, we did not preprocess the data for the transformer models, since those models capture the context of a sentence and use a already specialized built-in tokenizer (Devlin et al., 2019). All of our baseline models were evaluated on a stratified 90% training and 10% validation split.

4.2.2 Experiments

The SVM was trained on the training split using a Radial Basis Function (RBF) and a linear kernel. The best results were achieved with the RBF-kernel. In the RSLVQ model the number of prototypes per class was varied having the best results with two prototypes per class. Already pre-trained FastText embeddings were used as an input for the MLP, where we concatenated the extracted features with the textual input during training (MLP-C) and before (MLP-B). Even though the precision and recall were higher compared to the other models, we found inconsistency in the evaluation plots of the metrics of both models - and due to a high loss during validation, it seemed that both MLPs were overfitting.

Model	Val Pre	Val Rec	Val F1
SVM*	0.57	0.63	0.60
RSLVQ*	0.70	0.43	0.54
MLP-C*	0.65	0.99	0.78
MLP-B*	0.66	0.98	0.79
BERT	0.66	0.65	0.64
DistilBERT	0.67	0.67	0.66
XLM-R	0.71	0.68	0.67

Table 3: Baseline results on the validation split of the GermEval 2021 training data.
*Additional German tweets used for word embeddings.

Finally, we fine-tuned a German BERT (Devlin et al., 2019) and DistilBERT (Sanh et al., 2019) model (bert-base-german-cased (Chan et al.), distilbert-base-german-cased (Chaumond)) provided by the HuggingFace library (Wolf et al., 2020) for 10 epochs, a batch size of 16, a learning rate of 2e-5, Adam (Kingma and Ba, 2015) as an optimizer and a maximum sequence length of 256. The multi-lingual transformer XLM-R (Conneau et al., 2019) was fine-tuned with the same parameters, except a learning rate of 1e-5 instead.

4.3 Models

In total we submitted three different models for each run as shown in Figure 1.

- **Transformer (TAB):** We decided to enhance our best transformer model from our baseline by using the additional German tweets for re-training. This has been shown to help boost the classification accuracy as shown in (Schütz et al., 2021). Re-training means that the pre-trained model is further trained in an unsupervised manner, before fine-tuning it for the NLP downstream task. We chose to re-train with the the german-bert-base-cased model for 5 epochs, with a batch size of 32 and a learning rate of 2e-5. Afterwards, we fine-tuned our re-trained German-BERT model

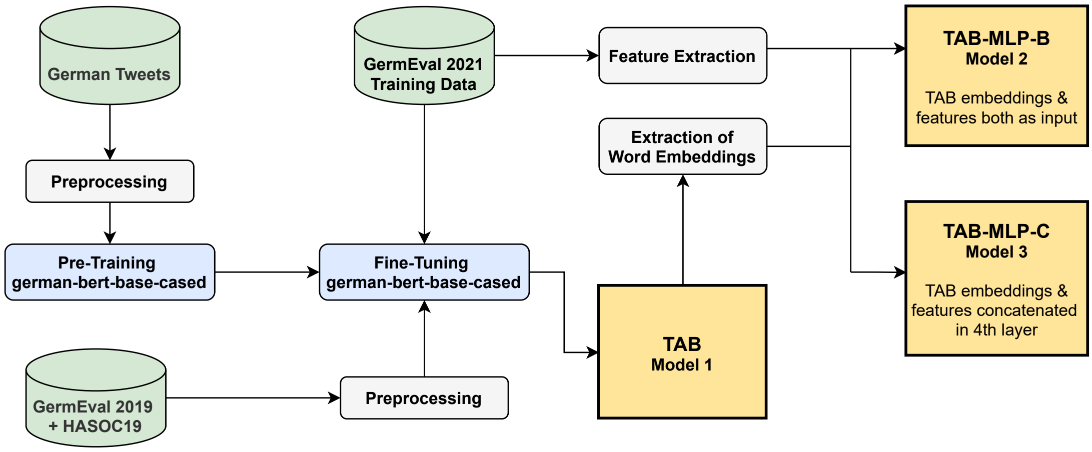

German Tweets
Preprocessing
Pre-Training german-bert-base-cased
GermEval 2021 Training Data
Fine-Tuning german-bert-base-cased
GermEval 2019 + HASOC19
Preprocessing
Feature Extraction
Extraction of Word Embeddings
TAB Model 1
TAB-MLP-B Model 2
TAB embeddings & features both as input
TAB-MLP-C Model 3
TAB embeddings & features concatenated in 4th layer

Figure 1: Experimental setup for training our submitted models (Green: datasets; grey: processing steps; blue: transformer re-training & fine-tuning steps; yellow: final models).

on the GermEval 2021 training data, as well as the additional datasets (GermEval 2019 & HASOC 2019). The augmented dataset contained a total of 24,304 comments, where 5,414 we set as toxic and 18,890 as not toxic as described in section 3. However, we added one more preprocessing step, compared to the transformer baselines, for pre-training and fine-tuning our model, since the authors of the GermEval 2021 changed every username in the comments to ”@USER”. We applied this to the additional German tweets as well as to the GermEval 2019 and HASOC 2019 datasets to align all texts. For the evaluation of our model, we used 10% of the GermEval 2021 training dataset. Our final transformer model, called TAB (tweets-and-Additional-Datasets-BERT) was trained on this augmented data for 10 epochs, a batch size of 16, a learning rate of 2e-5, Adam as an optimizer, and a maximum sequence length of 256.

- **Multi-Layer Perceptron (TAB-MLP):** For our second and third run, we used the MLP model we created for the baseline. Its architecture consists of 5 dense layers, a dropout of 0.2, ReLU (Rectified Linear Unit) as an activation function and sigmoid for our final classification layer. Since the FastText embeddings seemed to overfit the model, we extracted the already fine-tuned word embeddings of the TAB model via the [CLS] token of each input. Lastly, the additional extracted features were normalized and used for two different training strategies:

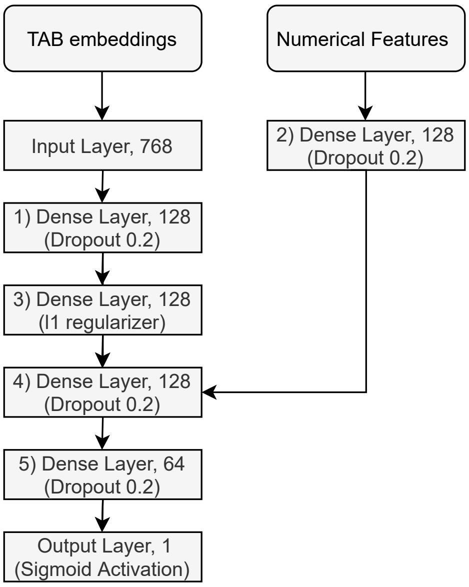

TAB embeddings
Numerical Features
Input Layer, 768
2) Dense Layer, 128 (Dropout 0.2)
1) Dense Layer, 128 (Dropout 0.2)
3) Dense Layer, 128 (l1 regularizer)
4) Dense Layer, 128 (Dropout 0.2)
5) Dense Layer, 64 (Dropout 0.2)
Output Layer, 1 (Sigmoid Activation)

Figure 2: Architecture of TAB-MLP-C.

- *TAB-MLP-B:* the model was fed with the text input as well as the features combined as one input vector for training.
- *TAB-MLP-C:* the model was trained on the textual input for 3 layers, the numerical features for 1 layer, and then concatenated in the 4th layer as shown in Figure 2.

Both models were trained for 25 epochs, a batch size of 32, a learning rate of 1e-2, and Stochastic Gradient Descent (SGD) as an optimizer. After plotting the curves of the evaluation metrics and comparing them with the FastText embeddings (Table 3) we found that the MLP did not seem to

Model	Run	Val Precision	Val Recall	Val F1	T Precision	T Recall	T F1
TAB	1	0.74	0.68	0.68	**0.6306**	**0.5535**	**0.5895**
TAB-MLP-C	2	**0.67**	**0.94**	**0.78**	0.3622	0.3597	0.3572
TAB-MLP-B	3	0.65	0.98	0.78	0.3854	0.3771	0.3812

Table 4: Results of our proposed models on the validation (Val) split of the training set and the test data (T).

Model	TP	TN	FP	FN
TAB	61	554	40	289
TAB-MLP-B	144	180	414	206
TAB-MLP-C	122	241	353	228

Table 5: Confusion matrix for each of our submitted models (TP: true positives, TN: true negatives, FP: false positives, FN: false negatives).

overfit with the already pre-trained TAB embeddings. Since we used a sigmoid activation function in our classification layer, we set a threshold for the predictions on the test set at 0.7, after calculating the mean and median value for each of our neural networks.

5 Results and Discussion

All of our models were evaluated with precision, recall, and a macro-averaged F1-score as shown in Table 4. The final results on the test data show that the transformer model gained by far the best results with its F1-score of 0.5895, even if it is still not as high as the value we expected after our training validation. Our neural networks TAB-MLP-B and TAB-MLP-C performed significantly worse on the test data, especially with regard to their high F1-score on the validation split.

Therefore, we explored whether we set the threshold too high for our predictions on the test data. Even though we experimented with setting the threshold to different values, we found that the predictions did not improve significantly (only $\approx$ 0.01), which shows that the neural networks probably overfitted on one class. We suspect this is also the reason for the very high validation recall in comparison to the precision. We plotted the confusion matrix for each model, shown in Table 5, which shows that both neural networks had a high count of false positives. In contrast to that, TAB had an issue with the false negatives. Therefore, we conclude several possible reasons why our neural networks did not perform well on the test set:

- the size of the dense layers, type of activation function and dropout have to be adjusted.
- the additional features have no positive impact on the models.
- another embedding strategy for the transformer models carries more information than the extraction of the [CLS] token. A possible solution could be a concatenation of a number of hidden layer outputs.

6 Conclusion

In this work we presented our submitted models for the GermEval Shared Task 2021 on toxic comment classification. We decided to combine standard supervised methods with transformers and textual features, and to enhance the models with additional training data.

Our best model was a German BERT that was re-trained on over 1.5 million additional German tweets from the first half year of 2019 and fine-tuned with two augmented datasets from similar tasks, such as hate speech and offensive language detection, as well as the GermEval 2021 training data. Even though our two multilayer perceptrons - which were trained on the extracted word embeddings by our transformer - showed better evaluation results during validation, our BERT model still had a more robust prediction on the test set. For future work, we will further explore the combination of sequential neural networks and word embeddings by transformers and test several extraction and concatenation strategies.

7 Acknowledgements

This work is enhanced by the Darmstadt University of Applied Sciences in collaboration with the Fraunhofer Institute for Secure Information Technology. The Darmstadt University supported this work with the research in Information Science (https://sis.h-da.de/). Additionally, this contribution has been funded by the project ”DeTox” (Cybersecurity research funding of the Hessian Ministry of the Interior and Sports).

References

Betty van Aken, Julian Risch, Ralf Krestel, and Alexander Löser. 2018. Challenges for toxic comment classification: An in-depth error analysis. In *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*, pages 33–42, Brussels, Belgium. Association for Computational Linguistics.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Branden Chan, Timo Möller, Malte Pietsch, and Tanay Soni. HuggingFace German BERT. Accessed: 2021-06-10.

Julien Chaumond. HuggingFace German DistilBERT. Accessed: 2021-06-10.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Ashwin Geet D'Sa, Irina Illina, and Dominique Fohr. 2020. BERT and fastText embeddings for automatic detection of toxic speech. In *2020 International Multi-Conference on: "Organization of Knowledge and Advanced Technologies" (OCTA)*, pages 1–5.

Spiros V. Georgakopoulos, Sotiris K. Tasoulis, Aristidis G. Vrahatis, and Vassilis P. Plagianakos. 2018. Convolutional neural networks for toxic comment classification. In *Proceedings of the 10th Hellenic Conference on Artificial Intelligence*, SETN '18, New York, NY, USA. Association for Computing Machinery.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Teuvo Kohonen. 1997. Learning vector quantization. In *Self-Organizing Maps*, pages 203–217. Springer Berlin Heidelberg.

Ritesh Kumar, Atul Kr. Ojha, Shervin Malmasi, and Marcos Zampieri. 2020. Evaluating aggression identification in social media. In *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, pages 1–5, Marseille, France. European Language Resources Association (ELRA).

Thomas Mandl, Sandip Modha, Prasenjit Majumder, Daksh Patel, Mohana Dave, Chintak Mandlia, and Aditya Patel. 2019. Overview of the HASOC track at FIRE 2019: Hate speech and offensive content identification in indo-european languages. In *Proceedings of the 11th Forum for Information Retrieval Evaluation*, pages 14–17.

Viera Maslej-Krešňáková, Martin Sarnovský, Peter Butka, and Kristína Machová. 2020. Comparison of deep learning models and various text preprocessing techniques for the toxic comments classification. *Applied Sciences*, 10(23).

Julian Risch, Anke Stoll, Lena Wilms, and Michael Wiegand. 2021. 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 co-located with KONVENS*, pages 1–12.

David Robinson, Ziqi Zhang, and Jonathan Tepper. 2018. Hate speech detection on twitter: Feature engineering v.s. feature selection. In *The Semantic Web: ESWC 2018 Satellite Events*, pages 46–49, Cham. Springer International Publishing.

Francisco Rodríguez-Sánchez, Jorge Carrillo-de Albornoz, and Laura Plaza. 2020. Automatic classification of sexism in social networks: An empirical study on Twitter data. *IEEE Access*, 8:219563–219576.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Petra Schneider, Michael Biehl, and Barbara Hammer. 2009. Distance learning in discriminative vector quantization. *Neural Computation*, 21(10):2942–2969.

Mina Schütz, Jaqueline Boeck, Daria Liakhovets, Djordje Slijepčević, Armin Kirchknopf, Manuel Hecht, Johannes Bogensperger, Sven Schlarb, Alexander Schindler, and Matthias Zeppelzauer. 2021. Automatic sexism detection with multilingual transformer models. *arXiv preprint arXiv:2106.04908*.

Hajung Sohn and Hyunju Lee. 2019. MC-BERT4HATE: Hate speech detection using multi-channel BERT for different languages and translations. In *2019 International Conference on Data Mining Workshops (ICDMW)*, pages 551–559.

Julia Maria Struß, Melanie Siegel, Josep Ruppenhofer, Michael Wiegand, and Manfred Klenner. 2019. Overview of GermEval task 2, 2019 shared task on the identification of offensive language. In

Proceedings of the 15th Conference on Natural Language Processing (KONVENS 2019), pages 354–365, Erlangen, Germany. German Society for Computational Linguistics & Language Technology.

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

Svitlana Volkova and Jin Yea Jang. 2018. Misleading or falsification: Inferring deceptive strategies and types in online news and social media. In *Companion Proceedings of the The Web Conference 2018*, WWW '18, page 575–583, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Hajime Watanabe, Mondher Bouazizi, and Tomoaki Ohtsuki. 2018. Hate speech on Twitter: A pragmatic approach to collect hateful and offensive expressions and perform hate speech detection. *IEEE Access*, 6:13825–13835.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Marcos Zampieri, Preslav Nakov, Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Hamdy Mubarak, Leon Derczynski, Zeses Pitenis, and Çağrı Çöltekin. 2020. Semeval-2020 task 12: Multilingual offensive language identification in social media (offenseval 2020). *arXiv preprint arXiv:2006.07235*.

Zhixue Zhao, Ziqi Zhang, and Frank Hopfgartner. 2021. *A Comparative Study of Using Pre-Trained Language Models for Toxic Comment Classification*, page 500–507. Association for Computing Machinery, New York, NY, USA.

Universität Regensburg MaxS at GermEval 2021 Task 1: Synthetic Data in Toxic Comment Classification

Maximilian Schmidhuber
Universität Regensburg / Fakultät für Sprache, Literatur und Kultur
maximilian.schmidhuber@stud.uni-regensburg.de

Abstract

We report on our submission to Task **1** of the GermEval 2021 challenge – toxic comment classification. We investigate different ways of bolstering scarce training data to improve off-the-shelf model performance on a toxic comment classification task. To help address the limitations of a small dataset, we use data synthetically generated by a German GPT-2 model.

The use of *synthetic data* has only recently been taking off as a possible solution to addressing training data sparseness in NLP, and initial results are promising. However, our model did not see measurable improvement through the use of synthetic data. We discuss possible reasons for this finding and explore future works in the field.

1 Introduction

In recent years, social media platforms have become an integral part of our everyday lives. Together with their enormous rise in use and popularity, they have also faced several troubles. These range from PR problems due to privacy concerns[1] to Fake News (Wells et al., 2019). There have also been incidents related to deplatforming controversial individuals of public interest[2].
In 2018, for instance, Facebook was used to incite a Genocide against the Rohingya people of Myanmar[3]. The ongoing global pandemic has seen an increase in xenophobic and antisemitic hate (Greenblatt, 2020). In the light of these and other developments, the task of detecting toxicity on the internet has seen increased attention in recent years. By now, the legislature of multiple countries is getting modified to accommodate laws counteracting the incitement of hatred online.
The German *Netzwerkdurchsetzungsgesetz* (network enforcement act), for instance, requires social media providers with over 2 million users registered in Germany to report hate speech on their platform to legal authorities[4]. These steps try to prevent the marginalization of populations. Users exposed to hate online may no longer take part in debates or discourse. This work aims to advance the state of the art in the field of toxicity detection by providing an additional avenue of working with scarce data. Any code used for this work is available on GitHub[5].

2 Related Work

There have been numerous developments in the space of toxicity detection since GermEval in 2019. Struß et al. (2019) give a well-composed overview of the state of affairs in 2019. We will therefore focus on recent work and concepts closely related to the challenge.

2.1 Recent Developments in Toxicity Detection

In the scope of the Shared Task, the concept of toxicity includes *"uncivil forms of communication that can violate the rules of polite behaviour, such as insulting discussion participants, using vulgar or sarcastic language or implied volume via capital letters"* (Risch et al., 2021). A more thorough definition of the annotation guidelines is provided by Risch et al. (2021).

In terms of 'traditional' hate speech detection, Mathew et al. (2020) proposed HateXplain. HateXplain is a new benchmark dataset for hate speech

[1] https://www.wired.com/story/facebook-privacy-ftc-changes/

[2] https://www.theatlantic.com/ideas/archive/2021/05/facebooks-trump-ban-effects/618818/

[3] https://www.nytimes.com/2018/10/15/technology/myanmar-facebook-genocide.html

[4] https://www.bmjv.de/DE/Themen/FokusThemen/NetzDG/NetzDG_node.html

[5] https://github.com/khaliso/GermEval2021_submission

detection, which tries to factor in hate speech bias and interpretability aspects.

Another notable contribution was made by Rosenthal et al. (2020). They created a new dataset called *SOLID* using a semi-supervised learning approach using an ensemble of four different models. It is the largest available dataset in the field right now.

A recent work provided by Sheth et al. (2021) notes that context is of key importance for toxicity detection. As surrounding conversation would mitigate a potentially toxic comment, exchange history would inform the determination of toxicity. Therefore, an exchange history surrounding the potentially toxic comment is needed in the corpus. Sheth et al. (2021) also note a potential problem with current transformer-based state-of-the-art systems such as BERT and GPT-2/GPT-3 (**G**enerative **P**retrained **T**ransformer). These models are designed to predict the next token given previous tokens from the dataset they were trained on. As these datasets have been collected from the web, corpus bias and incidentally confounded features can result in models that may cause harm to individuals or society (Kursuncu et al., 2020; McGuffie and Newhouse, 2020). There are indications that BERT embeddings may have racist or toxic tendencies (Zhang et al., 2020). Solaiman et al. (2019) of *OpenAI* note that GPT-2 is capable of producing extremist text if trained on suitable data. However, machine-generated content detection tools such as *Grover* by Zellers et al. (2019) can spot GPT-2 generated content in most cases. They also note that *the skills and resources required for using language models, both beneficially and maliciously, will decrease over time* (Solaiman et al., 2019). GPT-3 by Brown et al. (2020) does develop in this direction.

2.2 Synthetic Data

Shu et al. (2020) note that limited labelled data is becoming the largest bottleneck for supervised learning systems. This is especially the case for many real-world tasks where large scale annotated examples can be too expensive to acquire. Therefore, they proposed a technique using semi-supervised learning; however, there have also been different approaches to face this task. For example, GPT-2 (Radford et al., 2019) is a Text Generation model created by *OpenAI* using the transformers architecture. Both GPT-2 and, more prominently, its successor GPT-3 by Brown et al. (2020) are most well-known for their ability to create text that is almost indistinguishable from text written by humans. Moreover, as Budzianowski and Vulić (2019) found, the model also *holds promise to mitigate the data scarcity problem*. With these recent advancements on the horizon, interest in Synthetic Data Generation has grown in many areas of research, including NLP. Works include generating synthetic data for Lexical Normalization (Dekker and van der Goot, 2020) or Neural Grammatical Error Correction Systems (Grundkiewicz et al., 2019). Recent works in the field of Toxic Comment Classification appear to be very promising, but this particular field of research is still very young (Juuti et al., 2020; Whitfield, 2021).

Synthetic data has implications besides its potential use in bolstering datasets: This approach poses a possible solution to ethical, security and privacy concerns with real datasets (Surendra and Mohan, 2017).

2.3 Related Challenges

In terms of state-of-the-art systems, other recent Shared Tasks can give a good overview:

1. GermEval-2019 Task 2: Identification of offensive language (Struß et al., 2019)
2. Kaggle, 2020: Jigsaw Multilingual Toxic Comment Classification[6]
3. SemEval-2020 Task 12: Multilingual offensive language identification in social media (Zampieri et al., 2020; Ranasinghe and Hettiarachchi, 2020)
4. SemEval 2021 Task 5: Toxic Spans Detection was more fine-grained than previous tasks, as participants in this shared task were asked to determine which span(s) of text in a post were responsible for the classification of the entire post (Pavlopoulos et al., 2021)

The best-performing systems in the field of toxicity detection online as found by Zampieri et al. (2020) were mainly based on XLM-RoBERTa, ALBERT or ERNIE 2.0 (Safaya et al., 2020). These are among the newest iterations of BERT-based models.

However, to train and fine-tune such models, large quantities of - preferably labelled - data are required.

[6] https://www.kaggle.com/c/jigsaw-multilingual-toxic-comment-classification/

3 Methodology

In this work, we investigate whether or not synthetically generated data can improve the baseline of a model solely trained on a scarce dataset.

3.1 Data analysis

The Dataset used for this task was provided by Risch et al. (2021) and consists of:

- 3244 Facebook comments with binary labels for each of the three tasks at hand. This work focusses on **Task 1:** *Toxic Comment Classification.*
- 2122 of the comments were labelled as 'non-toxic', and
- 1122 were labelled 'toxic'.
- The longest toxic comment had a length of 2035 tokens;
- The longest non-toxic comment had 2833 tokens.

The Dataset is drawn from the Facebook page of a political talk show of a German television broadcaster and includes user discussions from February-July 2019.

