Fraunhofer SIT at CheckThat! 2023: Enhancing the Detection of Multimodal and Multigenre Check-Worthiness Using Optical Character Recognition and Model Souping

Notebook for the CheckThat! Lab at CLEF 2023

Raphael Antonius Frick[1,*,†], Inna Vogel[1,†] and Jeong-Eun Choi[1,†]

[1]Fraunhofer Institute for Secure Information Technology SIT | ATHENE — National Research Center for Applied Cybersecurity, Rheinstrasse 75, Darmstadt, 64295, Germany,
url=https://www.sit.fraunhofer.de/

Abstract

This paper describes the approach developed by the Fraunhofer SIT team in the CLEF-2023 CheckThat! lab challenge for check-worthiness detection in multimodal and unimodal content. Check-worthiness detection aims to facilitate manual fact-checking efforts by prioritizing the statements that fact-checkers should consider first. It can also be seen as the first step of a fact-checking system. Our approach was ranked first in Task 1A and second in Task 1B. The goal of Task 1A is to determine whether a claim in a tweet that contains both a snippet of text and an image is worth fact-checking. For this task, we propose a novel way to detect check-worthiness. It takes advantage of two classifiers, each trained on a single modality. For image data, extracting the embedded text with an OCR analysis has shown to perform best. By combining the two classifiers, the proposed solution was able to place first in Task 1A with an F_1 score of 0.7297 achieved on the private test set. The aim of Task 1B is to determine whether a text snippet from a political debate it should be assessed for check-worthiness. Our best-performing method takes advantage of an ensemble classification scheme centered on Model Souping. When applied to the English data set, our submitted model achieved an overall F_1 score of 0.878 and was ranked as the second-best model in the competition.

Keywords

Check-Worthiness Detection, Multimodality, Optical Character Recognition, Model Souping

1. Introduction

In today's digitally connected world, social media platforms have become leading channels for the dissemination of information and play a crucial role in shaping public opinion. However, the proliferation of fake news and false information poses a major challenge to the reliability and trustworthiness of content disseminated on these platforms. To combat such false or misleading information, several manual fact-checking initiatives have been launched, such as:

CLEF 2023: Conference and Labs of the Evaluation Forum, September 18–21, 2023, Thessaloniki, Greece

*Corresponding author.

raphael.frick@sit.fraunhofer.de (R. A. Frick); inna.vogel@sit.fraunhofer.de (I. Vogel); jeong-eun.choi@sit.fraunhofer.de (J. Choi)

FactCheck.org[1], PolitiFact[2] or Snopes[3]. With billions of data shared on social media platforms every second, it is even for computers infeasible to review all of the data. Therefore, automatic identification of most worthy and prioritized claims for fact-checking can be very useful for human experts. The check-worthiness task can be considered as the first of three steps in the fact-checking pipeline, which traditionally consists of [1]:

1. Detect check-worthy statements in a text.
2. Retrieve claims that could be useful to fact-check and that have been verified in the past
3. Automated veracity estimation.

While much attention has been paid to the detection of review-worthy tweets and political debates in text form [2, 3], detecting check-worthiness in content that includes both images and text is still a relatively unexplored area with only a few publications tackling this issue [4]. Not only is multimedia content frequently shared with text on social media these days, but it can also assist in the spread of disinformation. They can serve to attract the reader's attention, but also contain false information. For example, images and videos can be taken out of context and used in a new context, or be manipulated using AI-assisted tools or manual retouching. In some cases in the past, images consisting only of text were posted without a descriptive text to circumvent the automatic reporting mechanisms of social media platforms such as Facebook. This demonstrates the need to extend the check-worthiness estimation from text-only data to multimodal data.

The CheckThat! Lab has been tackling this scientific problem for the past several years. This year, CheckThat! Lab [5, 6] offered two kinds of data for the check-worthiness subtask [7]. For Subtask 1A (multimodal), a text snippet (tweet) plus an image had to be assessed for check-worthiness. The aim of Task 1B (Multigenre) was to identify check-worthy statements from a tweet or a political debate/ speech transcription. Fraunhofer SIT participated in Task 1A and 1B of the CLEF 2023 CheckThat! Lab Challenge for the English language. We achieved first place in Task 1A and second place in Task 1B. This paper describes both approaches for identifying relevant claims in English multimodal tweets and political debates.

Our proposed methodology for Subtask 1A involves a multimodal approach that combines textual cues in the provided images and descriptive texts, and that uses a pair of BERT-based transformation models to extract meaningful features from them. The classifier for Subtask 1B is based on an ensemble learning scheme to improve upon the uncertainty of single classifiers. Since traditional stacking-based ensemble classifiers cause high computational overhead leading to long inference times, they are not always suitable for analyzing large data sets, especially data from social media. Therefore, in this paper, we present an approach for detecting check-worthiness in texts that uses Model Souping to benefit from ensemble classification while consuming fewer resources and having low inference times.

