How Entangled is Factuality and Deception in German?

Aswathy Velutharambath[1,2,3], **Amelie Wührl**[1,3] and **Roman Klinger**[3]
[1]Institut für Maschinelle Sprachverarbeitung, University of Stuttgart, Germany
[2]Psychological AI (100 Worte Sprachanalyse GmbH), Heilbronn, Germany
[3]Fundamentals of Natural Language Processing, University of Bamberg, Germany
aswathy.velutharambath@100worte.de, amelie.wuehrl@ims.uni-stuttgart.de
roman.klinger@uni-bamberg.de

Abstract

The statement "*The earth is flat*" is factually inaccurate, but if someone truly believes and argues in its favor, it is not deceptive. Research on deception detection and fact checking often conflates factual accuracy with the truthfulness of statements. This assumption makes it difficult to (a) study subtle distinctions and interactions between the two and (b) gauge their effects on downstream tasks. The belief-based deception framework disentangles these properties by defining texts as deceptive when there is a mismatch between what people say and what they truly believe. In this study, we assess if presumed patterns of deception generalize to German language texts. We test the effectiveness of computational models in detecting deception using an established corpus of belief-based argumentation. Finally, we gauge the impact of deception on the downstream task of fact checking and explore if this property confounds verification models. Surprisingly, our analysis finds no correlation with established cues of deception. Previous work claimed that computational models can outperform humans in deception detection accuracy, however, our experiments show that both traditional and state-of-the-art models struggle with the task, performing no better than random guessing. For fact checking, we find that natural language inference-based verification performs worse on non-factual and deceptive content, while prompting large language models for the same task is less sensitive to these properties.

1 Introduction

In NLP, a vast body of the existing research on deception has focused on detecting the veracity of a statement in a given context or domain (Ott et al., 2011; Salvetti et al., 2016; Pérez-Rosas and Mihalcea, 2014; Capuozzo et al., 2020), often ignoring the difference between factual accuracy and truthfulness, implicitly equating the two. For example, consider the statement "*The Colosseum in Rome is an iconic ancient amphitheater. My family owns a secret underground passage that leads directly to it*". The first part of the statement is factually accurate[1] and verifiable, while the second part is likely fabricated, illustrating the importance of disentangling truthfulness from factual accuracy to accurately assess deceptive intent.

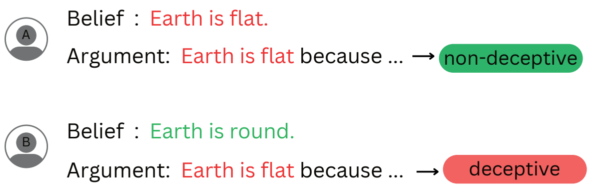
A
Belief : Earth is flat.
Argument: Earth is flat because ... → non-deceptive
B
Belief : Earth is round.
Argument: Earth is flat because ... → deceptive

Figure 1: Belief-based deception framework assigns deception label to arguments based on the alignment of the person's belief and argumentation, irrespective of the factual accuracy (green = factually accurate, red = not fact. accurate) of the claim.

Similarly, in fact verification, mis- and disinformation – differing in terms of the underlying deceptive intent – are typically conflated (Boland et al., 2022). Both in deception detection and fact checking, this conventional focus leaves a gap in understanding how deceptive intent operates independently of factual accuracy, which is crucial for developing more sophisticated models that can discern subtle cues of deceptions even when embedded within truthful contexts.

In Velutharambath et al. (2024), we disentangle factuality and deception by introducing a framework in which deception is defined as arguing against one's own beliefs (see Figure 1). This approach emphasizes the deceptive intent rather than the factual correctness of the statement, thereby providing a better understanding of how deception can occur even when parts of a statement are factually accurate.

[1] *Factual accuracy* here refers to the correctness of a statement based on objective evidence that can be verified. It indicates that the statement aligns with the actual state of affairs and can be proven true or false using reliable sources.

Findings of the Association for Computational Linguistics: EMNLP 2024, pages 9538–9554
November 12-16, 2024 ©2024 Association for Computational Linguistics

The belief-based deception corpus (DeFaBel) developed using this framework provides the opportunity to explore the relationship between personal beliefs, factual accuracy, and deceptive intent. The dataset comprises of both deceptive and non-deceptive arguments supporting the same topic statement. Moreover, the DEFABEL corpus is the only publicly available dataset on deception in German, facilitating an investigation into the generalizability of linguistic cues across languages. With this paper, we perform the previously lacking experiments to assess the effectiveness of automatic deception models within this corpus.