The Dataset is anonymized by not sharing comment IDs and user information. Furthermore, links to users are replaced by @USER and links to the show replaced with @MEDIUM. Links to the show's moderator are replaced with @MODERATOR. This raises a couple of challenges:

1. The **first** challenge is the anonymized nature of the dataset, making it impossible to take context into account (Sheth et al., 2021).
2. The **second** challenge we face is the relatively small amount of data available. This issue is well-known (Shu et al., 2020) and not exclusive to the field of NLP.

The **first** challenge, lack of context. A rudimentary form of context can be constructed, as the dataset provides the *@USER*, *@MODERATOR* and *@MEDIUM* tags. It is, however, not possible to extract more fine-grained types of relationships between commenters. For example, a comment determined to be potentially toxic can be both harmless bickering within common groups and a toxic remark to outsiders. To more accurately determine the correct label, relationship data is required (Sheth et al., 2021). However, supplying the data necessary to create these relationships would raise severe privacy implications.

Therefore, the main variable we are ethically and technically able to influence is the **second** challenge, the size of the dataset. There have been several ways in the past to bolster a dataset.

3.2 Bolstering the Dataset

The standard approach, even in the history of the GermEval shared task, is to use additional, related datasets (Paraschiv and Cercel, 2019). There are a number of options:

1. **GermEval 2018 Dataset** by Wiegand et al. (2018) contains 8541 labeled offensive German tweets with an inter-author agreement of k = 0.66.
2. **GermEval 2019 Dataset** by Struß et al. (2019) contains two separate datasets. The training Dataset for Task 1 and 2 (binary and fine-grained classification) consists of 3995 annotated German offensive tweets, while the Dataset provided for Task 3 (explicit or implicit offensive language classification) consists of 1958 annotated German tweets
3. **German Federal Election Dataset** by Kratzke (2017), containing 1.212.220 unlabeled tweets crawled around the German Federal Election 2017
4. **OLID** by Zampieri et al. (2019) contains over 14.000 labeled English tweets
5. **SOLID** by Rosenthal et al. (2020) contains over 9 million English tweets labeled in a semi-supervised manner

The main issue we face is that the most available datasets for toxicity detection online are English.

Another method of bolstering datasets has been explored for the task of object detection on images by augmenting available data by rotating images or similar modification methods (Zoph et al., 2020). A comparable approach for the field of NLP is the [MASK] token used by Devlin et al. (2018) for BERT, the current gold-standard model for a wide variety of NLP tasks.

> ’Ziemlich traurig, das ganze Nachrichten zu einem zweiten Geburtstag.(1) The President of the Federal Republic of Germany shall be elected for a four-year term by the Bundestag on the basis of proportional representation by direct universal suffrage. (2) Die Bundesrat der Bundesregierung ist ein Bundesgesetz über eine gesetzliche Verfassungsgerichtshof, die durch die Bundesversammlung ausgeführt werden, soweit sie in dem Bundesverwaltung des Bundesministeriums und des Landesministers aufgehoben wird. Diese Fähigkeit wurde darüber häufig zur Verwandteilung der Fachberechtigung eines Bundesrates’

Figure 1: Output of GPT-2 fine-tuned on toxic comments from the dataset

With advancements in text generation models, another avenue of research has opened up: Synthetic Data.

3.3 Selecting the Data Generation Model

An initial evaluation, as seen in Figure 1, revealed that text generated by a fine-tuned GPT-2 included both English and German phrases. Fine-tuned German GPT-2 (gGPT-2) by Schweter (2020) on the other hand created German-sounding, yet incoherent sentences (Figure 2). gGPT-2 was fine-tuned on the normalized version of *Faust I and II* by Johann Wolfgang von Goethe. The model has not yet been used in the reviewed literature. However, initial experiments on German recipes[7] and German medical reviews appeared promising[8]. The initial reasoning was that the more German-sounding text created by gGPT-2 could benefit the trained system, as coherence might be less relevant on the token level if the readable text was also part of the finished dataset.

GPT-3 was not an option for this work, as we did not get access to the API in time.

3.4 Data Generation Model Description

The German GPT-2 Model is part of Huggingface’s *transformers* library. We selected a batch size of 16, and set the maximum sequence length to 1024. Similar to the approach used by Whitfield (2021), we fine-tuned two distinct models. Model 1 was fine-tuned on the data labelled as ’toxic’ and model 2 on the ’non-toxic’ data. Similar to the training data, we generated 2000 non-toxic synthetic comments and 1000 synthetic toxic comments. The synthetic comments were then merged with the original dataset. The train/test split was set to 80/20.

> ’Woche in die Sozial für Deutschen müssig wicht in eingeword und dann sich abende sind kontraum den Zu vällen. Diese vorlose Ihre wir die Viel einschauen dann nicht geworden. Und wurde vollwerk viel von Menschen die Grünen darf und dahler Ihr einmal das einmal in wollen in um dann nicht, noch mal schleiner ist die’

Figure 2: Output of German GPT-2 (Schweter, 2020) fine-tuned on toxic comments from the dataset

3.5 Classification Model Description

For the binary classification task at hand here, *BERT Multilingual Cased* was selected, as it is the gold standard for non-English NLP tasks (Miranda-Escalada et al., 2020; Keung et al., 2020). The focus of this work was not to create a top-performing system but to investigate the effects of fine-tuning using synthetic data. It is likely that more robust results could be achieved using one of the models mentioned previously.

A traint/test split of 80/20 was applied, and the model was trained over 5 epochs. We set the batch size to 6 and the maximum sequence length to 192. The *Adam* optimizer was used. As seen in Table 1, an initial test comparing an mBERT model trained solely on original data and another mBERT model trained on the merged dataset appeared promising.

4 Results and Discussion

As seen in Table 2, the validation results were not replicable on the test data. Therefore, the test results imply no measurable impact on the system’s effectiveness through synthetic data generated by the used methodology. In the light of these results, a couple of methodological mistakes need to be addressed.

First, the discrepancy in the validation and testing performance of the model using synthetic data is possibly due to the initial validation results being achieved on data that included synthetic data. This could be problematic, as the data generated by

[7] https://towardsdatascience.com/fine-tune-a-non-english-gpt-2-model-with-huggingface-9acc2dc7635b

[8] https://data-dive.com/finetune-German-gpt2-on-tpu-transformers-tensorflow-for-text-generation-of-reviews

Model	F1	Precision	Recall
mBERT[9]	0.651	0.667	0.645
mBERT - synthetic[10]	0.766	0.772	0.761

Table 1: Initial validation results of an mBERT model fine-tuned using the base dataset and another fine-tuned on the merged dataset

Model	F1	Precision	Recall
mBERT[11]	0.618	0.635	0.599
mBERT - synthetic[12]	0.615	0.623	0.608

Table 2: Results of both models when applied on the test data

gGPT-2 could have caused artificially high testing results. The testing data must be composed solely of original data to avoid potential impacts created by gGPT - 2 on testing results Therefore, future investigations will only use original data for validation testing.

Another issue is the selected Data Generation Model, gGPT-2. The output of GPT-2, as seen in Figure 1, is composed of both English and German sentences. This composition is not the desired outcome, but English and German are sister languages. The output generated by gGPT-2, as seen in Figure 2, on the other hand, appears to be effective on the token level, but in some cases not capable of generating coherent sentences or words. Therefore, we should have selected GPT-2 over gGPT-2.

In light of these mishaps, we still deem the approach of using synthetic data to be successful. Synthetic Data is comparatively easy, cheap and fast to use. It did not negatively affect the baseline approach.

Furthermore, ethical and practical implications of Synthetic Data are major issues. Privacy concerns of real datasets can be mitigated if Synthetic Data can be generated using real-world datasets that can not be published themselves.

5 Conclusion and Future Work

We conclude that synthetic data does seem to be a promising avenue of research. However, this particular work did not find a measurable improvement over the baseline model using synthetic data. Future work will use either GPT-3 or GPT-2, and we will rework the methodology. Once a more robust method is formulated, we will test it on several different datasets. We suggest further research on Synthetic Data in classification tasks, as the comparatively poor performance of our model may be due to the limitations of our chosen methodology. Other possible future avenues of research include the training and evaluation of a model solely trained on synthetic data.

References

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

Paweł Budzianowski and Ivan Vulić. 2019. Hello, it's GPT-2–how can I help you? towards the use of pretrained language models for task-oriented dialogue systems. *arXiv preprint arXiv:1907.05774*.

Kelly Dekker and Rob van der Goot. 2020. Synthetic data for English lexical normalization: How close can we get to manually annotated data? In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6300–6309, Marseille, France. European Language Resources Association.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Jonathan A Greenblatt. 2020. Fighting Hate in the Era of Coronavirus. *Horizons: Journal of International Relations and Sustainable Development*, (17):208–221.

Roman Grundkiewicz, Marcin Junczys-Dowmunt, and Kenneth Heafield. 2019. Neural grammatical error correction systems with unsupervised pre-training on synthetic data. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 252–263.

Mika Juuti, Tommi Gröndahl, Adrian Flanagan, and N Asokan. 2020. A little goes a long way: Improving toxic language classification despite data scarcity. *arXiv preprint arXiv:2009.12344*.

Phillip Keung, Yichao Lu, György Szarvas, and Noah A Smith. 2020. The multilingual Amazon reviews corpus. *arXiv preprint arXiv:2010.02573*.

Nane Kratzke. 2017. The# btw17 Twitter dataset–recorded tweets of the federal election campaigns of 2017 for the 19th German Bundestag. *Data*, 2(4):34.

Ugur Kursuncu, Yelena Mejova, Jeremy Blackburn, and Amit Sheth. 2020. Cyber social threats 2020 workshop meta-report: Covid-19, challenges, methodological and ethical considerations.

Binny Mathew, Punyajoy Saha, Seid Muhie Yimam, Chris Biemann, Pawan Goyal, and Animesh Mukherjee. 2020. Hatexplain: A benchmark dataset for explainable hate speech detection. *arXiv preprint arXiv:2012.10289*.

Kris McGuffie and Alex Newhouse. 2020. The radicalization risks of GPT-3 and advanced neural language models. *arXiv preprint arXiv:2009.06807*.

Antonio Miranda-Escalada, Aitor Gonzalez-Agirre, Jordi Armengol-Estapé, and Martin Krallinger. 2020. Overview of Automatic Clinical Coding: Annotations, Guidelines, and Solutions for non-English Clinical Cases at CodiEsp Track of CLEF eHealth 2020. In *CLEF (Working Notes)*.

Andrei Paraschiv and Dumitru-Clementin Cercel. 2019. UPB at GermEval-2019 Task 2: BERT-Based Offensive Language Classification of German Tweets. In *Proceedings of the 15th Conference on Natural Language Processing (KONVENS 2019)*.

John Pavlopoulos, Léo Laugier, Jeffrey Sorensen, and Ion Androutsopoulos. 2021. Semeval-2021 task 5: Toxic spans detection. *Proceedings of SemEval*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Tharindu Ranasinghe and Hansi Hettiarachchi. 2020. BRUMS at SemEval-2020 task 12: Transformer based multilingual offensive language identification in social media. *arXiv preprint arXiv:2010.06278*.

Julian Risch, Anke Stoll, Lena Wilms, and Michael Wiegand. 2021. 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 co-located with KONVENS*, pages 1–12.

Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Marcos Zampieri, and Preslav Nakov. 2020. A large-scale semi-supervised dataset for offensive language identification. *arXiv preprint arXiv:2004.14454*.

Ali Safaya, Moutasem Abdullatif, and Deniz Yuret. 2020. Kuisail at semeval-2020 task 12: Bert-cnn for offensive speech identification in social media. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 2054–2059.

Stefan Schweter. 2020. German GPT-2 model.

Amit Sheth, Valerie L Shalin, and Ugur Kursuncu. 2021. Defining and Detecting Toxicity on Social Media: Context and Knowledge are Key. *arXiv preprint arXiv:2104.10788*.

Kai Shu, Suhang Wang, Dongwon Lee, and Huan Liu. 2020. Mining disinformation and fake news: concepts, methods, and recent advancements. In *Disinformation, Misinformation, and Fake News in Social Media*, pages 1–19. Springer.

Irene Solaiman, Miles Brundage, Jack Clark, Amanda Askell, Ariel Herbert-Voss, Jeff Wu, Alec Radford, Gretchen Krueger, Jong Wook Kim, Sarah Kreps, et al. 2019. Release strategies and the social impacts of language models. *arXiv preprint arXiv:1908.09203*.

Julia Maria Struß, Melanie Siegel, Josef Ruppenhofer, Michael Wiegand, Manfred Klenner, et al. 2019. Overview of germeval task 2, 2019 shared task on the identification of offensive language. In *Proceedings of the 15th Conference on Natural Language Processing (KONVENS 2019)*, pages 354–365. sa.

HMHS Surendra and HS Mohan. 2017. A review of synthetic data generation methods for privacy preserving data publishing. *International Journal of Scientific & Technology Research*, 6(3):95–101.

John R Wells, Carola A Winkler, and Carole A Winkler. 2019. *Facebook fake news in the post-truth world*. Harvard Business Publishing Education, September 14.

Dewayne Whitfield. 2021. Using gpt-2 to create synthetic data to improve the prediction performance of nlp machine learning classification models. *arXiv preprint arXiv:2104.10658*.

Michael Wiegand, Melanie Siegel, and Josef Ruppenhofer. 2018. Overview of the germeval 2018 shared task on the identification of offensive language. In *Proceedings of GermEval 2018, 14th Conference on Natural Language Processing (KONVENS 2018)*, pages 1–10.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. Predicting the type and target of offensive posts in social media. *arXiv preprint arXiv:1902.09666*.

Marcos Zampieri, Preslav Nakov, Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Hamdy Mubarak, Leon Derczynski, Zeses Pitenis, and Çağrı Çöltekin. 2020. SemEval-2020 task 12: Multilingual offensive language identification in social media (OffensEval 2020). *arXiv preprint arXiv:2006.07235*.

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. Defending against neural fake news. *arXiv preprint arXiv:1905.12616*.

Haoran Zhang, Amy X Lu, Mohamed Abdalla, Matthew McDermott, and Marzyeh Ghassemi. 2020. Hurtful words: quantifying biases in clinical contextual word embeddings. In *proceedings of the ACM Conference on Health, Inference, and Learning*, pages 110–120.

Barret Zoph, Ekin D Cubuk, Golnaz Ghiasi, Tsung-Yi Lin, Jonathon Shlens, and Quoc V Le. 2020. Learning data augmentation strategies for object detection. In *European Conference on Computer Vision*, pages 566–583. Springer.

TUW-Inf at GermEval2021: Rule-based and Hybrid Methods for Detecting Toxic, Engaging, and Fact-Claiming Comments

Kinga Gémes
TU Wien
kinga.gemes@tuwien.ac.at

Gábor Recski
TU Wien
gabor.recski@tuwien.ac.at

Abstract

This paper describes our methods submitted for the GermEval 2021 shared task on identifying toxic, engaging and fact-claiming comments in social media texts (Risch et al., 2021). We explore simple strategies for semi-automatic generation of rule-based systems with high precision and low recall, and use them to achieve slight overall improvements over a standard BERT-based classifier.

1 Introduction

We present our systems submitted to the GermEval 2021 shared task on identifying toxic, engaging and fact-claiming comments in social media texts (Risch et al., 2021). We focus on strategies for building simple rule-based systems that are both explainable and customizable for end users. We also train a simple BERT-based classifier for comparison, and to evaluate its performance when combined with our high-precision rule-based systems. After a short overview in Section 2 of the task and the datasets used we describe our methods for creating rule-based systems in Section 4 and the BERT-based baseline system in Section 3. Section 5 describes how these systems were combined into simple ensemble models, Section 6 presents quantitative results on the 2021 test set, and a manual qualitative analysis on a sample of our output for Subtask 1 is provided in Section 7. All systems described in this paper are publicly available under an MIT license from the repository `tuw-inf-germeval2021`[1], along with instructions for reproducing our results.

2 Task and datasets

The dataset of the 2021 shared task contains 3,244 comments from the Facebook page of a German news broadcast, from discussions between February and July 2019, manually annotated for three categories corresponding to the three subtasks: whether a comment is toxic, engaging, and/or fact-claiming. Definitions for each category and a detailed description of the annotation process are given in the task overview paper (Risch et al., 2021). For developing our rule-based system for toxicity detection we also used a corpus of annotated tweets from Germeval challenges of previous years (Wiegand et al., 2018; Struß et al., 2019), the 2018 and 2019 datasets contain nearly 11,000 German tweets.

Comments in the 2021 dataset were parts of discussion threads related to individual news items. The dataset does not contain such threads, only individual comments, and this is also how they were presented to annotators. However, some fragments of the initial posts ('teaser texts') were made available to annotators as context, but were not included in the dataset because of privacy concerns (Wilms, 2021). This means that in some cases our models may not have had access to the full information that led annotators to their decisions. Some possible examples will be presented in the manual analysis in Section 7. When experimenting with our methods, we split the 3,244 comments of the training dataset into two parts, training our ML models and developing our rules using only 2,434 comments (75 %) and validating our approach on the remaining 811 (25 %).

Some entities in the dataset have been anonymized by the organizers, introducing

[1] `https://github.com/GKingA/tuw-inf-germeval2021`

placeholders such as *@USER*, *@MEDIUM*, and *@MODERATOR*. In addition we also masked URLs, currency symbols, and numbers. For our BERT-based experiments we also replaced emoticons with their German textual representations, using the `emoji` library[2]. A German dictionary has been added to this library only days before the submission deadline. In our submissions we use our own dictionary[3], created from the English resource using the Google Translate API via the `translate` Python library[4].

3 BERT-based classification

Language models based on the Transformer architecture (Vaswani et al., 2017) such as BERT (Devlin et al., 2019) provide the basis of strong baseline systems across a wide range of tasks in natural language processing, and some of the top-performing systems in the 2019 GermEval challenge also use BERT (Paraschiv and Cercel, 2019; Graf and Salini, 2019). For our experiments we used the model `bert-base-german-cased`[5] a publicly available BERT model trained only on German data. For each subtask we trained a neural network with a single linear classification layer on top of BERT. Metaparameters were set based on performance on the validation set. We used Adam optimizer with a weight decay value of 10^{-5} and initial learning rate of 10^{-5}. We set batch size to 8 and trained each model for 10 epochs and determined the optimal number of iterations based on either F-score or precision on the validation set (see Section 5 for details). For the final submissions we trained on the full training set (including the validation set).

4 Rule-based methods

We explore simple strategies for both manual and semi-automatic generation of lists of words and phrases that can be used in rule-based systems that consider a comment toxic if and only if it contains any of the words or phrases in a list. Our goal is to facilitate the rapid creation of such simple rule-based systems because they are often preferred in real-world applications due to their fully transparent and explainable nature. Any decision made by such a system, whether true or false, positive or negative, can be directly attributed to one or more terms in the input or to the fact that no terms in the input are present in the list of key terms. This offers straightforward ways for users to update the rules in a way that changes a particular decision, by removing keyphrases causing false positives or adding them to fix false negatives. Whether or not this process is actually beneficial for the overall accuracy of a model, it is in line with common business needs, most typically with the common experience that once users have reported an error, they expect it to be corrected. The experiments in this section and the qualitative analysis in Section 7 were performed only for the toxicity detection task.

For the toxicity detection task we experimented with simple strategies for automatic bootstrapping of keyword lists, which are then reviewed and corrected manually. The method involves extracting simple patterns from comments in the training data and ranking them according to their potential as rules, i.e. looking for patterns that in themselves have a very high precision as predictors of toxicity. We searched for patterns characteristic of comments labeled as toxic in the form of a few simple feature types, including unigrams and bigrams of words or lemmas, with or without part-of-speech tag for potential disambiguation purposes. We also tried limiting the space of unigram features to nouns only, or nouns and adjectives only. For tokenization, lemmatization, and part-of-speech tagging we use the `Stanza` library (Qi et al., 2020) using the `gsd` German model with `resources_version` 1.2.0. We achieved best results when limiting our search to word unigrams only and ranking them separately for nouns (including proper nouns) and for all other parts-of-speech.

To extract patterns with a high potential as rules we experimented with simple scoring schemes for ranking all features based on the number of true and false positives they would contribute if used as strict rules, i.e. the number of positive and negative examples containing them as patterns. To assess the efficiency of these strategies, i.e. whether the patterns they extract are generally good rule candidates that can be edited into curated lists, we observed their behaviour in the portion of the dataset

[2] `https://github.com/carpedm20/emoji`
[3] `https://github.com/GKingA/emoji`
[4] `https://pypi.org/project/translate/`
[5] `https://deepset.ai/german-bert`

used for training the ML models. The validation set was only used for infrequent overall quantitative evaluations and not for observing patterns, since for the purposes of manual rule creation this would have meant using the validation data for training. The strategy we found most effective was to consider patterns with at least 5 occurrences in the training dataset and rank them with the scoring scheme $TP - 100 \cdot FP$, where TP and FP are the number of true and false positives detected by that pattern.

Lists created this way require manual editing so as not to introduce artefacts. For example, the top words in the training dataset for this year's Germeval task contain words like *Hamburg* and *fleissig* 'hard-working' because these words happened to occur in several comments labelled as toxic but none of the non-toxic ones, thereby getting ranked just as high as *Dummheit* 'stupidity' and *Bullshit*, terms that we actually want to keep for the edited list. The majority of good patterns comes from the larger toxicity dataset available to us, the 2018 and 2019 Germeval training datasets (Wiegand et al., 2018; Struß et al., 2019). While in the smaller 2021 dataset the top-ranked patterns occurred in no more than 3 or 4 positive examples of toxicity, the combined training datasets of previous years allowed us to find patterns with 15-25 positive examples, a much stronger indicator that a word might be a good keyphrase for domain-independent detection of offensive speech. Indeed, the list of the 10 highest-ranked nouns barely need post-editing, they are *Vasall* 'vassal', *Invasoren* 'invaders', *Abschaum* 'scum', *Heuchler* 'howler', *Dumm* 'dumb', *Kreatur* 'creature', *Ficker* 'fucker', *Titten* 'boobs', *Scheisse* 'shit', *Volksverräterin* 'traitor of the people'. We note that emoji characters are also handled by stanza as individual words and some of them also appear in the final keyword lists, such as these characters: 🤮🤬🖕💩 . The extraction and ranking of patterns is implemented in the `ml` module of the `tuw-nlp` library, the manually curated rule lists and code to apply them are part of the `tuw-inf-germeval2021` repository.

In an independent effort we also used the 2021 training dataset for all three subtasks to observe simple patterns that can be used as high-precision predictors of each category. Two of the patterns we identified were introduced in the final rule-based system: for the toxicity detection task we categorize a comment as toxic if it contains at least two words with at least four characters each written in ALL-CAPS. This rule on its own achieved 91% precision and 4% recall on our validation set. For Subtask 3, if a comment contains an URL and this URL is not the only content of the comment, we categorize it as fact-claiming. This rule achieved 93% precision and 10% recall on the validation set.