The paper is structured as follows: Section 2 summarizes the related work and some winning approaches from the last iterations of the challenge. In Section 3, we describe our solution for detecting check-worthiness in multimodal tweets, whereas Section 4 contains a description of

[1] http://www.factcheck.org

[2] http://www.politifact.com

[3] https://www.snopes.com/fact-check/

our approaches to estimate the check-worthiness in written text along with their results on the respective data sets. The last section concludes our work with a brief discussion.

2. Related Work

The initial check-worthiness detection methods were based on extracting meaningful features. Given U.S. presidential election transcripts, ClaimBuster [1] predicts check-worthiness by extracting a set of 6,615 features in total (sentiment, word count, tf-idf weighted bag-of-words, Part-of Speech tags, entity type), and used an SVM classifier for the prediction. Gencheva et al. [8] extended the features used by ClaimBuster by including contextual features such as the sentence's position, the size of a segment belonging to a speaker, topics, or word embeddings. Using all features in combination with a neural network (FNN) outperformed the ClaimBuster version achieving a MAP of 0.427.

In the CheckThat! 2018 competition on check-worthiness detection Hansen et al. [9] showed that an RNN with multiple word representations (word embeddings, POS tagging, and syntactic dependencies) could obtain state-of-the-art results for check-worthiness prediction. The authors later [10] extended their work by applying weak supervision using a collection of unlabeled political speeches and showed significant improvements.

The objective of the CheckThat! challenge in 2021 was to determine which tweets within a set of COVID-19 related tweets are worth checking. The authors of the best performing model [2] fine-tuned several pretrained transformer models. BERTweet achieved the best results (MAP 0.849 on the development set), a model that was trained on 850 million English tweets and 23 million COVID-19 related English tweets using RoBERTa.

Savchev [3] experimented in the CheckThat! 2022 competition with three different pretrained transformer models: BERT, DistilBERT and RoBERTa. Back translation (English tweets were translated to French and back to English) was applied to increase the training set. The best results (F_1 0.90, Accuracy 0.85), and thus the first place in the competitions, were achieved by combining data augmentation and the RoBERTa model.

Gao et al. [4] participated in the AAAI 2022 Multimodal Fact Verification Factify Challenge by implementing two baseline solutions including an ensemble model and an end-to-end multimodal entailment model. The ensemble model outperformed the end-to-end model. They combined two uni-modal models and a multimodal attention network using a 3-way textual entailment classifier, visual similarity with a pre-trained CNN model, and heuristics learned from the dataset. They additionally explored the multimodal fusion technique to model the interaction between different modalities in claim-document pairs and combine information from them. Their best performing model was ranked first by obtaining a weighted average F_1 score of 0.77 on both the validation and test set.

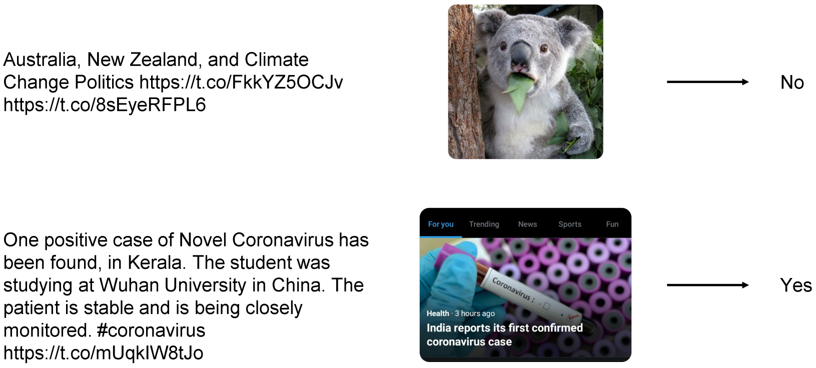
3. Task 1A: Mixing Unimodal Classifiers to Estimate the Check-Worthiness of Multimodal Tweets

In the following, the data set, the examined approaches and their results for detecting check-worthiness in multimodal tweets will be described.

3.1. Data Set Description

The CheckThat! Lab Subtask 1A covered the Arabic and English language; we only participated in the subtask dealing with English data. The data set consisted of social media posts collected by Twitter through its API. Each entry in the dataset contained the text of the tweets, an image, and a text determined by an optical character recognition on the associated image. Examples from the data set are shown in Figure 1.