We make the following contributions: (1) We investigate how linguistic cues of deception manifest in German-language texts. Surprisingly, we find no statistically significant correlations between deception labels and established cues for deceptive content. (2) We assess the effectiveness of transformer-based and large language models in identifying deceptive arguments. Our observations indicate that these models may not be accurately capturing deception cues from language, as their performance is close to random guessing. (3) Finally, we explore how evidence-based fact checking is confounded by deception and factuality. Our findings suggest that non-factual and deceptive content poses particularly challenging for fact checking models[2].

2 Background

2.1 Deception

Deception in communication involves intentionally causing another to adopt a belief known by the deceiver to be false (Zuckerman et al., 1981; Mahon, 2007). This encompasses lies, exaggerations, omissions, and distortions as various manifestations of deceptive acts (Turner et al., 1975; Metts, 1989). Despite variations in definition across disciplines, the consensus underscores the deliberate nature of deception (Mahon, 2007; Gupta et al., 2013).

Corpora. Automatic deception detection from text heavily relies on labeled corpora. Unlike other NLP tasks, the gold labels cannot be assigned post-data collection, as the veracity of the statement hinges on the intention of the author. Deceptive instances have been solicited via crowd-sourcing, as fake reviews (Ott et al., 2011, 2013; Salvetti et al., 2016) or false opinions on controversial topics (Pérez-Rosas and Mihalcea, 2014; Capuozzo et al., 2020; Lloyd et al., 2019) and by monitoring fake review generation tasks (Yao et al., 2017) or users exhibiting suspicious activity (Fornaciari et al., 2020). Deceptive instances have also been extracted from dialogue in strategic deception games such as Mafiascum[3], Box of Lies, Diplomacy and, To Tell the Truth (de Ruiter and Kachergis, 2018; Soldner et al., 2019; Peskov et al., 2020; Skalicky et al., 2020; Hazra and Majumder, 2024) based on specific game rules. None of the mentioned datasets explicitly address the distinction between factual inaccuracy and deceptive intent. In contrast, Velutharambath et al. (2024) tackles this by creating a corpus of argumentative texts, where the deception label is assigned based on the author's true beliefs, irrespective of factual accuracy, thereby disentangling the two.

Most deception corpora focus on English, with a comprehensive overview provided by Velutharambath and Klinger (2023). Attempts at deception detection in other languages have been made, albeit to a lesser extent, including Polish (Sarzynska-Wawer et al., 2023), Bulgarian (Temnikova et al., 2023), Italian (Capuozzo et al., 2020), Russian (Pisarevskaya et al., 2017), Dutch (Verhoeven and Daelemans, 2014), and Spanish (Almela et al., 2012) texts. In this study we make use of the DEFABEL corpus introduced by Velutharambath et al. (2024), the only publicly available corpus to study deception in German.

Linguistic Cues. Previous research has explored linguistic cues of deception across various modalities, including written statements, spoken dialogues, and online discourse (Newman et al., 2003; Bond and Lee, 2005; Zhou et al., 2004). These cues have proven valuable in automated deception detection, especially in computer-mediated communication (Zhou et al., 2004). For instance, Newman et al. (2003) found that reduced self-references in deceptive statements suggest liars create distance from their falsehoods, while the use of exclusive words (e.g., *but*, *rather*) introduces ambiguity. Hancock (2009) noted variations in word count, pronoun usage, emotion word frequency, and cognitive complexity across discourse types and mediums (e.g., real-world vs. online, monologue vs. dialogue). Linguistic analyses have been integrated into machine learning models alongside other features like n-grams (Fornaciari and Poesio, 2014; Fornaciari et al., 2020; Ott et al., 2011), part-of-

[2] Code & data: https://www.ims.uni-stuttgart.de/data/defabel

[3] https://www.mafiascum.net/

speech tags (Lloyd et al., 2019; Fornaciari et al., 2020; Pérez-Rosas and Mihalcea, 2015), LIWC psychological categories (Pérez-Rosas and Mihalcea, 2014; Yao et al., 2017), and syntactic production rules (Yao et al., 2017; Pérez-Rosas and Mihalcea, 2015). Velutharambath and Klinger (2023) evaluates the generalizability of different linguistic cues across multiple deception datasets. Extensive surveys of linguistic deception cues include works by Duran et al. (2010), Swol et al. (2012), and Hauch et al. (2015). In the current study, we include the most commonly discussed linguistic cues in our analysis.