5 Ensemble

Our three submissions for the shared task are combinations of the systems described in Sections 3 and 4. Our first submission contains the decisions of the BERT-based classifiers for each subtask, using the number of iterations determined as yielding the highest F-score (2nd for Subtask 1, 1st for Subtask 2, and 5th for Subtask 3). Submission 2 is the union of Submission 1 and our rule-based systems, i.e. for each subtask we label comments as toxic/fact-claiming if either the BERT-based model or our rule-based system would classify it as such (we did not use any rules for subtask 2). Finally, Submission 3 is our attempt at a system with higher precision at the cost of recall, here we use our rules together with BERT models from the iterations yielding the highest precision (8th for Subtask 1, 1st for Subtask 2, and 1st for Subtask 3). We note that this is different from training a machine learning model for high precision, which could be achieved by e.g. a weighted loss function. In case of Subtask 2, both precision and F-score were optimal after the same number of training epochs. Since we did not use any rules for detecting engaging comments, our output for this subtask was identical in all three submissions.

6 Quantitative results

Quantitative evaluation of our methods is performed based on the test set provided by the organizers. We follow the official evaluation methodology and calculate precision, recall, and F-score for both classes in each subtask and also the macro-average across classes for each figure. Results on toxicity detection (Subtask 1) are presented in Table 1. Our rule-based system did not achieve higher precision than the BERT-based system, but it selected a somewhat different set of comments and increased the recall and F-score of the ensemble system in Submission 2. We shall take a closer look into the contributions of the rule-based system as part of our qualitative analysis in Section 7. The BERT model chosen for high precision performed worse, possibly because of overfitting on the training dataset (it was trained for 8 epochs as opposed to the 2 epochs of the high F model).

For the task of detecting engaging comments (Subtask 2) we did not develop any rule-based system and the same BERT model was determined to be optimal for both precision and F-score, therefore we used the output from the same BERT model in all our submissions. Here we omit results for this subtask due to lack of space. Results on detecting fact-claiming comments (Subtask 3) are presented in Table 2. Although the rule-based system achieves high precision, the comments it identifies as fact-claiming (based on the single rule regarding URLs, see Section 4) form a subset of the comments identified as such by the BERT model, hence our labels for Submissions 1 and 2 are identical. The BERT model chosen for high precision indeed makes very few false positive decisions and can be slightly improved in terms of both precision and recall by adding labels from the rule-based system (Submission 3).

7 Qualitative analysis

The main focus of our rule-based experiments was the toxicity detection. We performed a detailed qualitative analysis on a sample of the test dataset on this subtask. Based on the labels assigned by the BERT model of Submission 1 and the ground truth labels we extracted a sample of 40 comments, 10 from each of the four categories true positive (TP), false positive (FP), true negative (TN), and false negative (FN). Our goal with this setup is to go beyond error analysis, which would only focus on false positives and false negatives. Given the subjectivity of the task and of the possible discrepancies between the information available to annotators and to our models (see Section 2), we wished to inspect a sample that is balanced across both predicted and ground truth labels. Since the BERT model of Submission 3 performed poorly, probably due to overfitting, for the purposes of this analysis we shall focus on the output of the BERT model of Submission 1 (trained for 2 epochs, for maximum F-score) and our rule-based system. The number of comments in each category for both of these systems and their combination is presented in Table 3. Figure 1 shows the comments in our samples of TP, FP, TN, and FN predictions of the BERT model, respectively. An asterisk (*) marks an agreement with the rule-based system, e.g. TP1 and TP2 were classified as toxic by both models and TP3-10 were classified as toxic by BERT but not by the rule-based system, and all ten have been labeled as toxic by the annotators (hence *true* positive).

The sample of true positives, as expected, contains many comments that are clearly identifiable as such based on surface patterns such as the word *dumm* 'stupid' (TP1, TP10) or the emojis 💩 and 👎 (TP2). False negatives (FN) would be expected to exhibit the opposite pattern, these are comments that humans agreed are toxic but models failed to detect them as such. Indeed this group contains several examples where a deep understanding of the comment is necessary to account for its toxicity, demonstrating the complexity of the task. For example, to understand that comment FN9 *Der Deutsche war schon immer naiv...* 'The Germans have always been naive' is in some way uncivil, one must know that some types of statements about some types of groups are not acceptable and at the same time be able to identify *Germans* and *naive* as concepts belonging to these 'types'. Indeed, if a human expert were to build a complex rule-based model for the toxicity detection task, it may very well contain patterns such as *PROTECTED_GROUP + NEGATIVE_PREDICATE* and lexica for what words and phrases are to be considered as belonging to each of these categories. Other examples require knowledge of idioms, such as the comment FN1 *Geht's noch?* which as a phrase can be translated as 'Are you crazy?'[6]. Perhaps the most puzzling examples of false negatives are those that were probably interpreted as sarcastic by annotators, such as FN2 *Good Luck Mr. President Trump 👍👏❤️ Make America Great Again. Ich würde ihn wählen.*

[6] https://de.wiktionary.org/wiki/geht%E2%80%99s_noch

TP1* Meine Güte redet die Dame Quatsch! Frauen sind also Feiglinge und Männer zu dumm . Aha

TP2* Putin ist sehr nerwös 😎💨💩😂👎

TP3 @USER Ja. Er ist blödsinnig.

TP4 Schon wieder ein klugscheiss!!

TP5 DANKE Carla super weiter so ❤️❤️❤️

TP6 Trump 👍🏽👍🏽👍🏽👍🏽🤣🤣 Biden 🇨🇳🇨🇳🇨🇳🇨🇳🇮🇷🇮🇷🇮🇷

TP7 Was war jetzt so schlimm in den letzten vier Jahren? Unter Trump bleibt die Welt friedlich. Gewinnt Biden, wirds Krieg geben. Guten Tipp an Zamperoni, verlass deine Frau!

TP8 😂🤣🤦‍♂️🤦‍♂️🤦‍♂️

TP9 Covid Karls Paniktour 😂😂😂😂

TP10 @USER seit ihr so naiv, oder tut ihr nur so dumm?????? 🎩🌹

FP1* @USER es gibt halt leider in jeder Altersgruppe asoziale Mitmenschen.

FP2* @USER Gott sei Dank gibt es eine Meinungsfreiheit und verschiedene Auffassungen. Und ganz nebenbei gefragt, was hat diese Dame denn für Vorschläge gemacht?

FP3* Frau ”Blaerbock” in ihrem Element. Fehlt nur noch eine Horde ”Kobolde”!

FP4 @USER stimmt schon, gegen den würde Reagen wie Bernie Sanders wirken.

FP5 pandemie? 🤣 97% der deutschen sind weder infiziert, noch krank!

FP6 Schnell viel Blödsinn reden...

FP7 Sitzt da die Schwester von Didi Hamann(Augenzucken)?

FP8 ...keine ahnung...woll?!? 😉

FP9 Dünnes Eis Frau Prof.

FP10 Tja, mit Ideologie wird es kalt und dunkel hier! 🤦‍♀️

TN1* @USER woher wissen Sie das? Glauben heißt nicht Wissen 🤓

TN2* Sie wissen schon dass dies in der Konsequenz vielen Menschen die sich schutzlos infizieren das Leben kosten wird ?

TN3* @USER paint it red

TN4* @USER weil er es kann. 🤷🏻‍♂️

TN5* @USER ne 🙈

TN6* @USER das ist leider auch wahr

TN7* Wer's glaubt 👏👍

TN8* @USER sehr guter Kommentar 👍

TN9* 👍 Trump 👍

TN10* Weiterhin gute Besserung!

FN1* Geht's noch?

FN2* Good Luck Mr. President Trump 👍👏❤️ Make America Great Again. Ich würde ihn wählen.

FN3* Tourett?

FN4* Politiker mit Verstand ?...und das in Zusammenhang mit dem Namen Trump ?

FN5* Und wenn ich mir die Frage stelle, ob ich einen Arbeitsplatz habe und die Miete bezahlen kann, dann wähle ich Biden? 😂😂😂

FN6* Trump wird gewinnen und das ist gut so. Auf die Gesichter der Hetzmedien bin ich gespannt

FN7* @USER immer politisch Korrekt verstehst Du 😉😉

FN8* @USER viel Meinung für wenig Ahnung!

FN9* @USER Der Deutsche war schon immer naiv, und sie sind es auch.

FN10* Mimimimi

Figure 1: Sample of 10 comments each from true positive (TP), false positive (FP), true negative (TN), and false negative (FN) labels predicted by the BERT model on the test set for Subtask 1. An asterisk (*) marks comments where the rule-based system made the same prediction as the BERT model

	Other			Toxic			Average		
	P	R	F	P	R	F	P	R	F
Rules	64.0	**98.3**	77.5	**67.7**	6.0	11.0	65.9	52.2	58.2
BERT (S1)	72.4	87.9	79.4	**67.7**	43.1	52.7	70.1	65.5	67.7
BERT + Rules (S2)	**73.2**	87.0	**79.5**	67.6	**46.0**	**54.8**	**70.4**	**66.5**	**68.4**
BERT-high-prec	71.9	86.4	78.5	64.9	42.9	51.6	68.4	64.6	66.5
BERT-high-prec + Rules (S3)	72.9	85.9	78.8	65.6	45.7	53.9	69.2	65.8	67.5

Table 1: Results on Subtask 1

	Other			Fact-Claiming			Average		
	P	R	F	P	R	F	P	R	F
Rules	68.0	99.7	80.8	90.0	5.7	10.8	79.0	52.7	63.2
BERT (S1)	**83.8**	74.9	79.1	58.5	**71.0**	64.2	71.2	**73.0**	**72.1**
BERT + Rules (S2)	**83.8**	74.9	79.1	58.5	**71.0**	64.2	71.2	**73.0**	**72.1**
BERT-high-prec	71.0	**99.4**	**82.8**	93.5	18.5	30.9	82.3	58.9	68.7
BERT-high-prec + Rules (S3)	71.1	**99.4**	**82.9**	**93.7**	18.8	31.3	**82.4**	59.1	68.8

Table 2: Results on Subtask 3

	TP	FP	TN	FN
BERT	151	72	522	199
Rules	21	10	584	329
BERT + Rules	161	77	517	189

Table 3: Number of true positive (TP), false positive (FP), true negative (TN), and false negative (FN) labels from each of our systems on the test set for Subtask 1

'I would vote for him'. A similar example is TP5: *DANKE Carla super weiter so* ❤️❤️❤️ 'Thanks Carla super keep it up', which may have been detected by the BERT model because of the capitalized word, but to account for the positive label in the ground truth we can only speculate once again that annotators have interpreted it as sarcastic.

Turning to comments that were not labeled as toxic by annotators, in the sample of true negatives (TN) we did not find any controversial examples. The false positives (FP), on the other hand, once again provide examples of the inherent difficulty and subjectivity of the task. Consider e.g. FP6 *Schnell viel Blödsinn reden…* 'Quickly talk a lot of nonsense…' and FP10 *Tja, mit Ideologie wird es kalt und dunkel hier!* 🤦‍♀️ 'Well, with ideology it gets cold and dark here!'. We believe that it would be necessary to also consider the post fragments that annotators had access to but were not included in the dataset (see Section 2) to determine how such comments could have been unanimously labeled as non-toxic. Several FP examples, however, are clearly not toxic, and in case of BERT one can only speculate as to why they were falsely classified as such. The three comments that were also false positives of the rule-based system (FP1-3) were misclassified because of the presence of the words *asozial* 'asocial', *Meinungsfreiheit* 'freedom of opinion', and *Horde* 'horde', illustrating the limitations of purely keyword-based methods.

The analysis in this section was intended to provide examples of the types of challenges a model of toxicity must concern itself with. While it is limited to a small sample from a single dataset, we believe it illustrates a range of problems that are typical for this task. In particular, false negative predictions are responsible for more than 70% of errors made by both of our top-performing systems, and our analysis suggests that identifying most of these would require more complex rules for modeling specific types of toxicity and the ability to detect sarcasm.

8 Conclusion

We described simple methods for the semi-automatic construction of rule-based systems for detecting toxicity in social media, and used them to improve the performance of a BERT-based classifier on the dataset of the 2021 GermEval shared task. A manual error analysis was provided to illustrate the most challenging aspects of the task.

Acknowledgments

Research conducted in collaboration with Botium GmbH.

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of NAACL*.

Tim Graf and Luca Salini. 2019. bertzh at germeval 2019: Fine-grained classification of german offensive language using fine-tuned bert. In *Proceedings of the 15th Conference on Natural Language Processing (KONVENS 2019)*, pages 434–437, Erlangen, Germany. German Society for Computational Linguistics & Language Technology.

Andrei Paraschiv and Dumitru-Clementin Cercel. 2019. Upb at germeval-2019 task 2: Bert-based offensive language classification of german tweets. In *Proceedings of the 15th Conference on Natural Language Processing (KONVENS 2019)*, pages 398–404, Erlangen, Germany. German Society for Computational Linguistics & Language Technology.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108, Online. Association for Computational Linguistics.

Julian Risch, Anke Stoll, Lena Wilms, and Michael Wiegand. 2021. 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 co-located with KONVENS*, pages 1–12.

Julia Struß, Melanie Siegel, Josef Ruppenhofer, Michael Wiegand, and Manfred Klenner. 2019. Overview of germeval task 2, 2019 shared task on the identification of offensive language. In *Preliminary proceedings of the 15th Conference on Natural Language Processing (KONVENS 2019), October 9 – 11, 2019 at Friedrich-Alexander-Universität Erlangen-Nürnberg*, pages 352–363, München, Germany. German Society for Computational Linguistics & Language Technology und Friedrich-Alexander-Universität Erlangen-Nürnberg.

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

Michael Wiegand, Melanie Siegel, and Josef Ruppenhofer. 2018. Overview of the GermEval 2018 shared task on the identification of offensive language. In *Proceedings of GermEval 2018, 14th Conference on Natural Language Processing (KONVENS 2018), Vienna, Austria – September 21, 2018*, pages 1–10, Vienna, Austria. Austrian Academy of Sciences.

Lena Wilms. 2021. personal communication.

AIT_FHSTP at GermEval 2021: Automatic Fact Claiming Detection with Multilingual Transformer Models

Jaqueline Böck [2], Daria Liakhovets [1], Mina Schütz [1],
Armin Kirchknopf [2], Djordje Slijepčević [2], Matthias Zeppelzauer [2], Alexander Schindler [1]

[1] Austrian Institute of Technology GmbH
Giefinggasse 4, 1210 Vienna, Austria
{daria.liakhovets.fl, mina.schuetz, alexander.schindler}@ait.ac.at

[2] St. Pölten University of Applied Sciences
Institute of Creative Media Technologies
3100 St. Pölten, Austria
{jaquelineboeck1}@gmx.at
{armin.kirchknopf, djordje.slijepcevic, matthias.zeppelzauer}@fhstp.ac.at

Abstract

Spreading ones opinion on the internet is becoming more and more important. A problem is that in many discussions people often argue with supposed facts. This year's GermEval 2021 focuses on this topic by incorporating a shared task on the identification of fact-claiming comments. This paper presents the contribution of the AIT_FHSTP team at the GermEval 2021 benchmark for task 3: "identifying fact-claiming comments in social media texts". Our methodological approaches are based on transformers and incorporate 3 different models: multilingual BERT, GottBERT and XML-RoBERTa. To solve the fact claiming task, we fine-tuned these transformers with external data and the data provided by the GermEval task organizers. Our multilingual BERT model achieved a precision-score of 72.71%, a recall of 72.96% and an F1-Score of 72.84% on the GermEval test set. Our fine-tuned XML-RoBERTa model achieved a precision-score of 68.45%, a recall of 70.11% and a F1-Score of 69.27%. Our best model is GottBERT (i.e., a BERT transformer pre-trained on German texts) fine-tuned on the GermEval 2021 data. This transformer achieved a precision of 74.13%, a recall of 75.11% and an F1-Score of 74.62% on the test set.

1 Introduction

Today's social media platforms allow any individual to share information and opinions easily and quickly across a wide audience with almost no restrictions. However, not only obviously offensive comments, but also comments and posts with false information are becoming a serious problem on the Internet. The sheer amount of available information and content generated every day makes it impossible to verify all information. Thus, misinformation and false information can easily spread and influence people and their decisions, which has a strong impact on our society.

As a workshop part of the KONVENS 2021 (Konferenz zur Verarbeitung natürlicher Sprache / Conference of Natural Language Processing) the GermEval 2021 focuses on the problem of fact claiming, i.e., the identification of content in social media that contains potential facts that need to be checked (Risch et al., 2021). The identification of such fact claiming content is a first step in the information verification process to separate relevant from irrelevant information for fact checking. Our team participated in the fact claiming task (task 3: identification of fact-claiming comments) of GermEval 2021 and this paper presents our methodology and the results. To solve the task we fine-tuned (supervised) the pre-trained transformer models with the original GermEval 2021 data and external data, i.e., the ClaimBuster dataset (Arslan et al., 2020). The employed datasets and our general approach are described in Section 2. A detailed description of the transformer-based models is provided in Section 3. In Section 4, our experimental setup is introduced. The results can be found in Section 5 followed by a brief discussion and conclusion in Section 6.

2 Methodological Approach

The GermEval 2021 provided one labeled dataset for all three tasks (task 1 and 2 not considered in our contribution). The data for task 3 contained approx. 1/3 of content that mentions claimed facts and 2/3 with no claimed facts. We applied three pre-trained transformer models (Vaswani et al., 2017) to encode and classify the content for this task, namely: German OSCAR text trained BERT (GottBERT) (Scheible et al., 2020), multilingual BERT (mBERT) (Devlin et al., 2019b) and XLM-RoBERTa (XLM-R) (Conneau et al., 2019). Transformers are usually pre-trained on a large general corpus and can be used for many natural language processing (NLP) downstream tasks, which makes them especially useful for small training corpora (Liu et al., 2019). Compared to mBERT and XLM-R, which are both pre-trained on multilingual data, GottBERT is the only one that was trained on one language (German) only. We fine-tuned these models in a supervised manner for binary classification into fact claiming comments and non fact claiming comments. Since we employ two multilingual models, we chose to fine-tune one of those (mBERT) on the GermEval 2021 data and an additional dataset. In comparison, we fine-tuned our second multilingual model (XLM-R) and the German GottBERT model using only the training data provided by the GermEval 2021 shared task.

The applied method is derived from our approach (Schütz et al., 2021a) presented in the EXIST 2021 challenge. The first shared task on sEXism Identification in Social neTworks (EXIST) at IberLEF 2021 (Rodríguez-Sánchez et al., 2021; Montes et al., 2021), covering a wide spectrum of sexist content and aims to differentiate different types of sexist content. In our EXIST 2021 contribution a comparable set of transformer models and processing steps were applied (Schütz et al., 2021a).

2.1 GermEval 2021 Data & Preprocessing

The data provided by the organizers of GermEval 2021 is an annotated dataset consisting of over 3,244 German Facebook comments on a political talk show of a German television broadcaster and user discussions from February to July 2019. The dataset was annotated and standardized. Links to users were anonymized with @USER, links to the show were replaced with @MEDIUM and links to the moderator were replaced with @MODERATOR. The original dataset was provided in CSV format. A subset of user comments from two shows were used for the train data. The comments in the test data were drawn from other shows. The dataset contained 1,103 (34%) instances which were labeled as *fact claiming* and 2,141 (66%) instances without any fact claims. The provided dataset is described in more detail in the GermEval 2021 overview paper (Risch et al., 2021).

In initial experiments, we applied different preprocessing strategies to the dataset. We tested our models on a processed version where all links in the dataset were replaced with @MEDIUM, since not every link was connected to the show that was the source of the data. Similarly, as an additional step for our multilingual models, we replaced all emojis with their English translations[1]. However, the two preprocessing steps had a slightly negative impact of 1% on average for mBERT, while they had a clearly positive impact of 3% for XLM-R. Therefore, we used the preprocessed training data only for the XLM-R model. Similarly, the replacement of links did not have a positive influence for the monolingual GottBERT model, where we also used the unpreprocessed comments as an input for training.

We did not use conventional preprocessing steps, e.g., stop-word removal, lemmatization, or stemming, because transform models do not need these due to their ability to capture more context in their word embeddings through improved pretraining capabilities and multi-head attention mechanisms (Vaswani et al., 2017; Devlin et al., 2019a; Liu et al., 2019).

2.2 External Data

As external data we use the ClaimBuster (Arslan et al., 2020) dataset, which consists of English statements from all U.S. presidential debates from 1960-2016. The original part of this dataset consists of 23,533 records. In total, 32,072 sentences were spoken in these debates. The presidential candidates spoke 26,322 sentences, debate moderators spoke 4,292 sentences and 1,319 sentences were spoken by the questioners. Sentences from

[1] https://pypi.org/project/emoji/

the moderators and the questioners were discarded and only the sentences spoken by the presidential candidates were considered for creating the ClaimBuster dataset. Moreover, sentences shorter then 5 words were also removed (2,789 sentences). The resulting dataset (*crowdsourced.csv*) was annotated by recruited participants (mostly university students). In addition, three experts labeled a subset of this dataset containing 1,032 sentences to create a groundtruth dataset (*groundtruth.csv*). The provided ClaimBuster dataset consists of three CSV files:

- *all_sentences.csv* (32,072 sentences): all sentences of the debates
- *crowdsourced.csv* (23,533 sentences): sentences of presidential candidates longer than 5 words, labeled by recruited participants
- *groundtruth.csv* (1,032 sentences): sentences of presidential candidates longer than 5 words, labeled by experts

For the GermEval 2021 challenge we used only the *groundtruth.csv* file to ensure high-quality data. The records in the file are annotated as follows:

- non-factual statement (NFS)
- unimportant factual statement (UFS)
- check-worthy factual statement (CFS)

Referring to the original paper (Arslan et al., 2020), the dataset is imbalanced in terms of class distribution: 23.87% belong to CFS, 10.45% to UFS and 65.68% to NFS. The instances (sentences) are annotated as numerical categories (“-1”, “0”, “1”). In order to match the ClaimBuster data with the original GermEval 2021 data, it was necessary to get an overview of the sentences first and afterwards match the labels to a unified format. Therefore, the comments with the labels “0” and “-1” have been mapped to “0” (not claiming). The instances labeled as “1” were not changed and thus assigned to the class of fact claiming comments. In a next step, we translated the whole dataset into German using the Google Translator API. The translation of the dataset was only used for the mBERT model, since in former work (Schütz et al., 2021a) it was shown that using additional data for this exact model can improve the predictions on a similar NLP downstream task.

3 Models

In total we used three different architectures, which are all based on the original transformer (Vaswani et al., 2017) model:

mBERT is a multilingual transformer based on the original structure of BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2019a). However, BERT was only trained on English data in comparison to the multilingual model which was additionally trained on Wikipedia data in 100 languages (Devlin et al., 2019b). BERT in general consists - unlike the original transformer with its encoder / decoder architecture (Vaswani et al., 2017) - only of an encoder and is pre-trained using two different strategies: Masked Language Modeling (MLM) and Next Sentence Prediction (NSP) (Devlin et al., 2019a). MLM masks words with a specific pattern in a sequence that the model has to predict using its bidirectionality and multi-headed attention (reading a sentence from left-to-right and right-to-left). NSP is the task of predicting the following sentence in the text input (Devlin et al., 2019a).