Figure 1: Instances of check-worthy (Yes) and non-check-worthy (No) tweets for Task 1A



Australia, New Zealand, and Climate Change Politics https://t.co/FkkYZ5OCJv https://t.co/8sEyeRFPL6

One positive case of Novel Coronavirus has been found, in Kerala. The student was studying at Wuhan University in China. The patient is stable and is being closely monitored. #coronavirus https://t.co/mUqklW8tJo

The aim of Task 1A was to predict whether a given multimodal Tweet requires the need of undergoing a manual review by a human expert. Along with the contest, a data set was provided that was divided into four splits: a *train* split, a *dev* split, a *dev-test* split, and a *test* split. While labels for the train set, dev set, and dev-test set were provided upon release, the gold labels for the *test* split were not provided until after the competition was completed. In addition to the labeled data set, a set of unlabeled data was provided. The label distributions of each individual data set split are displayed in Table 1. As can be seen, the data set suffers from class imbalance. Within each split, there were almost twice as many tweets not worthy of verification as tweets worthy of verification.

Table 1
Class distribution of the CheckThat! Lab 2023 task 1B English data set

	Total	Yes	No
Train	2,356 / 100.00 %	820 / 34.80%	1,536 / 65.20%
Dev	271 / 100.00 %	87 / 32.10%	184 / 67.90%
Dev Test	548 / 100.00 %	174 / 31.75%	374 / 68.25%
Test	736 / 100.00 %	277 / 37.64%	459 / 62.36%
Sum	3,911 / 100.00 %	1,358 / 34.72%	2,553 / 65.28%
Unlabeled	110,173 / 100.00 %	?	?

3.2. Methods and Results

Detecting check-worthiness in multimodal tweets presents its own challenges. For one, the length of the texts are limited by a character count restriction. Moreover, both the text and the accompanying image contribute equally to the level of check-worthiness. During the Corona pandemic, text messages were embedded into images, as well as diagrams and charts misinterpreted deliberately. Here, an analysis of both, textual and imagery data is required to assess their check-worthiness fully. In this paper, we present a classification scheme that takes advantage of two classifiers that provide an initial prediction for each modality and then merge their predictions to make a final decision. By including a step that processes the tweets before training and inference, and by using fine-tuning, the classifiers are adapted to the particular writing style typically found in tweets.

3.2.1. Pre-Processing

Unlike text data in documents, books, and web pages, tweets often contain hashtags, emojis, and URLs. The package *pysentimiento* [11] provides methods for resolving emojis and converting hashtags and URLs to generic tokens. As URLs usually do not contribute much to the check-worthiness of a tweet, their analysis can be omitted. By resolving emojis into their descriptive meanings, they can be more easily processed by classifiers previously trained on generic text. An example of the pre-processing can be viewed in Table 2.

Table 2
Example of a pre-processed (PP) tweet.

	Instance
	@MMDA @gmanews Smoke belching Bus..Dapat eto tinatanggal sa road. This contributes increase of Smog!Global Warming! :cold_sweat: https://t.co/AX39EMqC5W
PP	@USER @USER Smoke belching Bus..Dapat eto tinatanggal sa road. This contributes increase of Smog!Global Warming! emoji anxious face with sweat emoji

3.2.2. Classifying the Textual Data

To analyze the textual data, a BERT-based[12] model was fine-tuned on the pre-processed tweets. Throughout the training process, an optimizer based on the Adam algorithm [13] was employed to take advantage of its adaptive learning rate mechanism. Initially, a learning rate of 0.0004 was selected. The model underwent fine-tuning over five epochs, utilizing a batch size of 24. To ensure optimal performance on the competition data set's development split, only the model checkpoint with the highest performance was retained.

The performance of a BERT model trained with and without pre-processing is displayed in Table 3. The classifier trained with pre-processed tweets has higher F_1 scores than the one trained without them. In the specific case of classifying the test set, the F_1 score increased from 0.5377 to 0.7172. Thus, it is advisable to take advantage of pre-processing.

3.2.3. Classifying the Visual Data

For the visual data of the data set, a separate classifier was trained. Two types of classifiers were tried for the challenge, which differed in the type of input data they process: *raw image data* and textual data extracted from an *optical character recognition.*

Using Vision Transformer To classify raw image data, a ViT-based Vision Transformer model was fine-tuned [14]. In particular, the *google/vit-base-patch16-224-in21k* from the hug-

Table 3
Scores achieved by each model on each data set. PP refers to models that took advantage of the pre-processed data. Regarding the metrics, A refers to the accuracy score, P to the precision score, R to the recall score and F_1 to the F_1 score. The character d denotes the dev data split, dt the dev-test data split and t the test split of the data set.