Automatic Detection Methods. Several studies have explored automated deception detection in textual data. Some employed feature-based classification methods with linguistic cues, using support vector machines (Ott et al., 2011; Pérez-Rosas and Mihalcea, 2014; Fornaciari and Poesio, 2014), logistic regression (de Ruiter and Kachergis, 2018), decision trees (Pérez-Rosas and Mihalcea, 2015), and random forests (Soldner et al., 2019; Pérez-Rosas and Mihalcea, 2015). Others integrated contextual information with recurrent neural networks (Peskov et al., 2020) and transformer-based models (Capuozzo et al., 2020; Peskov et al., 2020; Fornaciari et al., 2021). Transformers are not uniformly superior; BERT performs comparably to LSTMs (Peskov et al., 2020), however adding extra attention layers can enhance performance (Fornaciari et al., 2021). Cross-corpus and within-corpus experiments with the RoBERTa model reveal limited generalizability across domains (Velutharambath and Klinger, 2023). Recent studies have utilized Large Language Models (LLMs) like FLAN-T5 on English-language datasets, achieving state-of-the-art results, especially with larger models (Loconte et al., 2023). Hazra and Majumder (2024) used zero-shot prompting to extract various cues of deception from text and a discriminator to aggregate the final prediction. In our study, we assess feature-based models, fine-tuned transformer models, and state-of-the-art LLMs for deception detection.

2.2 Factuality

Fact verification is concerned with determining if a statement is factual, i.e., "generally accepted to be true" (Boland et al., 2022). While some work proposed to predict veracity or properties such as untrustworthiness based on claim characteristics, hypothesizing that mis- and disinformation is encoded in the linguistic properties of a claim (Wang, 2017; Rashkin et al., 2017), the verification process is typically evidence-based (Guo et al., 2022; Vladika and Matthes, 2023; Hardalov et al., 2022). The input to a fact checking model is a claim-evidence pair for which we predict a verdict indicating if the evidence entails, i.e., supports, or contradicts the claim. Importantly, fact verification usually does not differentiate intentionally spreading false information (disinformation), e.g., with a deceptive intent, from other misinformation.

Some work has studied the impact of claim and evidence properties on fact checking (Atanasova et al., 2022; Kelk et al., 2022; Hansen et al., 2021) and the effect of synthetic disinformation on fact checking performance (Du et al., 2022). However, the effect of a claim's factual accuracy and any deceptive intent in the evidence documents remains unstudied presumably because of a lack of resources that combines factuality and deception. We leverage the belief-based deception framework (Velutharambath et al., 2024) to understand the role of deceptive evidence and factuality in the verification process.

2.3 Belief-based Deception Framework

Concept. In belief-based deception framework (Velutharambath et al., 2024), deception is defined as arguing against one's own beliefs, emphasizing the deceptive intent rather than the factual correctness of the statement. In Figure 1, the statement "*The earth is flat*" is factually inaccurate, but if someone truly believes and argues in its favor, it is not deceptive. In this framework, content is deceptive if there is a mismatch between what people say and what they truly believe.

Dataset description. In this study, we use the DeFaBel corpus of belief-based deception (Velutharambath et al., 2024). It is a publicly available[4] corpus contains argumentative German texts collected via crowdsourcing. Participants were solicited to write persuasive arguments supporting a given statement irrespective of its factuality or their personal beliefs. When the argumentation is in contradiction with their own belief, the instance is labeled as deceptive. The belief of the annotator is collected after generating the text. Out of the 30 statements used for soliciting arguments, 19 were

[4] The dataset is available under a Creative Commons license CC BY-SA. We use the data in its original form without any modification.

factual (e.g., "*Camels store fat in their hump.*") and 11 were non-factual (e.g., "*Eating watermelon seeds can cause indigestion.*"). The corpus contains 1031 German texts labeled with deceptive intent. The distribution of labels is skewed, with a higher frequency of deceptive instances ($\approx 62\%$) compared to non-deceptive ones. Also, $\approx 60\%$ of the arguments support non-factual statements. We employ this corpus for all experiments in this paper.