GottBERT: is a monolingual transformer model, which is based on RoBERTa (Robustly Optimized BERT Pretraining Approach) (Liu et al., 2019). The latter used the BERT architecture, but was trained with more data over a longer time period. Additionally, NSP was not used for pre-training the model and MLM was changed from static to dynamic, where they use a different mask pattern for every sequence during training instead of the same as in BERT. RoBERTa outperforms BERT in several NLP downstream tasks (Liu et al., 2019). Since the original RoBERTa model was only trained on English data, GottBERT was trained from scratch, with the same parameters as the German BERT version, on the German data of the OSCAR corpus (Scheible et al., 2020).

XLM-R: is a self-supervised cross-lingual model that was - similarly as mBERT - trained with monolingual CommonCrawl data in 100 languages (Conneau et al., 2019). The architecture is based on RoBERTa (similarly as GottBERT) in combination with the multilingual XLM transformer (Conneau and Lample, 2019). XLM uses more language modeling approaches (Conneau and Lample, 2019) than RoBERTa and is only trained monolingually with MLM (Conneau et al., 2019). XLM-R outperforms mBERT on multiple tasks (Conneau et al., 2019). Evaluation results showed that XLM-R especially works well for languages with less available data in comparison to other models (Conneau et al., 2019).

The three models do not only differ in the number of languages that they were trained on: BERT and RoBERTa have different pre-training strategies, whereas the strategy of RoBERTa are used by GottBERT as well as XLM-R. As more training data is used, the vocabulary increases, resulting in longer pre-training and fine-tuning intervals. This usually has a positive influence on the performance of downstream tasks.

4 Experimental Setup

Figure 1 provides an overview of our experimental setup and the training strategies used to solve the fact claiming task. The two main approaches take two distinct parts of input data, i.e., only GermEval 2021 data or in addition ClaimBuster data as input. To evaluate the proposed methods we performed experiments by utilizing the following pre-trained transformer models provided by the HuggingFace (Wolf et al., 2020) library: mBERT (Devlin et al., 2019a), Gottbert[2], and XLM-R[3] (Conneau et al., 2019). The experimental setup for the three models is described in detail below.

4.1 mBERT

The cased multilingual BERT transformer (Devlin et al., 2019a) was fine-tuned on the original GermEval 2021 data as well as the additional English ClaimBuster data (Arslan et al., 2020) and its German translations. Note that since the model is multilingual, we expect the English ClaimBuster data to have a positive impact on model training. Both datasets were not subject to any further preprocessing. The model was trained for 4 epochs with a learning rate of 1e-5, batch size of 8 and a maximum sequence length of 284.

4.2 GottBERT

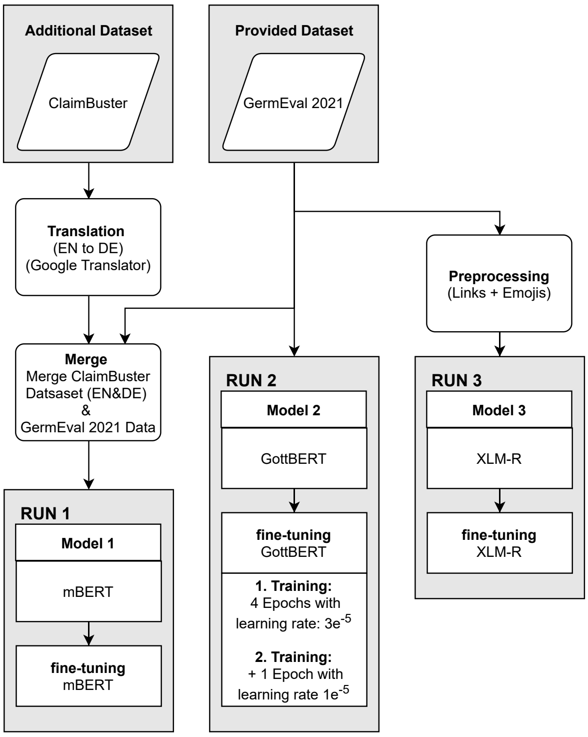


Figure 1: Overview of the setup of our submitted runs including the used models and data.

We fine-tuned the German RoBERTa model (GottBERT) (Scheible et al., 2020) using the GermEval 2021 data without any additional preprocessing. We trained the model for 4 epochs with a learning rate of 3e-5 and one more epoch with a learning rate of 1e-5, with a batch size of 8, a maximum sequence length of 128, weight decay of 0.01 and 500 warm-up steps.

4.3 XLM-R

The XLM-R (Conneau et al., 2019) model was trained on the preprocessed (replacing links with @MEDIUM and replacing emojis with their translated text) GermEval 2021 training data. We fine-tuned the model for 10 epochs, with a batch size of 16, a maximum sequence length of 256, a learning rate of 2e-5 without warm-up steps and Adam as an optimizer.

5 Results

The validation and test results in terms of precision, recall, and macro-averaged F1-score are presented in Table 1.

Run 1: The mBERT seems to generalize well, as the F1-score on the test set of 72.84% is at a similar performance level as on the validation set (76.09%). This result on the test set is the second highest achieved in our experiments.

[2] https://huggingface.co/uklfr/gottbert-base

[3] https://huggingface.co/xlm-roberta-base

Model	Run	P (val)	R (val)	F1 (val)	P (test)	R (test)	F1 (test)
mBERT	1	77.52	74.72	76.09	72.71	72.96	72.84
GottBERT	2	74.26	68.80	78.90	**74.13**	**75.11**	**74.62**
XLM-R	3	76.91	78.11	76.73	68.45	70.11	69.27

Table 1: Accuracy (A), precision (P), recall (R), and macro-averaged F1-scores (F1) for the GermEval 2021. Abbreviation "val" stands for our validation set and "test" for the official benchmark test set. The performance measures are expressed in percent (%).

Run 2: The fine-tuned GottBERT on the GermEval 2021 data achieved an overall F1-score of 74.62% on the test set (78.90% on the validation set). These results speak for the generalization ability of this network because the test performance is at a similar performance level as on the validation set. This result is the highest obtained for all our models.

Run 3: The XLM-R fine-tuned on the GermEval 2021 data achieved the lowest F1-score on the test set (69.27%). This approach seems to exhibit a strong overfitting behavior, as the results on the validation set are considerably higher (F1-score of 76.73%).

In conclusion, the GottBERT model (run 2) achieves the highest results in our experiments. These results indicate that the model that is pretrained on German data allows for a better modeling of the semantics of the task than a multilingual model. All other models are also beyond the zero-rule baseline which is at 66% for the test set.

A more detailed analysis of the results shows that all three models consistently predicted the same class in 560 cases (corresponds to approx. 60% of the test set). In the following, two examples are given for both classes:

Example 1 "@USER Sie würden wahrscheinlich auch einen Kriegstreiber/in wählen, wenn es gegen Trump ginge, warten sie es ab , vielleicht geht ihr Wunsch ja in Erfüllung...."
The ground truth and predictions of all models for this example are "0" (not fact claiming).

Example 2 "@USER , ich glaube,Sie verkrnnen gründlich die Situation. Deutschland mischt sich nicht ein, weil die letzte Einmischung in der Ukraine noch nicht bereinigt ist. Es geht nicht ums Militär"
The ground truth and predictions of all models for this example are "1" (fact claiming).

Furthermore, all three models consistently predicted the wrong class in 107 cases (corresponds to approx. 11% of the test set). In the following, two examples are given for both classes:

Example 1 ”Hackt nicht nimmer auf den Fussball rum. Bei allem Sportarten sind wieder Zuschauer erlaubt. Hygienekonzept vorausgesetzt."
The ground truth is "1" (fact claiming) and predictions of all models are "0" (not fact claiming).

Example 2 ”Biden gewinnt, Corona wird weggehen, Amerika wird reich,k alle bekommen ARbeit und die Welt wird schön. Also was sollst."
The ground truth is "0" (not fact claiming) and predictions of all models are "1" (fact claiming).

In the remaining 273 cases (corresponds to approx. 29% of the test set), one of the models did not predict the same as the others. mBERT and GottBERT predicted equally in 100 cases (70 correctly and 30 incorrectly). GottBERT and XLM-R predicted equally in 100 cases (47 correctly and 53 incorrectly). mBERT and XLM-R predicted equally in 73 cases (28 correctly and 45 incorrectly). These results show that even though both pairs mBERT and GottBERT on one side and mBERT and XLM-R on the other side predict equally in most cases (100), mBERT and GottBERT predict correctly in significantly more cases (70).

6 Conclusion & Future Work

In this paper, we described our submission to the "Fact-Claiming Comment Classification" task of the GermEval 2021. In our experiments GottBERT, a transformer-based machine learning model pretrained on German data only, achieved the best results, leading to an F1-score of 74.62% on the test set. For the multilingual transformer models, we obtained better results with mBERT (potentially because it was trained with an additional dataset) than with XLM-R, which seems to have slightly overfitted on the training data.

Future work will focus on evaluating the different models and approaches in more detail and to investigate how they specifically adapt to the underlying data. We will further investigate how the use of external data impacts the performance of all three investigated models, especially GottBERT, which seem to be the most promising option. Due to the similarity of the presented approaches in this challenge and our previous submission to the EXIST 2021 challenge, see (Schütz et al., 2021a), we plan to perform comparisons on how the applied models converge with respect to the different datasets, semantic concepts and downstream tasks addressed in the benchmarks. Furthermore, we will analyze whether the findings from this comparison can be applied to related tasks such as disinformation detection (Schütz et al., 2021b).

7 Acknowledgements

This contribution has been funded by the FFG Project "Defalsif-AI" (Austrian security research programme KIRAS of the Federal Ministry of Agriculture, Regions and Tourism(BMLRT), grant no. 879670) and the FFG Project "Big Data Analytics" (grant no. 866880).

References

Fatma Arslan, Naeemul Hassan, Chengkai Li, and Mark Tremayne. 2020. Claimbuster: A benchmark dataset of check-worthy factual claims.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019a. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019b. Multilingual BERT (mBERT). Accessed: 2010-06-02.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach.

Manuel Montes, Paolo Rosso, Julio Gonzalo, Ezra Aragón, Rodrigo Agerri, Miguel Ángel Álvarez Carmona, Elena Álvarez Mellado, Jorge Carrillo de Albornoz, Luis Chiruzzo, Larissa Freitas, Helena Gómez Adorno, Yoan Gutiérrez, Salud María Jiménez Zafra, Salvador Lima, Flor Miriam Plaza de Arco, and Mariona Taulé (eds.). 2021. Proceedings of the iberian languages evaluation forum (iberlef 2021). In *CEUR Workshop Proceedings*.

Julian Risch, Anke Stoll, Lena Wilms, and Michael Wiegand. 2021. Overview of the GermEval 2021 shared task on the identification of toxic, engaging, and fact-claiming comments. In *Proceedings of the GermEval 2021 SharedTask on the Identification of Toxic, Engaging, and Fact-Claiming Comments colocated with KONVENS*, pages 1–12.

Francisco Rodríguez-Sánchez, Jorge Carrillo de Albornoz, Laura Plaza, Julio Gonzalo, Paolo Rosso, Miriam Comet, and Trinidad Donoso. 2021. Overview of exist 2021: sexism identification in social networks. *Procesamiento del Lenguaje Natural*, 67(0).

Raphael Scheible, Fabian Thomczyk, Patric Tippmann, Victor Jaravine, and Martin Boeker. 2020. GottBERT: a pure german language model. *CoRR*, abs/2012.02110.

Mina Schütz, Jaqueline Boeck, Daria Liakhovets, Djordje Slijepčević, Armin Kirchknopf, Manuel Hecht, Johannes Bogensperger, Sven Schlarb, Alexander Schindler, and Matthias Zeppelzauer. 2021a. Automatic sexism detection with multilingual transformer models. *arXiv preprint arXiv:2106.04908*.

Mina Schütz, Alexander Schindler, Melanie Siegel, and Kawa Nazemi. 2021b. Automatic fake news detection with pre-trained transformer models. In *Pattern Recognition. ICPR International Workshops and Challenges. ICPR 2021. Lecture Notes in Computer Sciences*, volume 12667, Cham. Springer.

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

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

ur-iw-hnt at GermEval 2021:
An Ensembling Strategy with Multiple BERT Models

Hoai Nam Tran
Information Science
University of Regensburg
Regensburg, Germany
Hoai-Nam.Tran@student.ur.de

Udo Kruschwitz
Information Science
University of Regensburg
Regensburg, Germany
Udo.Kruschwitz@ur.de

Abstract

This paper describes our approach (ur-iw-hnt) for the Shared Task of GermEval2021 to identify toxic, engaging, and fact-claiming comments. We submitted three runs using an ensembling strategy by majority (hard) voting with multiple different BERT models of three different types: German-based, Twitter-based, and multilingual models. All ensemble models outperform single models, while BERTweet is the winner of all individual models in every subtask. Twitter-based models perform better than GermanBERT models, and multilingual models perform worse but by a small margin.

1 Introduction

Moderation of popular social media networks is a difficult task. Facebook alone has almost 2.8 billion active users on April 2021 (Kemp, 2021). Moderating discussions between users simultaneously all day is an impossible task, so moderators need help with this work. Also, fully automated solutions for content moderation are not possible, and human input is still required (Cambridge Consultants, 2019). An AI-based helper solution for harmful content detection is needed to make social networking less toxic and more pleasant instead.

The Shared Task of GermEval2021 focuses on highly relevant topics for moderators and community managers to moderate online discussion platforms (Risch et al., 2021). The challenge is not to specialize in one broad NLP task like harmful content detection but to detect other essential categories like which comments are engaging or fact-claiming.

We participated in all three subtasks (toxic, engaging and fact-claiming comment classification) to test our ensemble model to see whether multiple BERT-based models provide robust performance for different tasks without further customization. Moderators would benefit from a working system without having to change models or settings all the time.

This report discusses in detail the three runs we submitted in the GermEval2021 Shared Task (Risch et al., 2021). We start with a brief reflection on related work, only focussing on aspects that are closely aligned with the subtasks. We then explain the dataset and the shared tasks in more detail. Next, we present our experiments, some discussions of the results, and we finally draw some conclusions.

To encourage reproducibility of experimental work, we make all code available via GitHub[1].

2 Related Work

Detecting harmful content in social media platforms is not only a monolingual but a multilingual issue. A multilingual toxic text detection classifier uses a fusion strategy employing mBERT and XLM-RoBERTa on imbalanced sample distributions (Song et al., 2021). Deep learning ensembles also show their effectiveness in hate speech detection (Zimmerman et al., 2019). A taxonomy of engaging comments contains different possible classifications (Risch and Krestel, 2020). With the increasing spread of misinformation, more collaborations with IT companies specialized in fact-checking and more intelligence and monitoring tools are available to help to identify harmful content (Arnold, 2020). An attempt to fully automate fact-checking is the tool called ClaimBuster (Hassan et al., 2015). Another tool named CrowdTangle monitors social media platforms and alerts the user if specific keywords are triggered so manual fact-claim checking can be done (Arnold, 2020). In addition, an annotation schema for claim detection is also available (Konstantinovskiy et al., 2021).

[1] https://github.com/HN-Tran/GermEval2021

Dataset	Label	Subtask 1	Subtask 2	Subtask 3
Training	0	2122	2379	2141
	1	1122	865	1103
Test	0	594	691	630
	1	350	253	314

Table 1: Provided training and test dataset

3 Dataset & Shared Task

The dataset for the Shared Task of GermEval2021 consists of 3,244 annotated user discussion comments from a Facebook page of the German news broadcast in the timeframe of February to July 2019, labeled by four annotators in three different categories for binary classification: Toxic comments, engaging comments and fact-claiming comments (Risch et al., 2021). Since the labels are imbalanced, we first applied a stratified split onto the dataset so that 80% is for training. We then again apply a stratified split on what is left into two halves, the first part is the development set, and the second part is the holdout set for evaluation which we call the evaluation set here. After training, the ensemble strategy predicts the test dataset, consisting of 944 comments. Table 1 shows the imbalance in favor of the negative label. The organizers of GermEval2021 chose the metric Krippendorff's alpha to check each task's intercoder reliability (Risch et al., 2021).

3.1 Toxic Comment Classification

Toxic comments include many harmful and dangerous offenses like "hate speech, insults, threats, vulgar advertisements and misconceptions about political and religious tendencies" (Song et al., 2021). Such behavior only leads to users leaving the discussion or manual bans by the moderator, which can be overwhelming depending on the number of active toxic users (Risch and Krestel, 2020). For this subtask, the annotator agreement in the usage of insults, vulgar and sarcastic language is $0.73 < \alpha < 0.89$, and in the discrimination, discredition, accusations of lying or threats of violence, the agreement is at $0.83 < \alpha < 0.90$ (Risch et al., 2021).

3.2 Engaging Comment Classification

Engaging comments are, in general, attractive for users to participate in online discussions and get more interactions with other online users in the form of replies and upvotes. A taxonomy of engaging comments has been proposed to identify these comments for detection and classification, so moderators and community managers can reward these comments or posts (Risch and Krestel, 2020). This task has three different categories (Risch et al., 2021):

- Juristification, solution proposals, sharing of personal experiences ($0.71 < \alpha < 0.89$)
- Empathy with regard to other users' standpoints ($0.79 < \alpha < 0.91$)
- Polite interaction, mutual respect, mediation ($0.85 < \alpha < 1$)

3.3 Fact-Claiming Comment Classification

Detecting factual claims is part of the fact-checking process (Konstantinovskiy et al., 2021; Babakar and Moy, 2016; Nakov et al., 2021). The challenge here is to identify claims that have not been fact-checked before and go beyond one sentence that fits into this subtask (Babakar and Moy, 2016). Annotator's agreement in fact assertion and evidence provision is at $0.73 < \alpha < 0.84$ (Risch et al., 2021).

4 Experiments

4.1 System Architecture

For our system architecture (see Figure 1), we use three Python libraries/tools. Deep-Translator[2] translates all the German comments into English by choosing an external service, in our case, the free public Google Translate service. We use two different libraries for classification: Ernie[3] and Simple Transformers[4]. Both work on different versions of HuggingFace's Transformers (Wolf et al., 2020) and thus differently: Ernie is a beginner-friendly library last updated in 2020, based on Keras / TensorFlow 2, and uses the optimizer Adam (Kingma and Ba, 2015). Simple Transformers is based on PyTorch and has more extensive options for hyperparameter tuning and training customizations with AdamW (Loshchilov and Hutter, 2019) as the default optimizer. The default hyperparameter values for our experiments, as recommended for BERT, are in Table 2. The only pre-processing step is the tokenization by each BERT model using these libraries. Because of time constraints, cross-validation has not been conducted. After training

[2] https://github.com/nidhaloff/deep-translator

[3] https://github.com/labteral/ernie

[4] https://simpletransformers.ai/

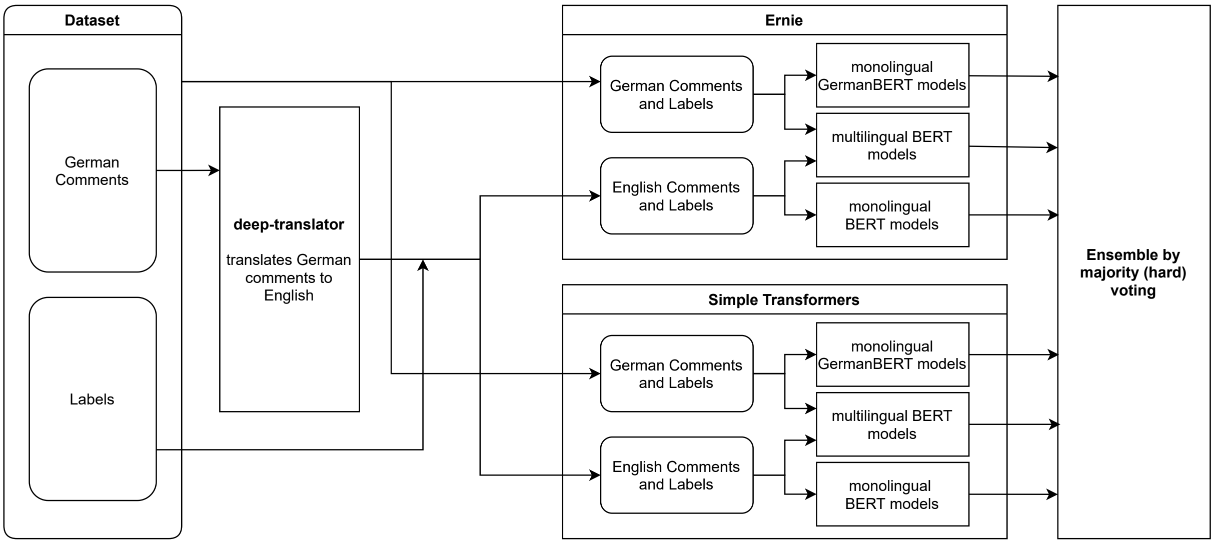

Dataset
German Comments
Labels
deep-translator
translates German comments to English
Ernie
German Comments and Labels
English Comments and Labels
monolingual GermanBERT models
multilingual BERT models
monolingual BERT models
Simple Transformers
German Comments and Labels
English Comments and Labels
monolingual GermanBERT models
multilingual BERT models
monolingual BERT models
Ensemble by majority (hard) voting

Figure 1: System architecture

Hyperparameter	Ernie	Simple Transformers
# epochs	3	3
max sequence length	128	128
learning rate	2e-5	4e-5
optimizer	Adam	AdamW

Table 2: Hyperparameter values

and evaluating the development and holdout set, the chosen models' predictions go to the ensemble strategy, which finally predicts the test dataset by majority (hard) voting.

4.2 BERT and its variants

BERT, which stands for Bidirectional Encoder Representations from Transformers, is a language model developed by Google and is known for its state-of-the-art (SOTA) performance in several NLP tasks (Devlin et al., 2019). The Shared Task consists of German Facebook comments, so we see it fit to choose German-based and English-translation-based models. Because Facebook comments have some similarity with Twitter comments, we also decide on Twitter-based models.

There are several versions of BERT with different pre-training or fine-tuning:

- German-based BERT models
  - DBMDZ GermanBERT[5]
  - Deepset.AI GermanBERT (Chan et al., 2020)
- Multilingual BERT models
  - mBERT Cased (Devlin et al., 2019)
  - XLM-RoBERTa (Conneau et al., 2019)
- Twitter-based BERT models
  - BERTweet (Nguyen et al., 2020)
  - XLM-T (Barbieri et al., 2021)

Table 3 shows the result of each BERT model on the evaluation/holdout set and on the test dataset with its labels for subtask 1 (which was provided after the submissions had been received).

4.3 Ensembling Strategy

The Ensemble Technique is a combination of classifiers' predictions for further classification (Opitz and Maclin, 1999). There are two popular types of ensembling: Bagging (Breiman, 1996) and Boosting (Freund and Schapire, 1999). Ensembles have been shown to be highly effective for a variety of NLP tasks, e.g., in the current top 10 of SQuAD 2.0[6], all models are ensembles. We went for simple majority ensembling using hard voting, which classifies with the largest sum of predictions from all models.