	BERT	BERT + PP	Vision Transformer	OCR	BERT + PP + OCR
A_d	**0.8155**	0.7565	0.6790	0.7048	0.7970
P_d	**0.7937**	0.5827	0.0000	0.5946	0.6379
R_d	0.5747	**0.8506**	0.0000	0.2529	**0.8506**
$F1_d$	0.6667	0.6916	0.0000	0.3548	**0.7291**
A_{dt}	**0.8321**	0.7865	0.6825	0.7190	0.8248
P_{dt}	**0.8361**	0.6245	0.0000	0.6563	0.6893
R_{dt}	0.5862	**0.8218**	0.0000	0.2414	0.8161
$F1_{dt}$	0.6892	0.7097	0.0000	0.3529	**0.7474**
A_t	0.7500	0.7772	0.6236	0.6685	**0.8057**
P_t	**0.8843**	0.6865	0.0000	0.6701	0.7659
R_t	0.3863	**0.7509**	0.0000	0.2347	0.6968
$F1_t$	0.5377	0.7172	0.0000	0.3476	**0.7297**

gingface repository was fine-tuned on the provided image train data using a batch size of 16 within 4 epochs. Similar to the fine-tuned BERT models, Adam was used as the optimizer with a learning rate of 0.0002 and model checkpoints were utilized.

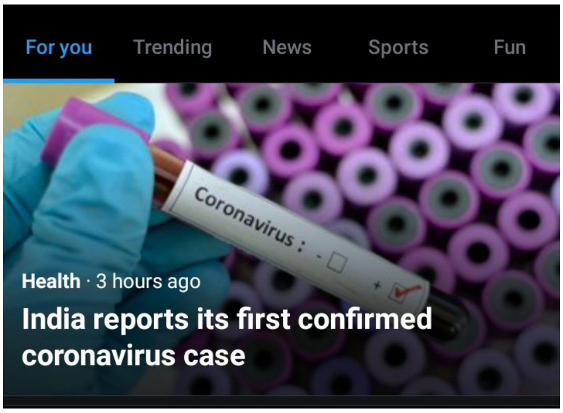
As shown in Table 3, the Vision Transformer was unable to learn meaningful patterns as indicated by the F_1-scores of 0.0000. In particular, the model learned that predicting the majority class (”No”) maximizes the validation loss. There could be several reasons for this. Here, the classifier may have tended to classify the majority class due to the class imbalance within the data set, and the images found in each class may not be sufficiently different for providing good class separation. For further investigation a CNN classifier based on the EfficientNet architecture [15] was trained and evaluated. However, the results did not differ significantly from those of the Vision Transformer. Therefore, the visual model was not included in the final classifier.

Using Optical Character Recognition Since the data-driven imaging models did not perform well in this task, another method for evaluating the information found in the shared images was investigated.

Many times images contain text that can provide additional information for detecting check-worthy content. To evaluate these, the *easyOCR* package was used. It is based on the work of Shi et al. [16] and supports the extraction of text in different languages. The extracted characters were combined into a single string (Figure 2), which then served as input to a fine-tuned BERT model. The BERT model was fine-tuned similar to the classifier predicting the check-worthiness of texts, except a batch size of 8 was utilized.

Compared to the model that predicts text data, the performance of the classifier that estimates check-worthiness based on text within images provides less good results. While the

Figure 2: Text string extracted by easyOCR: *”Trending For you News Sports Fun Coronavirus Health 3 hours ago India reports its first confirmed coronavirus case”*



India reports its first confirmed coronavirus case

accuracy is 69% on average, the F_1 scores obtained for each split are much lower. Since the data suffers from class imbalance, the F_1 score is preferred over the accuracy score. One reason for the lower scores is that not all images contain text and, on the other hand, some images in the data set were not written in English. Therefore, a multilingual model like XLM[17] could provide better performance.

Combining BERT with an OCR-Analysis For the final solution, the classifier that predicts check-worthiness based on text and the classifier that uses optical character recognition were combined. Models that perform better than others should have a greater impact on the final prediction than models that perform less well. Here, we first estimated the validation losses on the dev set for each classifier. Then, the loss values of the opposing models were used to weight the logits predicted by each classification model. By this, the text-based model that was able to produce better results on the dev set, had a greater impact on the final decision than the OCR-based visual model.

Combining the two classifiers resulted in a slight improvement in overall performance. The F_1 values (see Table 3) were improved across the classification of all data sets. With a F_1 score of 0.7297 it placed first in the competition.

4. Task 1B: Tackling Classification Uncertainty Using Model Souping on the Example of Check-Worthiness Classification

The following describes our solution to detecting check-worthiness in textual data using efficient ensemble learning.

4.1. Data Set Description

The ChackThat! Lab data sets for Subtask 1B cover the languages Arabic, English, and Spanish. While we only participated in the English language variant of the task, the described approach can also be adapted for other languages.

For the English task the data set consisted of political debates collected from the US presidential general election debates. Examples from the data set are shown in Table 4.