3 How Do Linguistic Cues of Deception Manifest in German Texts? (RQ1)

This study evaluates the reliability of linguistic cues widely presumed to encode deception, for developing an automated deception detection system in German. As a groundwork to investigate this, we conduct a correlation analysis to understand the relationship between each of the linguistic cues and the deception label.

3.1 Methods

We consider the most commonly used linguistic cues of deception from prior studies, as follows.

Complexity. Previous studies argue that deceptive statements tend to be simpler compared to truthful ones, attributing it to the increased cognitive load when deceiving someone which hinders creativity and complex sentence production (DePaulo et al., 2003; Hauch et al., 2015). This implies that deceptive responses may exhibit reduced lexical diversity, are shorter or less elaborate relative to truthful responses. To represent the complexity of arguments, we use token count, sentence count, unique token count, type-token ratio (lexical diversity) and average POS-tag counts. As a measure of sophistication of language, we use Flesch–Kincaid (Kincaid et al., 1975) and Gunning–Fog (Robert, 1968) readability scores.

Concreteness. Linguistic concreteness has been deemed to be important in detecting deception (Kleinberg et al., 2019), but its role is inconclusive from previous studies. Some argue that liars speak abstractly due to difficulty in recalling the details, while truth-tellers provide more specific information (Kleinberg et al., 2018). The contrary argues that deceptive narratives may include concrete actions to reduce the cognitive load (Newman et al., 2003). To assign a concreteness score for each argumentative text, we use a German lexical resource created by Köper and Schulte im Walde (2016), which assigns a concreteness value to a lemmatized word. In addition, we include the imageability score from the same resource, as concrete words are often said to have higher imageability compared to abstract words.

Sentiment. Studies like Hauch et al. (2015) and Vrij (2008) have reported that negative emotions are more prevalent in deceptive speech, possibly induced by guilt and fear of getting caught. As the arguments in the DeFaBel corpus support emotionally neutral statements, rather than emotions, we investigate sentiment as a cue to understand if the general sentiment varies in the case of deceptive and non-deceptive statements. We use a sentiment classifier to assign sentiment scores to each argumentative text. Additionally, we include the valence (pleasantness) and arousal (emotional intensity) scores from Köper and Schulte im Walde (2016) as they are closely related to emotions.

LIWC features. As noted by Newman et al. (2003), word usage patterns differ between truth-tellers and liars. Previous studies have extensively used LIWC (Pennebaker et al., 2015), a general lexicon capturing different psychological categories, to study these patterns in deceptive language (Hauch et al., 2015; Pérez-Rosas and Mihalcea, 2014; Yao et al., 2017). Along with other deception specific cues, we use all psychological categories available in LIWC, specifically for German, to account for potential variations in language use between deceptive and non-deceptive arguments.

3.2 Experimental Setup

We use the measure of *point-biserial correlation* (Glass and Hopkins, 1996) to study the correlation between the deception label (a discrete value) and the scores assigned to different features (a continuous value). The point-biserial correlation ρ_{pb} is

$$\rho_{\text{pb}} = \frac{\mu_{\text{decept}} - \mu_{\text{truth}}}{\sigma_n} \sqrt{\frac{n_{\text{decept}} n_{\text{truth}}}{n(n-1)}} \quad (1)$$

where n is the total of instances, μ_{truth} and μ_{decept} are the mean values of the continuous variable for deceptive and truthful instances respectively, σ_n the standard deviation of the continuous variable, and n_{truth} and n_{decept} the frequencies of the promotion and prevention labels, respectively, within the dataset. The point-biserial correlation coefficient, ranging from -1 to $+1$, signifies perfect negative and perfect positive correlations, respectively. A high positive correlation coefficient implies that the score tends to be higher when the instance label is deceptive. Conversely, a high negative correlation coefficient suggests that the score is higher when the label is non-deceptive. The magnitude and sign of the correlation coefficient offer insights into the strength and direction of the relationship between the deception label and the cues.

We use the point-biserial correlation implementation from `scipy`[5] to analyze the relationship between 128 linguistic cues and the deception label. We set the desired overall significance level (α) to 0.05 for *Bonferroni correction* (Bonferroni, 1936). Table 7 in the Appendix shows details of all linguistic cues.

3.3 Results

Surprisingly, from our analysis, we observe that none of the 128 cues show any statistically significant correlation with the deception label, implying the absence of a discernible linear relationship between the cues and deception. Given that *Bonferroni corrections* are typically stringent, we opted to consider the non-adjusted α (.05), resulting in 13 variables with significant but weak correlation and the scores ranging from $-.095$ to .068. While this may suggest there is a lack of strong association between linguistic cues and deception, it is essential to note that the absence of significant correlations does not necessarily imply the absence of meaningful relationships.