We decided to use the three runs for the Shared Task to test different combinations of BERT models for a robust and consistent result in the test dataset. That is why we chose five models for the first run, seven models for the second run, and for the third run, nine models ensembled together. The first ensemble consists of two GermanBERT models, the English BERT$_{base}$ model, one Twitter-based

[5] https://huggingface.co/dbmdz/bert-base-german-cased

[6] https://rajpurkar.github.io/SQuAD-explorer/

Classifier	**Language**	**macro F1**$_{eval}$	**macro F1**$_{test}$
1) BERT$_{base}$ Uncased (Devlin et al., 2019)	English	.6493	.6329
2) mBERT$_{base}$ Cased (Devlin et al., 2019)	English	.6247	.6194
3) mBERT$_{base}$ Cased (Devlin et al., 2019)	German	.6286	.6086
4) DBMDZ GermanBERT[5]	German	.6472	.6591
5) Deepset.AI GermanBERT (Chan et al., 2020)	German	.6481	.6608
6) BERTweet (Nguyen et al., 2020)	English	**.6798**	**.6832**
7) XLM-T (Barbieri et al., 2021)	English	.6553	.6681
8) XLM-T (Barbieri et al., 2021)	German	.6342	.6502
9) XLM-R$_{base}$ (Conneau et al., 2019)	English	.6421	.6482
10) XLM-R$_{base}$ (Conneau et al., 2019)	German	.3959	.3862

Table 3: BERT classifier result for subtask 1

model (BERTweet), and one multilingual model, so we have diversity for classification. For the second ensemble, one multilingual model and one Twitter-based model are added. The third ensemble has every classifier except the last one.

The results for each subtask are in Tables 4, 5, and 6, with precision, recall, and macro-averaged F1 score as the scoring metrics. The numbers in the column "Ensemble" refer to the classifier numbers from Table 3.

Run	**Ensemble**	**P**$_{test}$	**R**$_{test}$	**macro F1**$_{test}$
1	1,3,4,5,6	.7047	.6588	.6810
2	1,2,3,4,5,6,8	**.7183**	**.6635**	**.6898**
3	1,2,3,4,5,6,7,8,9	.7168	.6529	.6833

Table 4: Ensemble result for subtask 1

Run	**Ensemble**	**P**$_{test}$	**R**$_{test}$	**macro F1**$_{test}$
1	1,3,4,5,6	**.7228**	**.6653**	**.6929**
2	1,2,3,4,5,6,8	.7124	.6642	.6875
3	1,2,3,4,5,6,7,8,9	.7003	.6542	.6764

Table 5: Ensemble result for subtask 2

Run	**Ensemble**	**P**$_{test}$	**R**$_{test}$	**macro F1**$_{test}$
1	1,3,4,5,6	**.7791**	.7310	.7543
2	1,2,3,4,5,6,8	.7756	**.7454**	**.7602**
3	1,2,3,4,5,6,7,8,9	.7725	.7438	7579

Table 6: Ensemble result for subtask 3

5 Discussion

Our experiments demonstrate that BERTweet was showing better performance than every other model in every subtask, which is a surprise. We expected the monolingual GermanBERT models to perform best because of the cultural context in the integrated German language. Multilingual BERT models perform worst but by a close margin. Because of an overfitting issue, the tenth BERT classifier XLM-R performed faultily, only recognizing negative labels and thus the low macro-averaged F1 scores. The margin of each ensemble performance in subtasks 1 and 3 is around 1%, and for subtask 2 only around 2%. We conclude that the ensembling strategy shows robustness and consistency for the choice of good classifiers in a big enough amount for each task, and it could be a legitimate approach for the overfitting problem. Because of time constraints, no cross-validation was conducted, and since the holdout set was chosen not to be released for training, there is still improvement in the training quality of the BERT models so that more experiments are needed. Each part of a system like the GPU influences the training accuracy, so an identical replication is difficult to achieve, leading to different results. That is why reproducibility is not guaranteed, even if a manual seed is set[7]. Also, the amount and the imbalance of the dataset can lead to overfitting and lower scoring.

6 Conclusion and Future Work

We presented an ensemble strategy using ten BERT classifiers, including the use of machine translation, demonstrating robustness across tasks. While ensembles perform best overall, Twitter-based models (using standard BERT hyperparameter values) with translation to English perform best in a single model setting. This observation might change if cross-validation, early stopping, hyperparameter tuning, and other optimization techniques for each model are available for future work.

[7] https://pytorch.org/docs/stable/notes/randomness.html

Acknowledgements

This work was supported by the project *COURAGE: A Social Media Companion Safeguarding and Educating Students* funded by the Volkswagen Foundation, grant number 95564.

References

Phoebe Arnold. 2020. The challenges of online fact checking. Technical report, Full Fact, London, UK.

Mevan Babakar and Will Moy. 2016. The State of Automated Factchecking. Technical report, Full Fact, London, UK.

Francesco Barbieri, Luis Espinosa Anke, and José Camacho-Collados. 2021. XLM-T: A multilingual language model toolkit for twitter. *CoRR*, abs/2104.12250.

Leo Breiman. 1996. Bagging predictors. *Mach. Learn.*, 24(2):123–140.

Cambridge Consultants. 2019. Use of AI Online in online content moderation.

Branden Chan, Stefan Schweter, and Timo Möller. 2020. German's next language model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6788–6796, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv*.

J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.

Yoav Freund and Robert Schapire. 1999. A short introduction to boosting. *Journal-Japanese Society For Artificial Intelligence*, 14(771-780):1612.

Naeemul Hassan, Bill Adair, J. Hamilton, C. Li, M. Tremayne, Jun Yang, and Cong Yu. 2015. The Quest to Automate Fact-Checking. In *Proceedings of the 2015 computation+ journalism symposium*.

Simon Kemp. 2021. Digital 2021 april statshot report - datareportal – global digital insights.

Diederik P. Kingma and Jimmy Lei Ba. 2015. Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pages 1–15.

Lev Konstantinovskiy, Oliver Price, Mevan Babakar, and Arkaitz Zubiaga. 2021. Toward automated factchecking: Developing an annotation schema and benchmark for consistent automated claim detection. *Digital Threats: Research and Practice*, 2(2).

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. *7th International Conference on Learning Representations, ICLR 2019*.

Preslav Nakov, David P. A. Corney, Maram Hasanain, Firoj Alam, Tamer Elsayed, Alberto Barrón-Cedeño, Paolo Papotti, Shaden Shaar, and Giovanni Da San Martino. 2021. Automated fact-checking for assisting human fact-checkers. *CoRR*, abs/2103.07769.

Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. 2020. BERTweet: A pre-trained language model for English Tweets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, EMNLP 2020 - Demos, Online, November 16-20, 2020*, pages 9–14. Association for Computational Linguistics.

David Opitz and Richard Maclin. 1999. Popular Ensemble Methods: An Empirical Study. *Journal of Artificial Intelligence Research*, 11:169–198.

Julian Risch and Ralf Krestel. 2020. Top comment or flop comment? Predicting and explaining user engagement in online news discussions. *Proceedings of the 14th International AAAI Conference on Web and Social Media, ICWSM 2020*, pages 579–589.

Julian Risch, Anke Stoll, Lena Wilms, and Michael Wiegand. 2021. 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 co-located with KONVENS*, pages 1–12.

Guizhe Song, Degen Huang, and Zhifeng Xiao. 2021. A study of multilingual toxic text detection approaches under imbalanced sample distribution. *Information (Switzerland)*, 12(5):1–16.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Steven Zimmerman, Chris Fox, and Udo Kruschwitz. 2019. Improving hate speech detection with deep learning ensembles. *LREC 2018 - 11th International Conference on Language Resources and Evaluation*, pages 2546–2553.

Data Science Kitchen at GermEval 2021: A Fine Selection of Hand-Picked Features, Delivered Fresh from the Oven

Niclas Hildebrandt, Benedikt Boenninghoff, Dennis Orth, Christopher Schymura
Data Science Kitchen
{firstname.lastname}@data-science-kitchen.de

Abstract

This paper presents the contribution of the Data Science Kitchen at GermEval 2021 shared task on the identification of toxic, engaging, and fact-claiming comments. The task aims at extending the identification of offensive language, by including additional subtasks that identify comments which should be prioritized for fact-checking by moderators and community managers. Our contribution focuses on a feature-engineering approach with a conventional classification backend. We combine semantic and writing style embeddings derived from pre-trained deep neural networks with additional numerical features, specifically designed for this task. Ensembles of Logistic Regression classifiers and Support Vector Machines are used to derive predictions for each subtask via a majority voting scheme. Our best submission achieved macro-averaged F1-scores of 66.8%, 69.9% and 72.5% for the identification of toxic, engaging, and fact-claiming comments.

1 Introduction

In the early years after establishing social media platforms, setting up online discussion forums and installing comment areas on newspapers' websites, a door into this new digital world has been opened, allowing people to interconnect all over the world. Various communication platforms and social networks enabled new ways of sharing information with followers, exchanging opinions between politically interested people, and encouraging debates with the readers. Unfortunately, recent trends revealed the ugly face and adverse effects of these platforms when an increasing number of users make improper, illegal, or abusive use of such digital services (Mathew et al., 2019).

Nowadays, social media platforms are notorious for spreading toxic comments, in which the writers justify violence and discrimination against a person or groups of persons (Munn, 2020). Additionally, a second steadily growing trend is producing and sharing fake news or misinformation, seeking to dominate current discussions, and frame public debates (Mahid et al., 2018).

Both hate speech, fake news and their impact have become very prominent in recent years. However, the tremendous amount of shared and distributed toxic messages on social media platforms make it utterly infeasible to identify and tag or delete poisonous comments manually. The GermEval 2021 shared task tries to encounter this negative trend and motivates participants to work on automated solutions towards safer and more reliable digital rooms of interaction (Risch et al., 2021).

Therefore, the organizers of the task increased the difficulty of the competition by expanding the focus not only on the identification of toxic messages in online discussions but also on distinguishing between engaging and fact-claiming comments. The first task is similar to the GermEval tasks in 2018 (Wiegand et al., 2018) and 2019 (Struß et al., 2019) and deals with identifying toxic comments, including offensive, hateful and vulgar language or ruthless cynism. As novel subtasks, the participants are also invited to identify two additional categories of comments: The second category defines engaging comments, which are annotated as highly relevant contributions by the moderators. The third category concentrates on finding fact-claiming comments that should be considered for a manual fact-check with a higher priority.

2 Task and Data Description

Each subtask of GermEval 2021 is defined as a binary classification problem and all tasks share the same training and test data. The set of training data consists of 3,244 Facebook comments from a German news broadcast page. The anonymized comments were posted in the time span from February to July 2019 and were labeled by trained annotators. Binary labels were provided for each of the three categories. The test data is also extracted from Facebook discussions and include 944 comments. However, these comments had a different discussion topic than the training data. Precision, recall, and macro-averaged F1-score were defined as the relevant evaluation metrics.

2.1 Subtask 1: Toxic Comment Classification

Toxic comments are characterized by their offensive and hateful language, intended to blame other people or groups. For social media and content providers, it is important to detect such comments in a highly automated and scalable way. An example of a toxic comment from the training data of the GermEval shared task is: *"Na, welchem tech riesen hat er seine Eier verkauft..?"*. However, some of the comments which have been labeled as toxic can be quite hard to detect. Examples of such cases are: *"@USER eididei sieh mal an"* or *"ein schöner VW Golf Diesel.."*. Difficulties occur due to irony, subtle overtones and missing contextual information.

2.2 Subtask 2: Engaging Comment Classification

Engaging comments encourage other users to join the discussion, express their opinions and share ideas regarding the topic. They are characterised by being rational, respectful, and reciprocal and hence can foster a constructive and fruitful discussion. The comment *"Wie wär's mit einer Kostenteilung. Schließlich haben beide Parteien (Verkäufer und Käufer) etwas von der Tätigkeit des Maklers. Gilt gleichermassen für Vermietungen. Die Kosten werden so oder soweit verrechnet, eine Kostenreduktion ist somit nicht zu erwarten."* is an example of an engaging comment from the training data.

2.3 Subtask 3: Fact-Claiming Comment Classification

If a platform provider has to prevent the spread of fake news and misinformation, there is the demand of automatically identifying fact-claiming comments to assess their truthfulness. An example of a fact-claiming comment from the training data is the comment *"Dummerweise haben wir in der EU und in der USA einen viel höheren CO2 Fußabdruck als z.B. die Afrikaner oder Inder."*.

3 System Overview

The general system architecture is shown in Fig. 1. As the number of samples in the training dataset provided for GermEval 2021 is rather small, our proposed framework focuses on suitable feature engineering with a conventional classification backend. These features and further implementation details of our system are described in the following.

3.1 Preprocessing

Raw input text is preprocessed in three different processing streams that are handled in parallel. The first stream utilizes the tokenizer of a German BERT model (Chan et al., 2020) and crops the corresponding input text at a maximum length of 512 tokens. The second stream uses the SoMaJo tokenizer (Proisl and Uhrig, 2016) for German language and the third stream passes the raw text to the subsequent feature extraction stage without any preprocessing.

3.2 Feature Extraction

The feature extraction stage focuses on embedding-based features, as well as manually selected, numerical feature representations. Specific feature types are computed using one of the three preprocessing streams described in Sec. 3.1. This specific feature extraction setup was chosen to efficiently combine embedding representations that capture linguistic properties with "hand-crafted" features specifically designed for the GermEval 2021 tasks.

3.2.1 Semantic Embeddings

The first kind of embeddings used in our framework are document embeddings derived from a pre-trained German BERT model (Chan et al., 2020). Specifically, we used the `bert-base-german-cased` implementation from Huggingface[1]. This model was trained on a German Wikipedia dump, the OpenLegalData dump (Ostendorff et al., 2020) and news articles. Average pooling was used to compute 768-dimensional document-level embeddings from the BERT model output.

3.2.2 Writing Style Embeddings

Besides semantic document embeddings, we additionally experimented with neural stylometric embeddings that have been automatically extracted

[1] `https://huggingface.co/dbmdz/bert-base-german-cased`

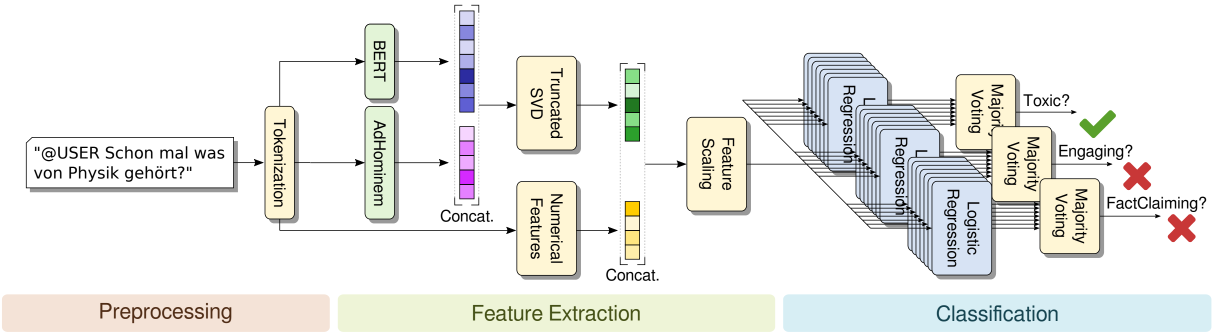


Figure 1: General architecture of the proposed framework to detect toxic, engaging and fact-claiming comments. Yellow boxes denote non-trainable, computational functions and transformations, green boxes represent pretrained models utilized for feature extraction. Trainable models whose parameters are optimized using the challenge dataset are shown in blue.

from the comments. More precisely, we used an extended framework of AdHominem (Boenninghoff et al., 2019) which outperformed all other systems that participated in the PAN 2020 and 2021 authorship verification tasks (Boenninghoff et al., 2021).

The overall framework consists of three components: In a first step, we perform neural feature extraction and deep metric learning (DML) to encode the writing style characteristics of a pair of raw documents into a pair of fixed-length representations, which is realized in the form of a Siamese network. Inspired by (Hochreiter and Schmidhuber, 1997), the Siamese network consists of a hierarchical LSTM-based topology. Next, the obtained representations are fed into a Bayes factor scoring (BFS) layer to compute the posterior probability for this trial. The idea of this second component is to take into account both, the similarity between the questioned documents and the typicality w.r.t. the relevant population represented by the training data. The third component is given by an uncertainty adaptation layer (UAL) aiming to correct possible misclassifications and to return corrected and calibrated posteriors. More details can be found in (Boenninghoff et al., 2021).

To train the model, we prepared a large dataset of Zeit-Online forum comments[2]. Altogether, we collected 9,812,924 comments written by 204,779 authors. Afterwards we split the dataset into training and validation sets. We took 10% of the authors to build the validation set and removed all comments with less than 60 tokens. Due to the fact that the provided dataset of the shared task also contains concise comments, we decided to leave all short comments in the training set. As a result, the datasets are disjoint w.r.t. the authors, i.e., all authors in the validation set have been removed from the training set. During training, we perform data augmentation by resampling new same-author and different-authors pairs in each epoch. Contrary, the pairs of the validation set are sampled once and then kept fixed. Since some authors contribute with hundreds of comments, we limited their influence by sampling not more than 20 comments per author. In summary, the training set contains approximately 234,500 same-author and 244,200 different-authors pairs in each epoch, where, on average, each comment consists of 75.90 ± 68.07 tokens. The validation set contains 15,125 same-author and 18,740 different-authors pairs, where, on average, each comment consists of 126.51 ± 65.31 tokens. Hence, both datasets are nearly balanced.

Table 1: Results for PAN 2021 evaluation metrics.

Model	PAN 2021 Evaluation Metrics					
	AUC	c@1/acc	f_05_u	F1	Brier	Overall
DML	87.4	79.3	81.7	81.0	85.1	82.9
BFS	87.4	79.5	80.5	82.0	85.5	83.0
UAL	87.6	79.5	81.6	81.4	85.6	83.2

We choose the PAN 2021 evaluation metrics to evaluate the performance as described in (Kestemont et al., 2021). Table. 1 summarizes the results, where all three system components are evaluated separately. It can be seen that we achieved overall scores between 82.9 and 83.2 for the components, which is mainly supported by higher values for the AUC and Brier scores. Comparing the c@1, F1 or f_05_u metrics, we generally obtained error rates of approximately 20% on this challenging dataset for a fixed threshold. After training, one part of the neural feature extraction component within the Siamese network is then used to extract the 100-dimensional writing style embeddings for the shared task data.

[2] www.zeit.de

Table 2: Overview of all features utilized in this work that are not based on embeddings.

Feature name	**Dim.**	**Description**
`NumCharacters`	1	Total number of characters, including white spaces.
`NumTokens`	1	Total number of tokens, after splitting at white spaces.
`AverageTokenLength`	1	Average number of characters in all tokens.
`TokenLengthStd`	1	Standard deviation of the number of characters in all tokens.
`StopwordRatio`	1	Number of stop words divided by the number of tokens.
`ExclamationMarkRatio`	1	Number of exclamation marks divided by the number of characters.
`NumReferences`	1	Number of hyperlinks in the comment.
`NumMediumAdressed`	1	Number of `@MEDIUM` mentions in the comment.
`NumUserAdressed`	1	Number of `@USER` mentions in the comment.
`AverageEmojiRepetition`	1	Average repetition number of emojis used in the comment.
`SpellingMistakes`	17	Number of specific grammar and spelling mistakes, cf. Sec. 3.2.3.
`SentimentBERT`	3	Sentiment scores of a pre-trained BERT model (Guhr et al., 2020).

3.2.3 Additional Numerical Features

In addition to the semantic and writing style embeddings, we integrated a set of specifically designed numerical features into our framework. An overview of these features, their dimensionality and corresponding descriptions is given in Tab. 2. We applied the natural logarithm to all strictly-positive numerical features.

The first group of features, `NumCharacters`, `NumTokens`, `AverageTokenLength` and `TokenLengthStd`, were chosen to reflect general structural properties of the comments in the dataset. In addition, we use the `StopwordRatio` and `ExclamationMarkRatio` features to explicitly reflect task-related semantic properties in the dataset. These task-specific features are accompanied by additional count-based features `NumMediumAdressed`, `NumUserAddressed`, `NumReferences` and `AverageEmojiRepetition`. We also included the scores (corresponding to the classes "positive", "neutral" and "negative") of a BERT model for sentiment classification trained on 1,834 million German-language samples derived from various sources (Guhr et al., 2020) as a dedicated `SentimentBERT` feature.

Lastly, we included an 17-dimensional feature denoted as `SpellingMistakes` into our set of additional features. This feature represents spelling and grammar mistakes from 17 different categories. We used a Python wrapper from the open-source grammar checker `LanguageTool`[3] to derive this feature. In particular, the following classes of mistakes were considered: *Typography*, *punctuation*, *grammar*, *upper/lowercase*, *support in punctuation*, *colloquialism*, *compounding*, *confused words*, *redundancy*, *typos*, *style*, *proper nouns*, *idioms*, *recommended spelling*, *miscellaneous*, *double punctuation*, *double exclamation mark*. For every category, we counted the number of mistakes and divided them by the number of tokens in the respective comment.

3.3 Classification Pipeline

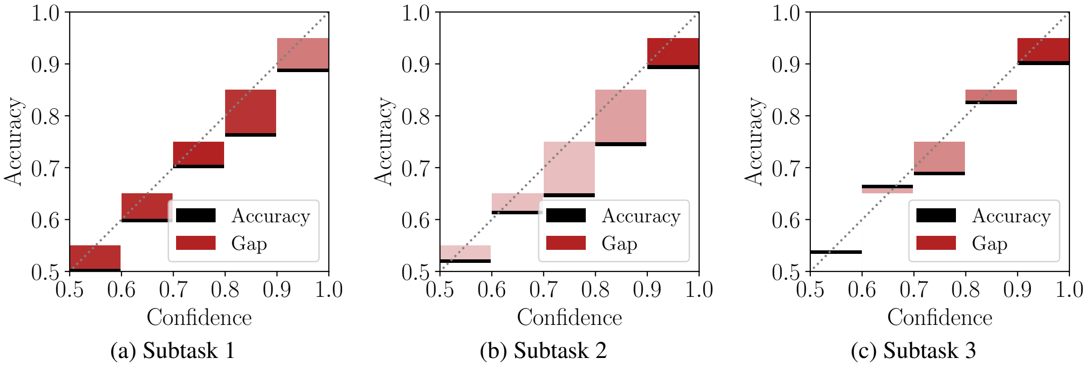
The classification pipeline used in this work is depicted in Fig. 1. The semantic and writing style embedding features described in Secs. 3.2.1 and 3.2.2 are concatenated, yielding a 868-dimensional joint embedding vector. A truncated singular-value decomposition (SVD) (Halko et al., 2011) is applied on this vector to reduce its dimensionality for subsequent processing. The number of dimensions kept is treated as a hyperparamter during training, cf. Sec. 4. The reduced joint embedding vector is then concatenated with the 28-dimensional vector of additional numerical features. The resulting vector is standardized to zero-mean and unit variance and serves as input to the classification stage.