Table 4
Instances of check-worthy (Yes) and non-check-worthy (No) sentences for Task 1B

	Instance	Class
1.	”And that means 98 percent of American families, 97 percent of small businesses, they will not see a tax increase.”	Yes
2.	I said we’d get tougher with child support and child support enforcement’s up 50 percent.	Yes
3.	But I’m not going to do that.	No
4.	But the important thing is what are we going to do now?	No

The aim of Task 1B was to predict whether a text snippet from a political debate has to be assessed manually by an expert by estimating its check-worthiness. The data set was annotated by human labelers. The label distributions and data set split were provided by the organizers and are shown in Table 5. The ”train” corpus consists of 16,876 entries. Each entry is labeled either ”Yes” or ”No” on whether it is worth fact-checking (YES) or not (No). The organizers have also provided a development set ”dev” (5,625 entries), a development test set ”dev test” (1,032 entries), and a test set with 318 statements. As it can be seen, the data set is highly imbalanced with about a quarter of the sentences being check-worthy. This is also due to the fact that attention-worthy sentences occur less frequently in the text than non-check-worthy sentences.

Table 5
Class distribution of the CheckThat! Lab 2023 Task 1B English data set

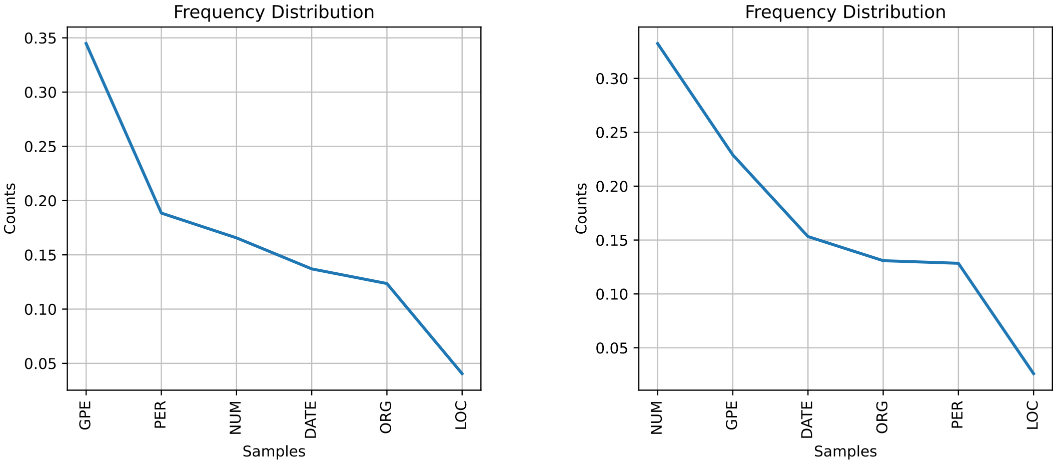
	Total	Yes	No
Train	16,876 / 100.00%	4,058 / 24.05%	12,818 / 75.95%
Dev	5,625 / 100.00%	1,355 / 24.09%	4,270 / 75.91%
Dev Test	1,032 / 100.00%	238 / 23.06%	794 / 76.94%
Test	318 / 100.00%	108 / 33.96%	210 / 66.04%
Sum	23,851 / 100.00%	5,759 / 24.15%	18,092 / 75.85%

4.2. Methodology and Results

Text data from social media and messenger applications such as Twitter and Telegram, news and blogging websites, and transcribed political debates may contain incorrect information that needs to be subjected to manual review by an expert. Hereby, texts of interest are those that contain asserted facts that can be proven or disproven. To identify if a text is check-worthy, three approaches have been tested as part of the CheckThat! 2023 competition: an *estimation using named entities*, a method *combining the named entity recognition with BERT* and the final solution consisting of an *ensemble classifier based on Model Souping*. In the following, the three approaches will be described and their performance discussed.

Estimation Using Named Entity Analysis Facts can often be expressed using named entities, such as names (person / corporation / location / event / objects) or numbers (cardinals /

Figure 3: Normalized distribution of named entity types



(a) In *non-check-worthy* texts

(b) In *check-worthy* texts

ordinals / quantities) and dates. A thorough examination of the train partitioning of the data set revealed that samples classified as check-worthy had a higher use of named entities than those classified as not worthy of reviewing. Using Flair [18], a named entity recognition model pre-trained on the OntoNotes data set[19], the named entities within each of the provided text snippets were extracted and categorized. Hereby, check-worthy texts contained on average 1.679 named entities, whereby non-attention-worthy texts featured only 0.662 named entities on average. Further analysis showed that in addition to the number of named entities featured, the types of entities also varied between the two classes. Figure 3 showcases the distribution of named entity types pre-grouped by similarity. The parent type *NUM* consists of ordinal numbers, cardinal numbers, quantities, percentages, and money, while *DATE* consists of time and dates. *GPE* consists of nationalities, countries, and states, and *LOC* consists of places and events. *PER* and *ORG* remained self-contained. The distribution shows that texts worth examining often contain numbers and counts, while nationalities, countries, and states are found less frequently.

The resulting information was then used to train a classifier, namely a logistic regression model, using the number of a given parent type as input. As indicated in Table 6, the model was able to achieve medium to high accuracies, especially when classifying dev and dev-test split of the data set. In comparison, however, the F_1-values are very low, making the model unsuitable for real-world applications. One reason for this lies in the imbalanced class distribution within the data set. Another one is that analyzing the occurrence of named entities alone does not provide enough information for a precise estimation. A text mentioning numerous named entities but which is written in a subjective tone and expresses an opinion is not worthy of review. Thus, to mitigate this problem, contextual information must be analyzed as well.