We use Logistic Regression (Berkson, 1944; Haggstrom, 1983) and Support Vector Machines (SVMs) (Boser et al., 1992) with radial basis function (RBF) kernel as base classifiers within individual ensembles. One ensemble of binary classifiers is utilized for each subtask. Each classifier in the ensembles is trained using a subset of the provided training data via a cross-validation setup, cf. Sec. 4. A hard majority-voting scheme is used in each ensemble to obtain the predicted labels.

4 Evaluation

Our framework is trained using a specific cross-validation and hyperparameter tuning scheme, which is described in the following.

[3] `https://languagetool.org/`



(a) Subtask 1 (b) Subtask 2 (c) Subtask 3

Figure 2: Reliability diagrams of our first submission for all three subtasks (see Section 4.1). The red bars are drawn darker for bins with a higher number of samples.

4.1 Evaluation Metrics

Precision, recall and macro average F1-score are used for model evaluation (Opitz and Burst, 2021) since they represents the evaluation metrics of the GermEval 2021 shared task. Additionally, we assess the calibration properties of our model by determining the expected calibration error (ECE) as well as the maximum calibration error (MCE), where the confidence interval is discretized into a fixed number of M bins (Naeini et al., 2015). The ECE is then computed as the weighted macro-averaged absolute error between confidence and accuracy of all bins,

$$\mathrm{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{N} \left|\mathrm{acc}(B_m) - \mathrm{conf}(B_m)\right|, \quad (1)$$

where N is the total number of samples and $\mathrm{acc}(B_m) - \mathrm{conf}(B_m)$ is the difference between the actual accuracy and classifier confidence within a fixed-size bin B_m in the confidence interval. Note that all confidence values lie within the interval $[0.5, 1]$,since we are dealing with binary classification tasks. Hence, to obtain confidence scores, the output predictions p are transformed w.r.t. to the estimated subtask label, showing $\mathrm{conf} = p$ if the $\mathrm{acc} >= 0.5$ and $\mathrm{conf} = 1 - p$ if $\mathrm{acc} < 0.5$. The MCE returns the maximum absolute error, given as

$$\mathrm{MCE} = \max_{m \in 1,\dots,M} \left|\mathrm{acc}(B_m) - \mathrm{conf}(B_m)\right|. \quad (2)$$

We further display the reliability diagrams in Fig. 2 which will be discussed in Section 5.

4.2 Experimental Setup

Our experimental setup involves dedicated model selection and hyperparameter tuning. The training set performance is evaluated in a stratified K-Fold cross validation setup preserving the class label distribution among all folds. One of the K folds is used as validation set. We utilized a 7-fold cross-validation scheme and computed the evaluation metrics described in Sec. 4.1 on the validation set of each fold.

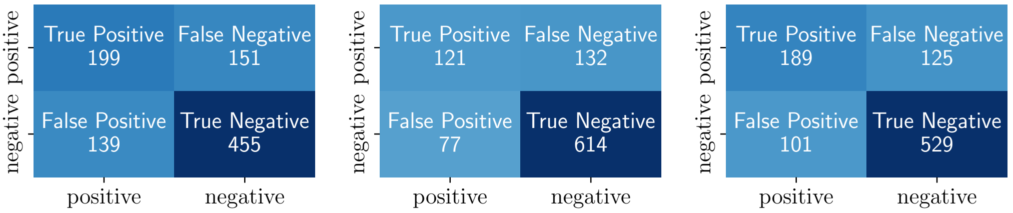
For submission one and two there are 7 logistic regression models for each subtask trained on different folds and stacked together in a voting ensemble returning the prediction of the majority. On each fold the L2-regularisation strength `C` and the number of features coming from the SVD dimension reduction are tuned with respect to the macro averaged F1 score over all subtasks. This means that hyperparameters may be slightly different from fold to fold but all three models trained on the same fold get the same hyperparameters – regardless the classification task.

Submission three uses a similar approach but the logistic regression models are replaced by SVMs having the same fold-wise hyperparameter tuning as mentioned above. In addition, task-wise tuned SVMs are added to the ensemble. Doubling the number of models and including a higher level of customisation to the task. The task-wise tuning includes optimisation of `kernel`, L2 regularisation strength `C`, `class weight`(whether or not to weight `C` with the class label distribution) and the kernel coefficient `gamma` as defined in the sklearn library (Pedregosa et al., 2011).

Hyperparameter tuning is performed with Bayesian optimisation using the Optuna library (Akiba et al., 2019). The macro average F1-score is chosen as optimisation target and the best hyperparameters among 100 trails are used in the ensemble.

Table 3: Final submission results on the test set including the calibration metrics for the first submission.

Run	**Subtask 1**					**Subtask 2**					**Subtask 3**				
	P	R	F1	ECE	MCE	P	R	F1	ECE	MCE	P	R	F1	ECE	MCE
Submission 1	65.95	63.67	64.79	5.5	8.0	69.70	67.78	68.72	6.9	10.4	73.25	71.44	72.34	3.5	6.0
Submission 2	64.89	62.71	63.78	–	–	69.26	67.43	68.33	–	–	73.39	71.52	72.44	–	–
Submission 3	66.98	66.73	66.85	–	–	71.71	68.34	69.98	–	–	73.03	72.08	72.55	–	–



(a) Subtask 1 (b) Subtask 2 (c) Subtask 3

Figure 3: Confusion matrices for submission 3.

5 Results and Discussion

The final submission results are provided in Table 3. Unexpectedly, the identification of toxic comments turns out to be the most challenging subtask while the detection of fact-claiming comments achieved the highest F1-score. This confirms our observations during hyperparameter tuning. For instance, the F1-score for the third submission after cross validation are given by 66.31 ± 1.76, 75.12 ± 2.07 and 74.68 ± 2.67 for subtasks 1-3, respectively. A comparison of our cross validation performance with the results on the test set shows two interesting findings: On the one side, we obtained very robust results of subtasks 1 and 3. On the other side, subtask 2 struggles with over-fitting effects.

In addition, Fig. 3 displays the confusion matrices of our third submission (representative for all submissions). It can be seen for all subtasks that the ratio of wrongly classified positively labeled samples is significantly larger than for negatively labeled samples. This behavior is supported by the reliabilty diagrams[4] in Fig. 2, where our submission delivers *over-confident* scores (i.e. conf $>$ acc) in nearly all bins. As a results, the higher proportion of wrongly classified comments for positively labeled comments leads to a lower performance in terms of the F1-score.

Finally, we visualize an estimated probability density function of the first submission using a non-parametric Gaussian kernel density estimator[5] in Fig. 4. Ideally, we would expect a bimodal probability density function. However, the plot shows that the system clearly tends towards self-confident predictions close to zero. But in regions closer to one, the systems behave more hesitant. This effect can be explained by the imbalanced distribution of the class labels.

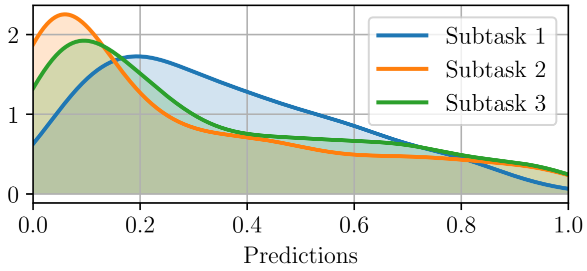


Figure 4: Gaussian kernel density estimates for the distributions of the first submission (bandwidth= 0.08).

6 Conclusions

Within this contribution to the shared task of the GermEval 2021 we have developed a modular feature extraction scheme which incorporates semantic and writing style embeddings as well as task specific numerical features. Less complex algorithms like logistic regression models and SVMs converge converge faster than complex models like deep neural networks and therefore need less training data. The combination with automated hyperparameter tuning and dimension reduction as well as the final agglomeration of multiple models in voting ensembles allow to achieve an macro-averaged F1-scores of 66.8%, 69.9% and 72.5% for the identification of toxic, engaging, and fact-claiming comments.

[4] https://github.com/hollance/reliability-diagrams

[5] https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KernelDensity.html

References

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A next-generation hyperparameter optimization framework.

Joseph Berkson. 1944. Application of the logistic function to bio-assay. *Journal of the American Statistical Association*, 39(227):357–365.

B. Boenninghoff, S. Hessler, D. Kolossa, and R. M. Nickel. 2019. Explainable Authorship Verification in Social Media via Attention-based Similarity Learning. In *IEEE International Conference on Big Data*, pages 36–45.

Benedikt Boenninghoff, Dorothea Kolossa, and Robert M. Nickel. 2021. Self-Calibrating Neural-Probabilistic Model for Authorship Verification Under Covariate Shift. In *12th International Conference of the CLEF Association (CLEF 2021)*. Springer.

Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. 1992. A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, COLT '92, page 144–152, New York, NY, USA. Association for Computing Machinery.

Branden Chan, Stefan Schweter, and Timo Möller. 2020. German's next language model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6788–6796, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Oliver Guhr, Anne-Kathrin Schumann, Frank Bahrmann, and Hans Joachim Böhme. 2020. Training a broad-coverage german sentiment classification model for dialog systems. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 1620–1625, Marseille, France. European Language Resources Association.

Gus W. Haggstrom. 1983. Logistic regression and discriminant analysis by ordinary least squares. *Journal of Business & Economic Statistics*, 1(3):229–238.

N. Halko, P. G. Martinsson, and J. A. Tropp. 2011. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, 53(2):217–288.

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

Mike Kestemont, Efstathios Stamatatos, Enrique Manjavacas, Janek Bevendorff, Martin Potthast, and Benno Stein. 2021. Overview of the Authorship Verification Task at PAN 2021. In *CLEF 2021 Labs and Workshops, Notebook Papers*. CEUR-WS.org.

Zaitul Iradah Mahid, Selvakumar Manickam, and Shankar Karuppayah. 2018. Fake news on social media: Brief review on detection techniques. In *2018 Fourth International Conference on Advances in Computing, Communication Automation (ICACCA)*, pages 1–5.

Binny Mathew, Ritam Dutt, Pawan Goyal, and Animesh Mukherjee. 2019. Spread of hate speech in online social media. WebSci '19, page 173–182, New York, NY, USA. Association for Computing Machinery.

Luke Munn. 2020. Angry by design: toxic communication and technical architectures. *Humanities and Social Sciences Communications*, 7(53):229–238.

Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. 2015. Obtaining well calibrated probabilities using bayesian binning.

Juri Opitz and Sebastian Burst. 2021. Macro f1 and macro f1.

Malte Ostendorff, Till Blume, and Saskia Ostendorff. 2020. Towards an open platform for legal information. In *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2020*, JCDL '20, page 385–388, New York, NY, USA. Association for Computing Machinery.

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

Thomas Proisl and Peter Uhrig. 2016. SoMaJo: State-of-the-art tokenization for German web and social media texts. In *Proceedings of the 10th Web as Corpus Workshop (WAC-X) and the EmpiriST Shared Task*, pages 57–62, Berlin. Association for Computational Linguistics (ACL).

Julian Risch, Anke Stoll, Lena Wilms, and Michael Wiegand. 2021. 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 co-located with KONVENS*, pages 1–12.

Julia Maria Struß, Melanie Siegel, Josep Ruppenhofer, Michael Wiegand, and Manfred Klenner. 2019. Overview of germeval task 2, 2019 shared task on the identification of offensive language. In *Proceedings of the 15th Conference on Natural Language Processing (KONVENS 2019)*, pages 354–365, Erlangen, Germany. German Society for Computational Linguistics & Language Technology.

Michael Wiegand, Melanie Siegel, and Josef Ruppenhofer. 2018. Overview of the germeval 2018 shared task on the identification of offensive language.

UR@NLP_A_Team @ GermEval 2021: Ensemble-based Classification of Toxic, Engaging and Fact-Claiming Comments

Kwabena Odame Akomeah
University of Regensburg
kwabena-odame.akomeah@ur.de

Udo Kruschwitz
University of Regensburg
udo.kruschwitz@ur.de

Bernd Ludwig
University of Regensburg
bernd.ludwig@ur.de

Abstract

In this paper, we report on our approach to addressing the *GermEval 2021 Shared Task on the Identification of Toxic, Engaging, and Fact-Claiming Comments* for the German language. We submitted three runs for each subtask based on ensembles of three models each using contextual embeddings from pre-trained language models using SVM and neural-network-based classifiers. We include language-specific as well as language-agnostic language models – both with and without fine-tuning. We observe that for the runs we submitted that the SVM models overfitted the training data and this affected the aggregation method (simple majority voting) of the ensembles. The model records a lower performance on the test set than on the training set. Exploring the issue of overfitting we uncovered that due to a bug in the pipeline the runs we submitted had not been trained on the full set but only on a small training set. Therefore in this paper we also include the results we get when trained on the full training set which demonstrate the power of ensembles.

1 Introduction

The need to check and moderate conversations and text on Social Media keeps increasing proportionally to the use of Social Media over the years (Shu et al., 2018; Rizoiu et al., 2019; Waseem and Hovy, 2016). Research into the identification of hate speech or toxic comment and fake news have recently become more popular in languages other than English because the abuse of free speech online and spread of information whether false or true extends farther than we can imagine (Vosoughi et al., 2018; Zampieri et al., 2020). GermEval 2021 (Risch et al., 2021) contains three subtasks not only aimed at identifying toxic comments in German text on social media platforms like in previous years (Struß et al., 2019) but also the classification of engaging and fact-claiming comments. In a way to help the situation of diffusing toxic content and promote positive content moderators on popular social media platforms also seek to promote texts that engage other users in a healthy conversation (Welch et al., 2016). The connection between hate speech and fake news is immense as the latter can rather stir up the masses into targeted hate towards a group of people or in some instances deadly violence (Moon et al., 2020). Therefore identifying social media content that makes a-need-to-check claim is as important as identifying hate content online.

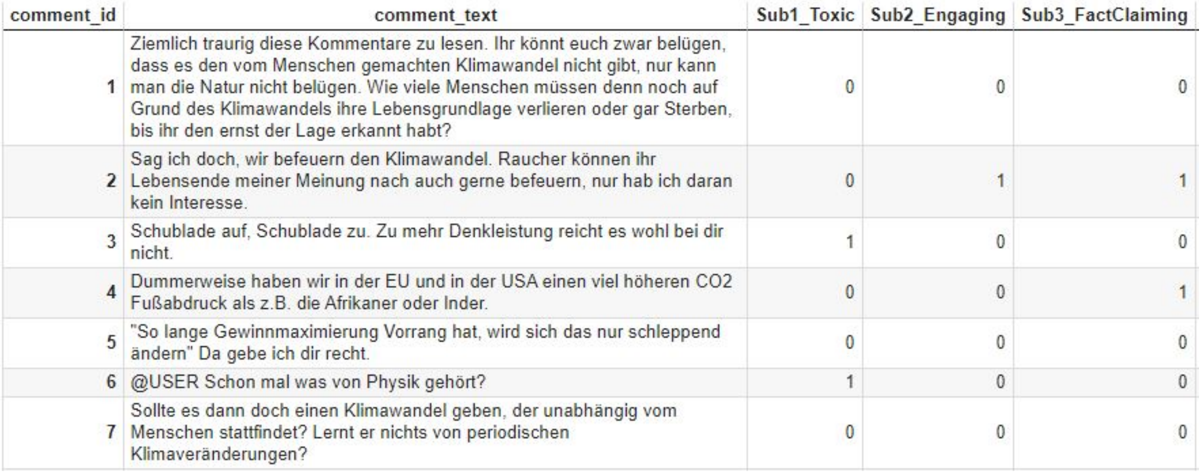
Our participation in GermEval 2021 was in all three subtasks and involved the use of the same model architectures on all three to learn, compare and analyse how models behave on subtasks. We applied Transformer-based embeddings (BERT), RNN-based embeddings (BiLSTM) with a classifier either utilising a densely connected output layer of a simple neural network or a Support Vector Machine in an ensemble constructed with majority voting of three models on all three subtasks.

The next sections discuss in detail the dataset used for our experiment and the model architectures applied. We also discuss and compare the performances of the models on the subtasks. All code used in this experiment can be accessed via GitHub.[1]

2 Dataset and Task

The dataset provided for this competition includes a trial set of 113 user comments, a training set of 3,244 user comments and a test set of 944 user comments of German text in csv format. The training set provided consists of over 3,000 Facebook anonymized user comments that were annotated by

[1] https://github.com/kaodamie/GermEval2021_Kobby_participation



comment_id	comment_text	Sub1_Toxic	Sub2_Engaging	Sub3_FactClaiming
1	Ziemlich traurig diese Kommentare zu lesen. Ihr könnt euch zwar belügen, dass es den vom Menschen gemachten Klimawandel nicht gibt, nur kann man die Natur nicht belügen. Wie viele Menschen müssen denn noch auf Grund des Klimawandels ihre Lebensgrundlage verlieren oder gar Sterben, bis ihr den ernst der Lage erkannt habt?	0	0	0
2	Sag ich doch, wir befeuern den Klimawandel. Raucher können ihr Lebensende meiner Meinung nach auch gerne befeuern, nur hab ich daran kein Interesse.	0	1	1
3	Schublade auf, Schublade zu. Zu mehr Denkleistung reicht es wohl bei dir nicht.	1	0	0
4	Dummerweise haben wir in der EU und in der USA einen viel höheren CO2 Fußabdruck als z.B. die Afrikaner oder Inder.	0	0	1
5	"So lange Gewinnmaximierung Vorrang hat, wird sich das nur schleppend ändern" Da gebe ich dir recht.	0	0	0
6	@USER Schon mal was von Physik gehört?	1	0	0
7	Sollte es dann doch einen Klimawandel geben, der unabhängig vom Menschen stattfindet? Lernt er nichts von periodischen Klimaveränderungen?	0	0	0

Figure 1: Small sample of the Training Data.

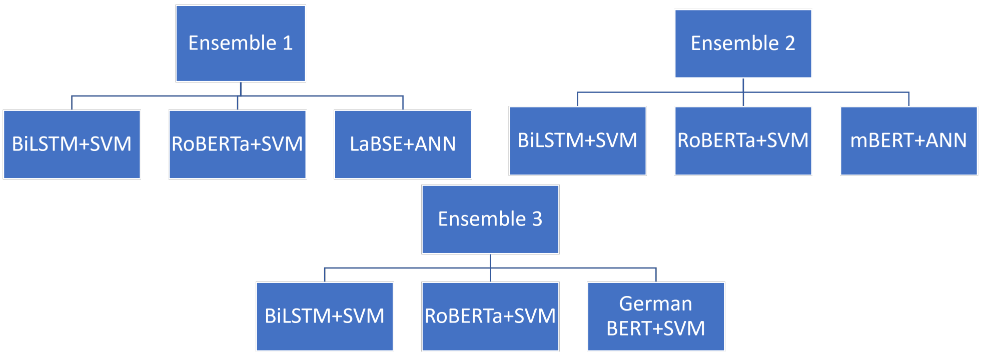


Figure 2: Ensemble models used for this experiment.

four trained annotators (Risch et al., 2021). The dataset was extracted from the home feed of the Facebook page of a political talk show of a German television broadcaster as well as the comment section discussions of posts from the same page from July 2019 till February 2021. It was shared in fully anonymized form and no user information or comment ids were revealed. Links referring to users were replaced by @USER, Links referring to the show were replaced by @MEDIUM, and links referring to the moderator of the show were replaced by @MODERATOR. The csv file contained all comments and labels for all 3 subtasks. That is to say, a user comment can be either toxic, engaging, fact-claiming or any of 2 of the labels or all 3 or neither of the labels (see Figure 1). GermEval 2021 consists of 3 subtasks (Risch et al., 2021). The first subtask is the identification of toxicity or hate speech from German text. The second and the third are the identification of engaging text and fact-claiming text, respectively. Participants were to choose any or all of the tasks they would participate in. We participated in all 3 tasks using a system of 3 different ensembles for each task (see Figure 2 for a quick overview). Submissions of the runs were submitted to Codalab.

3 Models architecture

Over the past few years, Long Short-Term Memory (LSTM) (Huang et al., 2015) and pre-trained transformer-based models (Devlin et al., 2019)

	Sub1-F1	Sub1-P	Sub1-R	Sub2-F1	Sub2-P	Sub2-R	Sub3-F1	Sub3-P	Sub3-R
Ens1	0.9750	1.0000	0.9751	0.9623	0.9273	1.0000	0.9587	0.9508	0.9667
Ens2	0.9402	1.0000	0.9024	0.9714	0.9444	1.0000	0.9594	0.9365	0.9833
Ens3	0.9750	1.0000	0.9512	0.9902	0.9808	1.0000	0.9836	0.9677	1.0000

Table 1: Results on the trial set after training on small dataset.

	Sub1-F1	Sub1-P	Sub1-R	Sub2-F1	Sub2-P	Sub2-R	Sub3-F1	Sub3-P	Sub3-R
Ens1	0.5547	0.5529	0.5565	0.6337	0.6211	0.6468	0.5970	0.5915	0.6026
Ens2	0.5545	0.5550	0.5540	0.6428	0.6406	0.6450	0.6316	0.6241	0.6392
Ens3	0.5559	0.5571	0.5547	0.6143	0.6107	0.6180	0.6150	0.6110	0.6191

Table 2: Results on the test set with models trained on *small* training set (actually submitted runs).

have proven to be effective in various NLP tasks through their ability to generate word or sentence embeddings (Qiu et al., 2020). One of such models that have widely been used in many NLP tasks is the Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2019). It is designed to pre-train deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context to learn and produce embeddings either on sentence or word level in a transformer based architecture. The Flair embedding architecture is also an example of a model that uses a variant of bidirectional recurrent neural networks (BiLSTMs) with a conditional random field (CRF) layer to generate contextual embeddings from both directions (Akbik et al., 2018; Huang et al., 2015). In this experiment, we applied both transformer based models and Bi-directional LSTM (BiLSTM) based models to generate embeddings and further applied a Support Vector Machine (SVM) or a sigmoid activated single-layered neural network as a classifier in an ensemble of 3 models with majority voting – a simple yet effective paradigm (Kanakaraj and Guddeti, 2015; Zimmerman et al., 2018).

Each of the 3 subtasks, that is, identifying toxic, engaging and fact-claiming comments were classified with the same ensemble models. Each ensemble model however, contained three sub-models. The models were run on a standard Google Colabs runtime with a RAM size of 12 gigabyte. Below are the summaries of the sub-models.

3.1 Ensemble 1

For Ensemble 1, a sub-model with emeddings generated from the *flair* framework[2] pre-trained on the German corpus was applied. A forward and backward contextualized embeddings were generated and stacked on top of each other and then mean-pooled. An SVM classifier was fitted to the model with a linear kernel, a regularization parameter of 1, a gamma of 1 and a degree of 3. Embeddings from the XLM-RoBERTa (Conneau et al., 2019) – a multi-lingual BERT-based model designed by Facebook's AI team – was also generated for another sub-model and was also fitted with an SVM classifier with a regularization parameter of 1, a linear kernel, a gamma of 1 and a degree of 3. Finally, the last sub-model applied the language-agnostic BERT-based sentence encoder (LaBSE) with a single layered output of a fully-connected neural network with a sigmoid activation. The sub-models were not fine-tuned on the dataset due to RAM limitations.

3.2 Ensemble 2

Ensemble 2 is very similar to Ensemble 1. The only difference is that one of the sub-models does not use embeddings from a sentence encoder unlike the first Ensemble but rather embeddings were generated from fine-tuning a multilingual BERT (mBERT) and further classified with a sigmoid activated single layered output of a fully-connected neural network. SVM parameters are maintained just as with Ensemble 1.

3.3 Ensemble 3

This Ensemble model applied only SVM classifiers for its sub-models with the same parameters as stated for the other 2 Ensemble models (Hoffmann and Kruschwitz, 2020). However, unlike the other two, the third sub-model of this Ensemble applied a German based BERT model designed by Deepset AI (Chan et al., 2020). No fine-tuning was performed.

[2] https://github.com/flairNLP/flair

	Sub1-F1	**Sub1-P**	**Sub1-R**	**Sub2-F1**	**Sub2-P**	**Sub2-R**	**Sub3-F1**	**Sub3-P**	**Sub3-R**
Ens1	0.7024	0.7957	0.6286	0.7869	0.8536	0.7299	0.7851	0.8280	0.7466
Ens2	0.7577	0.8174	0.7060	0.8389	0.8640	0.8154	0.8148	0.8251	0.8046
Ens3	0.7886	0.8412	0.7422	0.8522	0.8864	0.8206	0.8402	0.8613	0.8201

Table 3: Results on the trial set after training on full dataset.

	Sub1-F1	**Sub1-P**	**Sub1-R**	**Sub2-F1**	**Sub2-P**	**Sub2-R**	**Sub3-F1**	**Sub3-P**	**Sub3-R**
Ens1	0.6205	0.6914	0.5629	0.6721	0.7160	0.6333	0.7211	0.7695	0.6784
Ens2	**0.6472**	0.6936	0.6067	**0.6930**	0.7197	0.6684	**0.7343**	0.7443	0.7247
Ens3	0.6241	0.6574	0.5940	0.6770	0.7023	0.6536	0.7341	0.7596	0.7103

Table 4: Results on the test set with models trained on the *full* training set.

For each of the neural networks applied in Ensemble 1 and 2, the BERT-based embedding layer was fully connected to the output layer. The output layer was activated with a sigmoid function. The neural network had a learning rate of 1-e5 , batch size of 32 and was trained with a model checkpoint on validation loss. The models were setup with 50 training epochs with early stopping on the model checkpoint at a patience of 3 epochs. The training dataset was split for train-test-validation reasons with an initial ratio of 0.8 for training. The remaining 20% was further split into 0.8 and 0.2 for validation and testing respectively. The SVM models were fitted on the whole training data.

4 Results

The results of our officially submitted runs are displayed in Table 2 (and corresponding training performance in Table 1). Note however, that the results submitted were acquired from training on a trial set of 113 comments only – an error which we only noticed after having received the results.

We subsequently re-run the three approaches – this time trained on the full training set – as illustrated in Table 4 (with corresponding training data performance in Table 3). Highest F1 performances are in bold, and we observe that Ensemble 2 consistently performs best.

The results demonstrate that, as expected, an increase in the training data has a measurable positive effect on the overall performance across all metrics.

The results recorded after training shows that the SVM models had very high metrics on the trial set whereas the ANN models had relatively low metrics peaking at 62% for F1 score, precision and recall. An ensemble approach rather seemed balanced. The Ensemble models were slightly biased towards the SVM models because in a total of three models for each ensemble, two models were SVM models for both Ensemble 1 and 2. Ensemble 3 was a model of 3 SVM models. It is fair to say that the SVM models were overfitted on the trial set. The results from the test set were lower than the results for the training data (see Table 2). Considering the fact that the training set of 113 data points is substantially smaller than the test set of 994 entries, it is also not surprising the model performed worse on the test set. The more interesting observation is that even though the training was done for a tiny dataset the results seem better than what one might expect.

Most interesting are of course the findings we derive from running our three approaches on the full training data. We observe *robust performance* of our ensemble-based approaches. We also observe that *fine-tuning* one of the models in our ensembles appears to push up performance quite substantially.

5 Conclusion

Ensemble approaches have repeatedly been shown to offer great benefits but they nevertheless rely on good underlying individual models. In our runs we combined contextual embeddings using state-of-the-art models such as BiLSTM-CRF, BERT-based models and SVM and simple neural networks as classifiers in an ensemble approach to perform binary text classification in German. We observe robust performance across different tasks, we also note a positive impact of including fine-tuned models in our ensembles.

Acknowledgements

This work was supported by the project *COURAGE: A Social Media Companion Safeguarding and Educating Students* funded by the Volkswagen Foundation, grant number 95564.

References

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th international conference on computational linguistics*, pages 1638–1649.

Branden Chan, Stefan Schweter, and Timo Möller. 2020. German's next language model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6788–6796, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Julia Hoffmann and Udo Kruschwitz. 2020. UR_NLP @ haspeede 2 at EVALITA 2020: Towards robust hate speech detection with contextual embeddings. In *Proceedings of the Seventh Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2020), Online event, December 17th, 2020*, volume 2765 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Monisha Kanakaraj and Ram Mohana Reddy Guddeti. 2015. NLP based sentiment analysis on Twitter data using ensemble classifiers. In *2015 3rd international conference on signal processing, communication and networking (ICSCN)*, pages 1–5. IEEE.

Jihyung Moon, Won Ik Cho, and Junbum Lee. 2020. BEEP! Korean Corpus of Online News Comments for Toxic Speech Detection. *arXiv preprint arXiv:2005.12503*.

Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. 2020. Pre-trained models for natural language processing: A survey. *Science China Technological Sciences*, pages 1–26.

Julian Risch, Anke Stoll, Lena Wilms, and Michael Wiegand. 2021. 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 co-located with KONVENS*, pages 1–12.

Marian-Andrei Rizoiu, Tianyu Wang, Gabriela Ferraro, and Hanna Suominen. 2019. Transfer learning for hate speech detection in social media. *arXiv preprint arXiv:1906.03829*.

Kai Shu, Suhang Wang, and Huan Liu. 2018. Understanding user profiles on social media for fake news detection. In *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 430–435.

Julia Maria Struß, Melanie Siegel, Josef Ruppenhofer, Michael Wiegand, Manfred Klenner, et al. 2019. Overview of GermEval Task 2, 2019 Shared task on the identification of offensive language. In *Proceedings of the 15th Conference on Natural Language Processing (KONVENS 2019)*, pages 354–365.

Soroush Vosoughi, Deb Roy, and Sinan Aral. 2018. The spread of true and false news online. *Science*, 359(6380):1146–1151.

Zeerak Waseem and Dirk Hovy. 2016. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *Proceedings of the NAACL student research workshop*, pages 88–93.

Vivian Welch, Jennifer Petkovic, J Pardo Pardo, Tamara Rader, and Peter Tugwell. 2016. Interactive social media interventions to promote health equity: an overview of reviews. *Health promotion and chronic disease prevention in Canada: research, policy and practice*, 36(4):63.

Marcos Zampieri, Preslav Nakov, Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Hamdy Mubarak, Leon Derczynski, Zeses Pitenis, and Çağrı Çöltekin. 2020. SemEval-2020 task 12: Multilingual offensive language identification in social media (OffensEval 2020). *arXiv preprint arXiv:2006.07235*.

Steven Zimmerman, Udo Kruschwitz, and Chris Fox. 2018. Improving hate speech detection with deep learning ensembles. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

HunterSpeechLab at GermEval 2021: Does Your Comment Claim A Fact? Contextualized Embeddings for German Fact-Claiming Comment Classification

Subhadarshi Panda
Hunter College
City University of New York
spanda@gc.cuny.edu

Sarah Ita Levitan
Hunter College
City University of New York
sarah.levitan@hunter.cuny.edu

Abstract

In this paper we investigate the efficacy of using contextual embeddings from multilingual BERT and German BERT in identifying fact-claiming comments in German on social media. Additionally, we examine the impact of formulating the classification problem as a multi-task learning problem, where the model identifies toxicity and engagement of the comment in addition to identifying whether it is fact-claiming. We provide a thorough comparison of the two BERT based models compared with a logistic regression baseline and show that German BERT features trained using a multi-task objective achieves the best F1 score on the test set. This work was done as part of a submission to GermEval 2021 shared task on the identification of fact-claiming comments.[1]

1 Introduction

Deception and misinformation have become increasingly common on social media (Ciampaglia, 2018). With the endless cycle of production and consumption of information, social media users may be influenced to make choices which are unsafe for them and for society. As a result, misinformation media has attracted the interest of the NLP community to develop methods to combat it. Since information on social media is often not filtered based on veracity, it is necessary to apply fact-checking in order to verify the claims made in social media posts and comments (Thorne and Vlachos, 2018). In this paper, we address the problem of automatic fact-claiming comment detection in German, which is a crucial aspect of the fact checking process.

Fact check-worthiness detection has been studied in prior work by using Naive Bayes and SVM classifiers to identify fact check worthy statements in political debates (Hassan et al., 2015). Gencheva et al. (2017); Patwari et al. (2017) find that it is useful to use the context of a statement to detect fact worthiness. Vasileva et al. (2019) frame the problem as a multi-task prediction problem where the model predicts the labels for multiple fact worthiness annotation sources.

In this work we use contextualized embeddings from multilingual BERT and German BERT for detecting fact-claiming comments. We also train our model on two auxiliary tasks of toxicity and engagement prediction in a multi-task learning setup. Upon comparing the methods, we find that the multitask model based on German BERT achieves the best macro-average F-1 score on the test set.

We outline the remaining part of the paper by first describing the dataset used for our experiments in Section 2. Then we discuss the approaches we use in Section 3. Sections 4 and 5 contain details of the experiment setup and the results obtained. Section 6 concludes the paper.

2 Data

We use the data provided for the GermEval 2021 shared task (Risch et al., 2021). The sizes of the training and test splits along with the number of samples in each class are shown in Table 1. Since development data was not provided, we use 5-fold cross validation by splitting the training data into 5 folds using stratified splitting. We then train our models 5 times by picking 4 folds for training and the remaining fold for development. To report evaluation results, we use the mean and standard deviation of the scores on the 5 folds.

3 Methodology

The task is to classify a given comment as fact-claiming or not. It is a binary classification task.

[1] Code is available at https://github.com/subhadarship/GermEval2021.

Data split	Fact-claiming	Non fact-claiming	Total
Train	1103	2141	3244
Test	314	630	944

Table 1: Data statistics.

3.1 Baseline

We first build a linear logistic regression model as our baseline. The comments are converted to TF-IDF feature vectors with unigram word features.

3.2 BERT-based models

BERT (Devlin et al., 2019) is a large language model trained in a self-supervised task of masked language modeling. It was trained using a huge amount of text data from BookCorpus (Zhu et al., 2015) and Wikipedia. It has been shown that fine-tuning pre-trained BERT or using it as a feature extractor leads to improvements in performance in a wide range of downstream NLP tasks (Devlin et al., 2019). In this work which involves data in German, we use multilingual BERT (Devlin et al., 2019) and German BERT (Chan et al., 2020).[2] For both the models, we add a binary classification head on top for classification of comments to whether they are fact-claiming or not. The input to the classification head is the `[CLS]` representation. We minimize the cross-entropy loss during training.

Multilingual BERT Multilingual BERT (mBERT) is a single big language model trained using unsupervised corpora in 104 languages including German. Although not all of the 104 languages are represented with equal quality in mBERT (Pires et al., 2019), mBERT has been shown to perform well on a number of non-English languages (Devlin et al., 2019).

German BERT German BERT is a BERT model trained using a huge amount of German data which includes Wikipedia dump (6 GB), OpenLegalData (2.4 GB) and news articles (3.6 GB). It has been shown to achieve impressive results on German sentiment analysis, document classification and named entity recognition.

3.3 Multi-task training

Multi-task training has been successful in a wide range of NLP tasks where the model predicts the labels for multiple tasks for a given input (Chauhan et al., 2020; Barnes et al., 2021; Goo et al., 2018). We also employ multitask learning for our problem where instead of only predicting whether a comment is fact-claiming or not, the model also predicts whether the comment is toxic and engaging. The idea is that the model learns a representation that is useful for all three of these tasks, which may lead to better overall performance at the primary task which is fact-claiming classification. To do this we add three binary classification heads, one each for fact-claiming, toxicity and engagement prediction, on top of the base BERT model. The training is done with a multi-task objective where the total loss is computed as the sum of the cross entropy losses for the three subtasks.

4 Experiments

In this section we outline the baseline experiments using logistic regression model and experiments using BERT-based models.

4.1 Logistic regression

We used the TF-IDF feature extractor and the logistic regression model implemented in `scikit-learn`. L2 regularization was applied to the model parameters. The coefficient which specifies the inverse of the regularization strength (C parameter) was tuned across five values $\{1, 2, 3, 4, 5\}$. For each hyperparameter setting, 5 systems were trained corresponding to the 5 folds, where in each run 4 folds were used for training and the held out fold was used for evaluation.

4.2 BERT-based models

We used the `bert-base-multilingual-cased` and `bert-base-german-cased` identifiers in Huggingface transformers library (Wolf et al., 2020) for loading the models and tokenizers of pretrained mBERT and German BERT respectively. Both the BERT based models were trained by adding a linear classification layer on top, the hidden size of which was tuned across the values 128, 256, and 512. Optimizer used was Adam with a learning rate value tuned in $\{0.05, 0.005, 0.0005\}$. Gradients greater than 1 were clipped during training. The parameters of the

[2] https://www.deepset.ai/german-bert

Hyperparameter (C)	Train F-1	Dev F-1
1.0	0.800 ± 0.00	0.688 ± 0.00
2.0	0.876 ± 0.00	0.701 ± 0.01
3.0	0.919 ± 0.00	0.709 ± 0.01
4.0	0.946 ± 0.00	0.707 ± 0.01
5.0	**0.964** ± 0.00	**0.711** ± 0.01

Table 2: Baseline results using logistic regression model with TF-IDF features.

Model	F-1	Precision	Recall
Monotask mBERT	0.737 ± 0.01	0.745 ± 0.01	0.735 ± 0.01
Monotask German BERT	**0.762** ± 0.01	**0.774** ± 0.02	**0.754** ± 0.01
Multitask mBERT	0.737 ± 0.01	0.742 ± 0.01	0.734 ± 0.01
Multitask German BERT	0.759 ± 0.01	**0.774** ± 0.02	0.751 ± 0.01

Table 3: BERT based model results on dev data.

BERT layers were either frozen or were fine-tuned during the training process. We found that freezing the parameters of the BERT layers resulted in better scores consistently. Training was stopped when the development macro average F-1 score did not improve for 10 consecutive epochs. Similar to the logistic regression model training, 5 systems were trained for each hyperparameter setting using the 5 fold cross-validation data.

We found that for both mBERT and German BERT, less than 0.5% of the running tokens were out-of-vocabulary. However the best system for mBERT required 34 epochs to converge whereas the best system for German BERT required only 21 epochs to converge, where the best systems were decided based on the average cross-validation macro-average F-1 score. There was almost no difference in training time in epochs for monotask vs multitask best systems of German BERT.

5 Results

The predictions were evaluated using the macro-average F1 score. For the dev set results, the macro-average F-1 score is computed using the implementation provided in `scikit-learn` whereas for test set results the macro-average F-1 score is computed using the evaluation script provided by the organizers of the shared task.[3] The difference between the two is that while the former computes the arithmetic mean of the F-1 scores for each class, the latter computes the arithmetic mean of the precisions and the arithmetic mean of the recalls for each class which are then used to compute F-1 score using $F1 = \frac{2 \times P \times R}{P+R}$.[4]

5.1 Results on dev data

The mean and standard deviation of the F-1 scores for each hyperparameter setting of the logistic regression baseline model are shown in Table 2. The best training and development F-1 scores are obtained using $C = 5.0$.

The BERT-based results using monotask training and multitask training are shown in Table 3. All the results shown are for the case where the BERT layers were frozen. The best set of hyperparameters are hidden size 512 and learning rate 0.0005.

5.2 Results on test data

For a given model, we only used one of the 5 cross-validation trained systems to predict the labels of the test set. The test set results are shown in Table 4. Notably the logistic regression's F-1 and precision scores are higher than monotask mBERT scores. The German BERT scores are higher than both the logistic regression baseline and monotask mBERT. The overall best score is obtained using the multitask German BERT which achives 71.5% F-1 score, 72.72% precision and 70.32% recall.

5.3 Analysis of results

We analyze the test results using the confusion matrices of the three submitted BERT-based systems (see Figure 1). All the three systems have very close true negatives (149, 148 and 143). However, the multitask German BERT has the lowest false positives (87) as compared to monotask mBERT (107) and monotask German BERT (95).

[3] Link here.

[4] The two approaches result in almost identical scores.

System	F-1	Precision	Recall
Logistic regression	0.6873	0.7013	0.6738
Monotask mBERT	0.6851	0.6924	0.6778
Monotask German BERT	0.6991	0.7097	0.6889
Multitask German BERT	**0.7150**	**0.7272**	**0.7032**

Table 4: Results on the test set. Only the 3 BERT based systems were submitted for shared task evaluation.

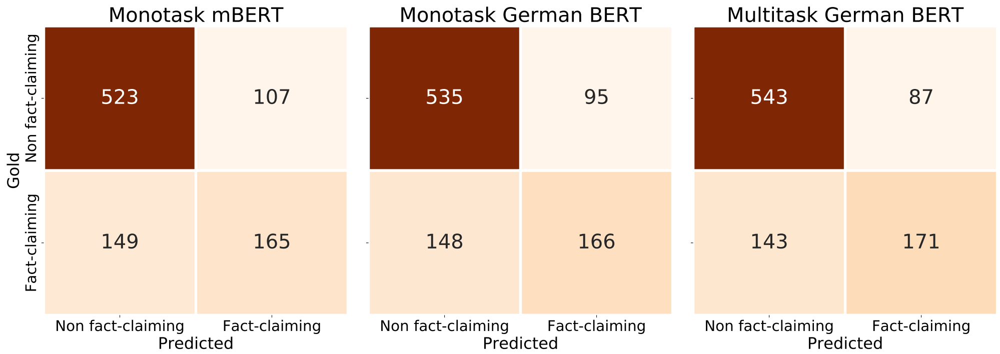


Figure 1: Confusion matrix on the test set using different BERT based systems.

6 Conclusion

In this paper we presented a baseline model based on logistic regression and two BERT-based models for identifying whether German comments in social media are fact-claiming or not. We thoroughly compared the two models: multilingual BERT which is pre-trained on 104 languages, and German BERT which is pre-trained only on German data. We also formulated the learning problem as a multitask problem by addition of two auxiliary tasks of toxicity and engagement classification. The multitask German BERT achieved the best results on the test set. This work contributes models and insights for detecting fact-claiming comments on social media, which is an important step towards hopefully combating misinformation that is pervasive on social media.

References

Jeremy Barnes, Erik Velldal, and Lilja Øvrelid. 2021. Improving sentiment analysis with multi-task learning of negation. *Natural Language Engineering*, 27(2):249–269.

Branden Chan, Stefan Schweter, and Timo Möller. 2020. German's next language model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6788–6796, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Dushyant Singh Chauhan, Dhanush S R, Asif Ekbal, and Pushpak Bhattacharyya. 2020. Sentiment and emotion help sarcasm? a multi-task learning framework for multi-modal sarcasm, sentiment and emotion analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4351–4360, Online. Association for Computational Linguistics.

Giovanni Luca Ciampaglia. 2018. *The Digital Misinformation Pipeline*, pages 413–421. Springer Fachmedien Wiesbaden, Wiesbaden.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Pepa Gencheva, Preslav Nakov, Lluís Màrquez, Alberto Barrón-Cedeño, and Ivan Koychev. 2017. A context-aware approach for detecting worth-checking claims in political debates. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 267–276, Varna, Bulgaria. INCOMA Ltd.

Chih-Wen Goo, Guang Gao, Yun-Kai Hsu, Chih-Li Huo, Tsung-Chieh Chen, Keng-Wei Hsu, and Yun-Nung Chen. 2018. Slot-gated modeling for joint slot filling and intent prediction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 753–757, New Orleans, Louisiana. Association for Computational Linguistics.

Naeemul Hassan, Chengkai Li, and Mark Tremayne. 2015. Detecting check-worthy factual claims in presidential debates. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, CIKM '15, page 1835–1838, New York, NY, USA. Association for Computing Machinery.

Ayush Patwari, Dan Goldwasser, and Saurabh Bagchi. 2017. *TATHYA: A Multi-Classifier System for Detecting Check-Worthy Statements in Political Debates*, page 2259–2262. Association for Computing Machinery, New York, NY, USA.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

Julian Risch, Anke Stoll, Lena Wilms, and Michael Wiegand. 2021. 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 co-located with KONVENS*, pages 1–12.

James Thorne and Andreas Vlachos. 2018. Automated fact checking: Task formulations, methods and future directions. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3346–3359, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Slavena Vasileva, Pepa Atanasova, Lluís Màrquez, Alberto Barrón-Cedeño, and Preslav Nakov. 2019. It takes nine to smell a rat: Neural multi-task learning for check-worthiness prediction. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 1229–1239, Varna, Bulgaria. INCOMA Ltd.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, and S. Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 19–27.

FHAC at GermEval 2021: Identifying German toxic, engaging, and fact-claiming comments with ensemble learning

Tobias Bornheim[1], Niklas Grieger[1] and Stephan Bialonski[1,2,*]

[1] *Department of Medical Engineering and Technomathematics*
FH Aachen University of Applied Sciences, Jülich, Germany

[2] *Institute for Data-Driven Technologies*
FH Aachen University of Applied Sciences, Jülich, Germany
[*] *bialonski@fh-aachen.de*

Abstract

The availability of language representations learned by large pretrained neural network models (such as BERT and ELECTRA) has led to improvements in many downstream Natural Language Processing tasks in recent years. Pretrained models usually differ in pretraining objectives, architectures, and datasets they are trained on which can affect downstream performance. In this contribution, we fine-tuned German BERT and German ELECTRA models to identify toxic (subtask 1), engaging (subtask 2), and fact-claiming comments (subtask 3) in Facebook data provided by the GermEval 2021 competition. We created ensembles of these models and investigated whether and how classification performance depends on the number of ensemble members and their composition. On out-of-sample data, our best ensemble achieved a macro-F1 score of 0.73 (for all subtasks), and F1 scores of 0.72, 0.70, and 0.76 for subtasks 1, 2, and 3, respectively.

1 Introduction

Social media plays a role in the spreading of problematic content, ranging from conspiracy theories and concerted misinformation campaigns to offensive language in user comments (Zhuravskaya et al., 2020). Moderating comments remains a challenge due to the ever-increasing amount of user-generated content created daily. One promising approach to addressing this challenge are techniques from Natural Language Processing (NLP) that support manual moderation processes by, for example, alerting human moderators to potentially problematic comments.