Table 6
Scores achieved by each model on each data set. A refers to the accuracy score, P to the precision score, R to the recall score, and F_1 to the F_1 score. The character d denotes the dev data split, dt the dev-test data split, and t the test split of the data set.

	Logistic Regression + NER	BERT + NER	BERT A	BERT B	BERT C	Model Souping + BERT
A_d	0.7909	**0.8796**	0.8728	0.8764	0.8565	0.8670
P_d	0.6751	**0.7834**	0.7524	0.7248	0.6608	0.6849
R_d	0.2546	0.6915	0.7041	0.7852	**0.8310**	0.8295
$F1_d$	0.3697	0.7346	0.7274	**0.7538**	0.7362	0.7503
A_{dt}	0.8430	0.9554	0.9690	**0.9729**	0.9680	0.9709
P_{dt}	0.8333	0.9444	**0.9558**	0.9303	0.8958	0.9094
R_{dt}	0.3992	0.8571	0.9076	0.9538	**0.9748**	0.9706
$F1_{dt}$	0.5398	0.8987	0.9310	**0.9419**	0.9336	0.9390
A_t	0.6981	0.8711	0.8710	**0.9308**	**0.9308**	0.9214
P_t	0.7727	**0.9855**	0.9351	0.9674	0.9216	0.9278
R_t	0.1574	0.6296	0.6667	0.8241	**0.8704**	0.8333
$F1_t$	0.2615	0.7684	0.7784	0.8900	**0.8952**	0.8780

Combining the Analysis of Named Entities with Language Models To include additional information about the context, the second attempt combined the named entity recognition with a language model. Here, BERT[12] was fine-tuned on data, in which the named entities found in the previous step were exchanged with special tokens reflecting their respective named entity type (see Table 7). For this, the tokenizer was modified to contain the six additional tokens *<NUM>*, *<DATE>*, *<LOC>*, *<GPE>*, *<PER>*, and *<ORG>*.

Table 7
Examples of a named entity extraction in check-worthy (Yes), and non-check-worthy (No) sentences

	Instance	Class
1.	”And that means <98 percent, NUM> of <American, GPE> families, <97 percent, NUM> of small businesses, they will not see a tax increase.”	Yes
2.	I said we’d get tougher with child support and child support enforcement’s up <50 percent, NUM>.	Yes
3.	But I’m not going to do that.	No
4.	But the important thing is what are we going to do now?	No

During training, an optimizer based on Adam[13] was utilized to leverage from the adaptive learning rate mechanism. A learning rate of 0.0004 was chosen as the initial learning rate. The model was fine-tuned in 5 epochs with a batch size of 24. Model checkpoints were used to keep only the model checkpoint, that performed best on the dev split of the competition data set.

Table 3 shows, that by combining a named entity recognition with a language model such as BERT, the performance can be further increased.

4.3. Final Solution Using Ensemble Learning Based on Model Souping

To compare the hybrid method with a fully data-driven approach, fine-tuning was performed using solely the raw text data. Again, a BERT model was chosen and fine-tuned using the configuration described in the previous section.

Training the model several times with different seeds showed large differences in performance (see BERT A, BERT B, and BERT C in Table 6). This is because the initial weights of the model are initialized differently depending on the set seed. The same applies to the way the training split is shuffled after each epoch. As a result, individual models converge differently and can find different local minima, resulting in a sometimes good or less good performance. Unable to determine which seed maximizes performance on the validation, and test sets, it is common to take advantage of ensemble learning.

There exist several approaches to perform ensemble classification, such as *bagging*, *boosting*, and *stacking*[20]. In each of the methods, individually trained classifiers called weak classifiers are combined to improve the classification uncertainty. The main disadvantage of ensemble learners, however, lies within their computational efficiency during inference. In particular, stacking-based ensemble classifiers, which consist of a combination of N models providing an initial prediction and a meta classifier taking these to form a final decision output, require the inference of $N+1$ models. As such, ensemble classification may not be applicable in real-world applications, in which large amounts of data need to be assessed in a timely manner using as less computational resources as possible.

To compensate for these problems, *Model Souping* as proposed by Wortsman et al. can be applied [21]. Model Souping removes the requirement of having multiple weak classifiers and a meta-classifier by providing a single master-model that is used during inference. Master-models can be built by taking the trained weights of each individual classifier and combining them by averaging, weighted averaging, or using a feedback loop. Initial tests with image and text classification tasks showed improved performance while maintaining resource efficiency. It should be noted, however, that Model Souping can only be applied with models sharing the same architecture.