Among the many factors that have driven recent progress in NLP, we note in particular (i) methodological advances in language modeling and (ii) the availability of annotated data due to shared tasks. Recent methodological advances can be traced back to the invention and availability of deep neural network models. A major contribution was the invention of the transformer architecture, which harnesses self-attention mechanisms to effectively model long-range correlations in series of tokens (e.g., sentences) (Vaswani et al., 2017). Based on the transformer architecture, neural network models such as BERT (Devlin et al., 2019; Rogers et al., 2020) were proposed and trained in a self-supervised fashion on large unannotated text corpora. Language representations learned by BERT turned out to be effective in many downstream tasks, leading to new state-of-the-art NLP systems. While *masked language modeling* and *next sentence prediction* are used as objectives in self-supervised pretraining to learn representations in BERT, other pretraining objectives such as *replaced token detection* (ELECTRA, Clark et al. (2020)) have been demonstrated to yield language representations that can be better suited for various downstream tasks (Xia et al., 2020). Furthermore, language representations have been learned in multilingual language models (such as mBERT) and in language specific BERT models (Nozza et al., 2020). Recent German specific language models include the BERT-based models GBERT (Chan et al., 2020) and GottBERT (Scheible et al., 2020) as well as the ELECTRA based model GELECTRA (Chan et al., 2020).

The second factor driving progress in NLP has been recurring shared tasks that foster the exchange of ideas, the development and comparative assessment of methods, as well as the availability of annotated data (Nissim et al., 2017). In addition to multilingual shared task campaigns (see, e.g., Mandl et al. (2019); Basile et al. (2019)), there exist language-specific shared task evaluations such as GermEval which focus on NLP for the German language. A series of GermEval tasks addressed the challenge of reliably identifying offensive language

Frau Barley war mit ihrem dummdreisten überheblichen Grinsen wirklich nicht zu ertragen. (TOXIC)

Da dreht sich jemand im Kreis. Die 7 Prozent kann der Vermieter doch auf die Miete schlagen. Diese starke Position ergibt sich durch den Markt (Angebot und Nachfrage), da ist es egal. [...] (ENGAGING, FACT-CLAIMING)

Figure 1: Samples (Facebook comments) from the dataset of the GermEval 2021 shared task.

(Wiegand et al., 2018) and distinguishing between profane, offensive, or abusive language found in Twitter tweets (Struß et al., 2019). The GermEval 2021 shared task on identifying toxic, engaging, and fact-claiming comments (Risch et al., 2021) provided German comments from a Facebook page of a political talk show of a German television broadcaster.

In this contribution, we investigate the ability of ensembles of GBERT and GELECTRA models to identify toxic, engaging, and fact-claiming comments. Our work was inspired by previous studies on German BERT models (Graf and Salini, 2019) and ensemble approaches (Risch and Krestel, 2018, 2020). We study the dependence of classification performance on the number of ensemble members and ensemble composition. Finally, we describe the models that were evaluated in the GermEval 2021 shared tasks and report performance scores achieved on out-of-sample data. The implementation details of our experiments are available online[1].

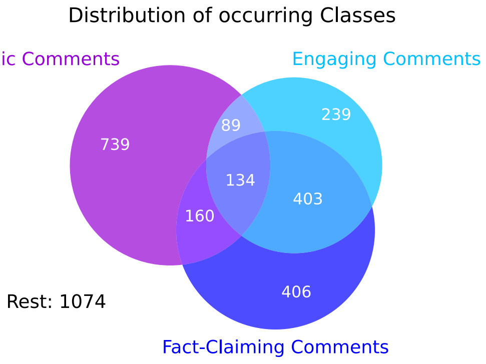


Figure 2: Venn diagram showing the numbers of comments that were labeled as toxic, engaging, or fact-claiming. 33 % of all comments were not assigned any class, whereas 24 % were attributed to more than one class.

2 Data and tasks

The dataset of the shared task consisted of 3244 annotated Facebook comments and was provided by the organizers of GermEval 2021 (Risch et al., 2021). The comments were drawn from a Facebook page of a political talk show of a German television broadcaster from February till July 2019 and were anonymized by replacing links to users by @USER, links to the show by @MEDIUM, and links to the moderator of the show by @MODERATOR. Four trained annotators labeled the data by three categories, indicating toxic, engaging, and fact-claiming comments (see figure 1).

The shared task consisted of three binary classification subtasks that aimed at predicting whether a given comment belonged to a category (class) or not (Risch et al., 2021). Comments were considered toxic (subtask 1) when they could violate the rules of polite behavior or violated democratic discourse values. Automated identification of such comments can be particularly valuable for managers of online communities. Comments were considered engaging (subtask 2) when they were in line with deliberative principles such as rationality, reciprocity, and mutual respect. Such comments might encourage user engagement and could be made more visible in online communities. Finally, comments were considered fact-claiming (subtask 3) if they contained assertion of facts and/or provided evidence by citing external sources. Identifying such comments may constitute a preprocessing step that could assist community managers to filter out misinformation and fake news.

We did not observe any major class imbalance (Haixiang et al., 2017) as all three classes occurred with similar frequencies in the dataset (35% toxic, 27% engaging, 34% fact-claiming). However, the Venn diagram (see figure 2) demonstrated significant overlap between classes where 24% of all comments were attributed to more than one class. For instance, the large overlap between the engaging and the fact-claiming classes may point towards a correlation between these two classes. Such label correlations can be exploited by *multi-label classification* approaches to improve

[1] https://github.com/fhac-fb9-ds/germeval2021

classification performance (Zhang and Zhou, 2014). Thus we pursued a two-fold strategy. (i) In our first approach, we trained a multi-label classifier to predict all the possible class attributions for a given comment. Such models are called *multi-label* in the following. (ii) In our second approach, we trained separate binary classifiers that aimed at distinguishing between toxic and non-toxic, engaging and non-engaging, or fact-claiming and non-fact-claiming classes, respectively. This approach of transforming a multi-label classification task into multiple single-label classification tasks is also known as a *binary relevance transformation* (Zhang and Zhou, 2014). We call such models *single-label* in the following. Note that in this case, training a size 30 ensemble to classify comments means training three separate size 30 ensembles, each making predictions for one of the three binary classification tasks.

3 Methods

3.1 Preprocessing and data splits

Preprocessing. All data (i.e., training and test data) was preprocessed as follows. First, all duplicates in the training data were removed, reducing the 3244 training samples to 3226 unique samples. In a second step, all in-word whitespaces were removed (e.g. transforming the sequence “A K T U E L L !” into the word “AKTUELL!”) (Paraschiv and Cercel, 2019). Third, emojis were buffered with additional whitespaces such that words immediately followed by an emoji were not tokenized as unknown and emojis were tokenized separately (e.g., transforming the sentence “I always start my day with a coffee☕☕☕” into “I always start my day with a coffee ☕ ☕ ☕”) (Risch and Krestel, 2020). Fourth, any leading, trailing or consecutive whitespaces were removed. Last, all comments were limited to a maximum length of 200 tokens to save computational resources and speed up training. Only 49 out of the 3226 unique sentences in the training data and 21 out of 944 sentences in the test data were affected by this step.

Data splits. During model exploration, models were trained with a 5-fold cross validation scheme (i.e., with 5 folds, each containing 20% of the randomly shuffled training data). The final models evaluated by GermEval 2021 were trained on all training data (i.e., on all folds) to optimize model fitting. Furthermore, during model exploration as well as for the final models, 10% of the data in the training folds was randomly selected to act as an *early stopping set* (see section 3.3) that was not used for training.

3.2 Models

We studied two recent transformer-based German language models (Chan et al., 2020) called GBERT, based on the BERT architecture (Devlin et al., 2019), and GELECTRA, based on the ELECTRA architecture (Clark et al., 2020). Both models use a tokenizer with a vocabulary size of 31k cased words. From the different pretrained versions of these models, we chose gbert-large[2] and gelectra-large[3], both with a hidden states count of 1024.

A classification head was added on top of the first output vector of both pretrained transformer models. In the GBERT architecture, the mentioned output vector was generated by inserting a classification token at the beginning of every input sequence, which is used for the *next sentence prediction* task during pretraining (Devlin et al., 2019). The classification head consisted of a linear layer with the same hidden size as the transformer model, followed by a tanh activation function and another linear layer (Wolf et al., 2020). Although the GELECTRA architecture does not use any *next sentence prediction* task during pretraining (Clark et al., 2020), a classification token is still prepended to the transformer input and can be used during fine-tuning. The classification head of GELECTRA had the same architecture as that of the GBERT model, except that a GELU activation (Hendrycks and Gimpel, 2016) was used instead of a tanh activation (Wolf et al., 2020).

All linear layers of both classification heads were initialized randomly, except for the first layer of the GBERT classifier, which was initialized with the weights learned during the pretraining task. Depending on whether the models were single-label or multi-label classifiers, the final linear layer consisted of either two outputs followed by a softmax function or three outputs followed by a sigmoid function.

3.3 Training

Evaluation scores. To evaluate the prediction performance of a model, we determined the F1 score

[2] `https://huggingface.co/deepset/gbert-large`

[3] `https://huggingface.co/deepset/gelectra-large`

following the definition used throughout the GermEval shared tasks (Wiegand, 2021). In GermEval, the F1 score of a binary classifier is determined by calculating precision and recall for the positive class (e.g., "toxic") and for the negative class (e.g. "non-toxic"). Precision and recall are then averaged over the two classes. The F1 score is calculated as harmonic mean over averaged recall and averaged precision. By taking the arithmetic mean of F1 scores of each binary classifier, we obtained the macro-F1 score $\overline{F1} = \frac{1}{3}(F1_{toxic} + F1_{engaging} + F1_{fact})$. During model exploration, $\overline{F1}$ scores were determined for all five validation folds, and their mean and standard deviation were determined. We considered a model to be superior to other models if its $\overline{F1}$ score averaged over all validation folds (of the cross validation) was larger than those of the other models.

Training scheme. Each model (i.e., transformer with classification head) was trained with a batch size of 24 samples for 10 epochs using the AdamW optimizer (Loshchilov and Hutter, 2019). We used a learning rate of $\eta = 5 \cdot 10^{-6}$ with a linear warmup on the first 30% of the training steps from 0 to η. Every 40 updates of the gradients, the models were evaluated on the early stopping data by calculating the macro-F1 score. If the score did not increase for two consecutive evaluations the training was interrupted and the model achieving the largest F1 score on the early stopping set was used for evaluation on the validation fold or test data.

Loss functions. When training single-label models, we used a negative log-likelihood loss function. Multi-label models were trained by minimizing the binary cross entropy loss function averaged over the three classes for every sample in a mini-batch.

Threshold selection. In multi-label models, a sample (comment) from the dataset was predicted to belong to those classes for which the respective class probabilities of the model exceeded a certain threshold. Since multi-label models can attribute a sample to three classes, three different thresholds needed to be determined. We chose these thresholds by evaluating the model for threshold values between 0 and 1 (exploring this range with a step size of 0.025) on the data reserved for early stopping and accepting the values achieving the highest F1 scores for each class separately. In single-label models, we did not need to chose any thresholds since the class membership of a sample was predicted by identifying the largest output probability of the two output neurons.

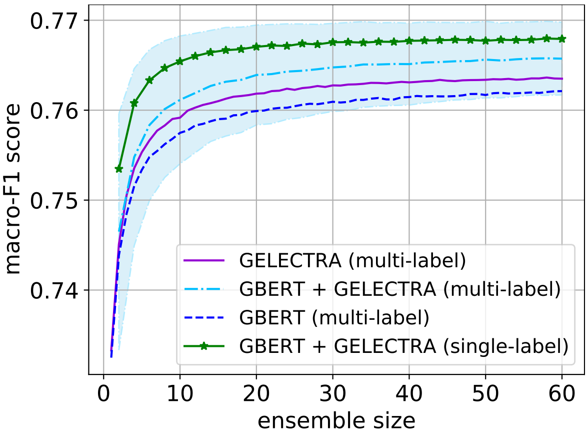


Figure 3: Dependence of the average macro-F1 score (lines) on ensemble size for different ensemble compositions. Standard deviations are shown as blue shaded area for the multi-label ensemble GBERT/GELECTRA. Note that for ensemble sizes larger than 30, average macro-F1 scores differed between ensembles only in their third decimal place, a variation that we considered insignificant.

3.4 Ensembling

Training complex models such as GBERT or GELECTRA on small datasets can lead to overfitting. Following the work by Risch and Krestel (2020), we counteracted this phenomenon by creating ensembles of models using bootstrap aggregation. Ensemble members differed in the initial weights of the classification layers and the data samples randomly selected for early stopping. The predictions of an ensemble were determined by averaging the predicted probabilities of the ensemble members (*soft majority voting*). In single-label models, a model's prediction was then determined by identifying the output neuron associated with the largest ensemble-averaged output probability. In multi-label models, a model predicted a sample to belong to certain classes if ensemble-averaged class probabilities exceeded optimal thresholds. The optimal thresholds were determined by evaluating each ensemble member for all thresholds on the early stopping data (see section 3.3) and accepting the thresholds with the highest macro-F1 score as the optimal values.

4 Results

Model exploration. We investigated whether and how classification performance (quantified by

	$\overline{\mathbf{F1}}$	$\mathbf{F1}_{\text{toxic}}$	$\mathbf{F1}_{\text{engaging}}$	$\mathbf{F1}_{\text{fact}}$
model exploration				
50 GELECTRA multi-label	0.765 (0.008)	0.730 (0.018)	0.782 (0.018)	0.784 (0.019)
50 GBERT multi-label	0.760 (0.002)	0.720 (0.006)	0.777 (0.015)	0.782 (0.013)
25+25 GELECTRA/GBERT multi-label	0.763 (0.007)	0.726 (0.010)	0.780 (0.015)	0.784 (0.015)
25+25 GELECTRA/GBERT single-label	0.768 (0.006)	0.736 (0.011)	0.782 (0.014)	0.787 (0.013)
final submissions				
200 GELECTRA multi-label	0.717	0.713	0.690	0.748
200+200 GELECTRA/GBERT multi-label	**0.726**	0.716	**0.699**	**0.763**
30+30 GELECTRA/GBERT single-label	0.699	**0.718**	0.658	0.723
corrected scores	**0.727**	**0.717**	0.697	**0.768**

Table 1: F1 scores achieved by different ensembles during model exploration on the validation folds (rows 1–4; mean and standard deviation over the folds) and F1 scores achieved by the submitted models on the test data as reported by the GermEval 2021 organizers (rows 5–7; best scores are shown in bold). The *corrected scores* shown in the last row were calculated after correcting an error identified after submission.

macro-F1 score) depended on (i) ensemble size, (ii) ensemble composition, and (iii) on whether ensemble members can exploit label correlations (multi-label models) or not (single-label models). To study the effect of (ii), we compared the classification performance of different ensemble compositions. The first ensemble consisted of fine-tuned multi-label GELECTRA models, while the second ensemble consisted of fine-tuned multi-label GBERT models. In a third ensemble we used equal parts of fine-tuned multi-label GELECTRA and GBERT models. To study the effect of (iii), we compared the third ensemble with a fourth ensemble which was composed of equal parts of fine-tuned single-label GELECTRA and GBERT models. Finally, we investigated the effect of (i) via a bootstrap experiment following Risch and Krestel (2020).

The bootstrap experiment was carried out using a 5-fold cross validation scheme. We trained 100 models each of multi-label GBERT and multi-label GELECTRA, and 50 models each of single-label GBERT and single-label GELECTRA on each cross-validation split. For a given ensemble size, we created 1000 ensembles by randomly sampling with replacement from the set of trained models. Each ensemble made predictions on a validation fold by *soft majority voting*. The average macro-F1 score of an ensemble was determined by averaging the macro-F1 scores obtained on each of the 5 validation folds. Thus, for a given ensemble size, we obtained 1000 average macro-F1 scores.

Figure 3 shows the mean of the average macro-F1 scores obtained for different ensemble sizes and ensemble compositions. We observed classification performance to increase with ensemble size, irrespective of model composition and of whether models could or could not exploit label correlations. Largest increases were found for ensemble sizes up to 15 ensemble members, which is consistent with a previous study on a different classification task (Risch and Krestel, 2020). Moreover, macro-F1 scores continued to increase beyond the ensemble size of 15.

For a given ensemble size larger than 30, classification performance between ensembles of the different compositions varied only in the third decimal of their macro-F1 score, a variation that we did not consider significant. Ensembles consisting of 100% GELECTRA models, 100% GBERT models, or 50% GELECTRA and 50% GBERT models yielded comparable macro-F1 scores. Likewise, ensembles consisting of either multi-label or single-label models showed comparable macro-F1 scores for a fixed ensemble size. These observations were confirmed by F1 scores obtained for ensembles of size 50, reported in table 1 (rows 1–4).

Submitted models. Three ensembles were submitted and evaluated on the test data of the shared tasks reflecting the lines of investigation laid out before. The evaluated ensembles were (1) an ensemble of 200 multi-label GELECTRA models, (2) an ensemble of 200 multi-label GELECTRA and 200 multi-label GBERT models, and (3) an ensemble of 30 single-label GELECTRA and 30 single-label GBERT models which were trained on all the training data (see section 3.3). We note that time and computational constraints limited ensemble sizes.

On the test data of the shared task, ensemble (2) achieved the largest macro-F1 score of 0.73, followed by ensemble (1) with 0.72 and (3) with 0.70 (see table 1, rows 5–7). We identified a software bug after submission deadline that affected the scores calculated for ensemble (3) which achieved a corrected macro-F1 score of 0.73. These results supported observations made during model exploration that ensemble composition and classification type did not significantly affect classification performance for ensemble sizes larger than 30.

5 Conclusion

We trained ensembles of fine-tuned German language models, namely GELECTRA and GBERT, to classify German toxic, engaging, and fact-claiming comments in the GermEval 2021 shared task. We investigated whether classification performance (quantified by macro-F1 scores) depended on (i) ensemble size, (ii) ensemble composition, or (iii) whether models were trained as multi-label classifiers (and thus potentially exploiting label correlations) or as single-label classifiers. We observed that ensemble size had a significant effect on classification performance, with more ensemble members leading to better macro-F1 scores, consistent with previous observations by Risch and Krestel (2020) on a different dataset. Neither ensemble composition nor model classification type (multi- or single-label) showed significant different classification performance for the studied parameters when the ensemble size was larger than 30. Two ensembles achieved the largest macro-F1 score (0.73) on the test data, namely the multi-label and single-label ensembles consisting of GELECTRA and GBERT models.

References

Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. 2019. SemEval-2019 task 5: Multilingual detection of hate speech against immigrants and women in Twitter. In *Proc. 13th Int. Workshop on Semantic Evaluation*, pages 54–63, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Branden Chan, Stefan Schweter, and Timo Möller. 2020. German's next language model. In *Proc. 28th Int. Conf. on Computational Linguistics, COLING 2020*, pages 6788–6796, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *8th Int. Conf. on Learning Representations, ICLR 2020*, Addis Ababa, Ethiopia. OpenReview.net.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proc. 2019 Conf. North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019*, volume 1, pages 4171–4186, Minneapolis, MN, USA. Association for Computational Linguistics.

Tim Graf and Luca Salini. 2019. bertZH at GermEval 2019: Fine-grained classification of German offensive language using fine-tuned BERT. In *Proc. 15th Conf. on Natural Language Processing, KONVENS 2019*, Erlangen, Germany.

Guo Haixiang, Li Yijing, Jennifer Shang, Gu Mingyun, Huang Yuanyue, and Gong Bing. 2017. Learning from class-imbalanced data: Review of methods and applications. *Expert. Syst. Appl.*, pages 220–239.

Dan Hendrycks and Kevin Gimpel. 2016. Bridging nonlinearities and stochastic regularizers with gaussian error linear units. *CoRR*, abs/1606.08415.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th Int. Conf. on Learning Representations, ICLR 2019*, New Orleans, LA, USA. OpenReview.net.

Thomas Mandl, Sandip Modha, Prasenjit Majumder, Daksh Patel, Mohana Dave, Chintak Mandlia, and Aditya Patel. 2019. Overview of the HASOC track at FIRE 2019: Hate speech and offensive content identification in indo-european languages. FIRE 19, page 14–17, New York, NY, USA. Association for Computing Machinery.

Malvina Nissim, Lasha Abzianidze, Kilian Evang, Rob van der Goot, Hessel Haagsma, Barbara Plank, and Martijn Wieling. 2017. Sharing is caring: The future of shared tasks. *Comput. Linguistics*, 43(4).

Debora Nozza, Federico Bianchi, and Dirk Hovy. 2020. What the [mask]? Making sense of language-specific BERT models. *CoRR*, abs/2003.02912.

Andrei Paraschiv and Dumitru-Clementin Cercel. 2019. UPB at GermEval-2019 task 2: BERT-based offensive language classification of German tweets. In *Proc. 15th Conf. on Natural Language Processing, KONVENS 2019*.

Julian Risch and Ralf Krestel. 2018. Aggression identification using deep learning and data augmentation. In *Proc. 1st Workshop on Trolling, Aggression and Cyberbullying, TRAC@COLING 2018*, pages 150–158, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Julian Risch and Ralf Krestel. 2020. Bagging BERT models for robust aggression identification. In *Proc. 2nd Workshop on Trolling, Aggression and Cyberbullying, TRAC@LREC 2020*, pages 55–61, Marseille, France. European Language Resources Association (ELRA).

Julian Risch, Anke Stoll, Lena Wilms, and Michael Wiegand. 2021. Overview of the GermEval 2021 shared task on the identification of toxic, engaging, and fact-claiming comments. In *Proc. GermEval 2021 Shared Task on the Identification of Toxic, Engaging, and Fact-Claiming Comments co-located with KONVENS*, pages 1–12.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in BERTology: What we know about how BERT works. *Trans. Assoc. Comput. Linguistics*, 8:842–866.

Raphael Scheible, Fabian Thomczyk, Patric Tippmann, Victor Jaravine, and Martin Boeker. 2020. Gottbert: A pure German language model. *CoRR*, abs/2012.02110.

Julia Maria Struß, Melanie Siegel, Josef Ruppenhofer, Michael Wiegand, and Manfred Klenner. 2019. Overview of GermEval task 2, 2019 shared task on the identification of offensive language. In *Proc. 15th Conf. on Natural Language Processing, KONVENS 2019*, Erlangen, Germany.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Annual Conf. Neural Information Processing Systems 2017*, pages 5998–6008, Long Beach, CA, USA.

Michael Wiegand. 2021. Personal communication.

Michael Wiegand, Melanie Siegel, and Josef Ruppenhofer. 2018. Overview of the GermEval 2018 shared task on the identification of offensive language. In *Proc. GermEval 2018, 14th Conf. on Natural Language Processing (KONVENS2018)*, pages 1–10. Austrian Academy of Sciences.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proc. 2020 Conf. on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Patrick Xia, Shijie Wu, and Benjamin Van Durme. 2020. Which *BERT? A survey organizing contextualized encoders. In *Proc. 2020 Conf. on Empirical Methods in Natural Language Processing, EMNLP 2020*, pages 7516–7533, Online. Association for Computational Linguistics.

Min-Ling Zhang and Zhi-Hua Zhou. 2014. A review on multi-label learning algorithms. *IEEE T. Knowl. Data En.*, 26:1819–1837.

Ekaterina Zhuravskaya, Maria Petrova, and Ruben Enikolopov. 2020. Political effects of the internet and social media. *Annual Review of Economics*, 12:415–438.