In this paper, we took advantage of Model Soups that adaptively adjust the influence of each individual model in the master model based on the performance on the dev split of the data set. Here, the fully data-driven models were used in favor of the hybrid models BERT with a named entity recognition due to their performance on the dev, and dev-test split. By evaluating each of the three trained models on the dev set, their test loss values were retrieved. Based on them, their influence-score I was calculated using the following formula:

$$I = L_{test}/L_{total} \tag{1}$$

Low-performing models should have a lower impact within the master model, whereas better-performing ones, should have a higher influence on the weights of the master model. The influence value I was then used to weight the trained weights of each model.

While the ensemble classifier was unable to outperform the best individual classifier (BERT C; $F_1 = 0.8952$) on the test data set, it helped with balancing out results from models (BERT A; $F_1 = 0.7784$) suffering from low performance. It should be noted, however, that if all weak classifiers perform equally well on a particular data set, the performance gain will be negligible.

The approach based on Model Souping was used to classify the private test set of this year's CheckThat! competition. It was able to place second best. Although it performed best among the three methods described, its capabilities in terms of explainability and transparency are limited due to the fact that it is a fully data-driven approach.

5. Conclusion

The detection of check-worthy texts can be seen as a first step towards identifying false information spread on the Internet. When used as a pre-filter, it can dramatically reduce the amount of data that needs to be manually reviewed by human experts. In this paper, we have described our approach to check-worthiness detection in multimodal and unimodal content.

Multimodal data in social media, such as Twitter, pose new challenges for check-worthiness detection. We presented a new method for detecting review-worthy tweets that contain an image in addition to the descriptive text of the tweet (Task 1A). It combines two classifiers trained separately for each modality. The experiments showed that when analyzing visual data, an OCR analysis outperformed a classifier trained on raw image data. Combining the BERT model trained on the tweet text with the BERT model trained on the extracted strings from an optical character recognition slightly improved the performance. The combined approach performed best in the competition with a F_1 score of 0.7297.

For Task 1B, we presented an ensemble classification scheme based on Model Souping. Experiments on the validation split and the private test set revealed that the proposed approach can be used to tackle the issue of classification uncertainty while reducing the computational overhead often associated with ensemble learning. The model was able to place second best in the competition with a F_1 score of 0.878. Future work may consider applying weight adjustments using a feedback loop to better compensate for the misclassification of edge cases as well as introducing other means to achieve explainability and transparency.

Acknowledgements

This work was supported by the German Federal Ministry of Education and Research (BMBF) and the Hessian Ministry of Higher Education, Research, Science and the Arts within their joint support of “ATHENE – CRISIS” and ”Lernlabor Cybersicherheit” (LLCS).

References

[1] N. Hassan, F. Arslan, C. Li, M. Tremayne, Toward automated fact-checking: Detecting check-worthy factual claims by claimbuster, Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2017).

[2] J. R. Martinez-Rico, J. Martínez-Romo, L. Araujo, Nlp&ir@uned at checkthat! 2021: Check-worthiness estimation and fake news detection using transformer models, in: G. Faggioli, N. Ferro, A. Joly, M. Maistro, F. Piroi (Eds.), Proceedings of the Working Notes of CLEF 2021 - Conference and Labs of the Evaluation Forum, Bucharest, Romania, September 21st - to - 24th, 2021, volume 2936 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2021, pp. 545–557. URL: https://ceur-ws.org/Vol-2936/paper-44.pdf.

[3] A. Savchev, AI rational at checkthat!-2022: Using transformer models for tweet classification, in: G. Faggioli, N. Ferro, A. Hanbury, M. Potthast (Eds.), Proceedings of the Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum, Bologna, Italy, September 5th - to - 8th, 2022, volume 3180 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2022, pp. 656–659. URL: https://ceur-ws.org/Vol-3180/paper-52.pdf.

[4] J. Gao, H. Hoffmann, S. Oikonomou, D. Kiskovski, A. Bandhakavi, Logically at factify 2022: Multimodal fact verfication, in: A. Das, T. Chakraborty, A. Ekbal, A. P. Sheth (Eds.), Proceedings of the Workshop on Multi-Modal Fake News and Hate-Speech Detection (DE-FACTIFY 2022) co-located with the Thirty-Sixth AAAI Conference on Artificial Intelligence (AAAI 2022), Virtual Event, Vancouver, Canada, February 27, 2022, volume 3199 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2022. URL: https://ceur-ws.org/Vol-3199/paper6.pdf.

[5] A. Barrón-Cedeño, F. Alam, A. Galassi, G. Da San Martino, P. Nakov, , T. Elsayed, D. Azizov, T. Caselli, G. Cheema, F. Haouari, M. Hasanain, M. Kutlu, C. Li, F. Ruggeri, J. M. Struß, W. Zaghouani, Overview of the CLEF–2023 CheckThat! Lab checkworthiness, subjectivity, political bias, factuality, and authority of news articles and their source, in: A. Arampatzis, E. Kanoulas, T. Tsikrika, S. Vrochidis, A. Giachanou, D. Li, M. Aliannejadi, M. Vlachos, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Fourteenth International Conference of the CLEF Association (CLEF 2023), 2023.

[6] A. Barrón-Cedeño, F. Alam, T. Caselli, G. Da San Martino, T. Elsayed, A. Galassi, F. Haouari, F. Ruggeri, J. M. Struß, R. N. Nandi, G. S. Cheema, D. Azizov, P. Nakov, The CLEF-2023 CheckThat! Lab: Checkworthiness, subjectivity, political bias, factuality, and authority, in: J. Kamps, L. Goeuriot, F. Crestani, M. Maistro, H. Joho, B. Davis, C. Gurrin, U. Kruschwitz, A. Caputo (Eds.), Advances in Information Retrieval, Springer Nature Switzerland, Cham, 2023, pp. 506–517.

[7] F. Alam, A. Barrón-Cedeño, G. S. Cheema, S. Hakimov, M. Hasanain, C. Li, R. Míguez, H. Mubarak, G. K. Shahi, W. Zaghouani, P. Nakov, Overview of the CLEF-2023 CheckThat! lab task 1 on check-worthiness in multimodal and multigenre content, in: M. Aliannejadi, G. Faggioli, N. Ferro, Vlachos, Michalis (Eds.), Working Notes of CLEF 2023–Conference and Labs of the Evaluation Forum, CLEF 2023, Thessaloniki, Greece, 2023.

[8] P. Gencheva, P. Nakov, L. Màrquez, A. Barrón-Cedeño, I. Koychev, 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, INCOMA Ltd., Varna, Bulgaria, 2017, pp. 267–276. URL: https://doi.org/10.26615/978-954-452-049-6_037. doi:10.26615/978-954-452-049-6_037.

[9] C. Hansen, C. Hansen, J. Simonsen, C. Lioma, The copenhagen team participation in the check-worthiness task of the competition of automatic identification and verification of claims in political debates of the clef-2018 checkthat! lab, in: L. Cappellato , N. Ferro , J. Nie, L. Soulier (Eds.), CLEF 2018 Working Notes, CEUR Workshop Proceedings, CEUR-WS.org, 2018. 19th Working Notes of CLEF Conference and Labs of the Evaluation Forum, CLEF 2018 ; Conference date: 10-09-2018 Through 14-09-2018.

[10] C. Hansen, C. Hansen, J. G. Simonsen, C. Lioma, Neural weakly supervised fact check-worthiness detection with contrastive sampling-based ranking loss, in: L. Cappellato,

N. Ferro, D. E. Losada, H. Müller (Eds.), Working Notes of CLEF 2019 - Conference and Labs of the Evaluation Forum, Lugano, Switzerland, September 9-12, 2019, volume 2380 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2019. URL: https://ceur-ws.org/Vol-2380/paper_56.pdf.

[11] J. M. Pérez, J. C. Giudici, F. Luque, pysentimiento: A python toolkit for sentiment analysis and socialnlp tasks, 2021. arXiv:2106.09462.

[12] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, 2019. arXiv:1810.04805.

[13] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2017. arXiv:1412.6980.

[14] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, An image is worth 16x16 words: Transformers for image recognition at scale, 2021. arXiv:2010.11929.

[15] M. Tan, Q. V. Le, Efficientnet: Rethinking model scaling for convolutional neural networks, 2020. arXiv:1905.11946.

[16] B. Shi, X. Bai, C. Yao, An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition, 2015. arXiv:1507.05717.

[17] G. Lample, A. Conneau, Cross-lingual language model pretraining, 2019. arXiv:1901.07291.

[18] A. Akbik, T. Bergmann, D. Blythe, K. Rasul, S. Schweter, R. Vollgraf, 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), 2019, pp. 54–59.

[19] Weischedel, Ralph, Palmer, Martha, Marcus, Mitchell, Hovy, Eduard, Pradhan, Sameer, Ramshaw, Lance, Xue, Nianwen, Taylor, Ann, Kaufman, Jeff, Franchini, Michelle, El-Bachouti, Mohammed, Belvin, Robert, Houston, Ann, Ontonotes release 5.0, 2013. URL: https://catalog.ldc.upenn.edu/LDC2013T19. doi:10.35111/XMHB-2B84.

[20] E. Bauer, R. Kohavi, An empirical comparison of voting classification algorithms : Bagging, boosting, and variants, Machine Learning 36 (1996) 1–38.

[21] M. Wortsman, G. Ilharco, S. Y. Gadre, R. Roelofs, R. Gontijo-Lopes, A. S. Morcos, H. Namkoong, A. Farhadi, Y. Carmon, S. Kornblith, L. Schmidt, Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time, 2022. arXiv:2203.05482.