The DEFABEL corpus is based on neutral topics that lack strong emotional commitment, with the stakes of lying simulated by offering participants incentives to persuade someone else. This approach may result in deception cues that diverge from those observed in real-life lying scenarios. This underscores the necessity to explore contextual, cultural and individualistic factors alongside linguistic cues.

4 How Efficient are Computational Models in Detecting Deception? (RQ2)

In Section 3, we demonstrate that previously reported deception cues do not significantly correlate with the deception labels in the DEFABEL dataset. However, previous studies have shown that automatic methods have achieved some success in predicting deception. Therefore, we evaluate the performance of traditional feature-based methods, fine-tuned transformer models, and instruction-tuned large language models on the DEFABEL corpus.

Data Split	Deceptive	Non-Deceptive	total
Dev. Data	491	263	754
Holdout Data	152	125	277
Total	643	388	1031

Table 1: Data split of the DEFABEL dataset used for experimentation.

4.1 Experimental Setup

We conduct the deception detection experiment on the DEFABEL corpus, containing 1031 argumentative texts. As shown in Table 1, we split the data into a development set comprising arguments related to 22 statements (754 arguments), and a holdout set consisting of arguments associated with 8 statements (277 arguments). The topics in the two splits do not have any overlap.

For models that we train or fine-tune, we perform 10-fold cross-validation. For instruction-tuned models, we evaluate them on the entire development set. To ensure a fair comparison between models, we aggregate the predictions from the 10 folds and use the micro-average across the complete development set. Finally, we evaluate the best-performing model on the holdout dataset

Models. We use a logistic regression classifier (**Log. Reg**) as implemented in `scikit-learn`[6], with default parameters, to weigh all linguistic cues. For support vector classification (**SVM**), we use `scikit-learn`[7]. As the transformer based model, we utilize the pre-trained German BERT model `deepset/GBERT-large` (**GBERT**). (Appendix A.1 shows modeling details.)

For prompt-based deception detection, we utilize `Mistral 8x7B`[8], a state-of-the-art open LLM (Jiang et al., 2024). The model claims superior performance and multilingual support for English, French, Italian, German, and Spanish.

We evaluate five different prompts in a one-shot setting (refer to Appendix B). In each case, the model is instructed to predict the deceptiveness

[5] https://docs.scipy.org/doc/scipy-0.14.0/reference/generated/scipy.stats.pointbiserialr.html

[6] https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

[7] https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html

[8] https://huggingface.co/mistralai/Mixtral-8x7B-Instruct-v0.1

	Non-Deceptive			Deceptive			
Model	Precision	Recall	F-score	Precision	Recall	F-score	Accuracy
random	**.40**	.52	.45	.64	.52	.58	.52
majority class	.00	.00	.00	.65	**1.00**	**.79**	**.65**
Log. Reg	.38	.29	.33	.66	.75	.70	.59
SVM	.38	.29	.33	.66	.75	.70	.59
GBERT	**1.00**	.00	.01	.65	**1.00**	**.79**	**.65**
Mistral (DECEPT)	.39	.50	.44	**.68**	.58	.63	.55
Mistral (DECEPT_FACT)	.37	.56	.45	**.68**	.50	.57	.52
Mistral (CONV_DECEPT)	.33	.61	.43	.62	.34	.44	.44
Mistral (CONV_DECEPT_FACT)	.35	**.88**	**.50**	.65	.11	.19	.38
Mistral (CONV_DECEPT_FACT_RETHINK)	.37	.47	.42	.67	.57	.62	.54
Evaluated best model on holdout-data							
random	.44	.50	.47	.54	.48	.50	.51
Mistral (DECEPT)	**.46**	.45	.45	.55	.56	.56	.51

Table 2: Results on evaluating models on development and holdout data.

of the provided argument. To determine whether prompting to predict the factuality of the argument improves prediction accuracy, we also instruct the model to predict both the factuality and deceptiveness of the given argument. Furthermore, we conduct experiments with prompts presented as single instructions, as well as in conversational or dialogue formats. Finally, we incorporate chain-of-thought prompting within the dialogue format. We use the following prompts in this study:

1. DECEPT: single instruction prompt, predicts deceptiveness
2. DECEPT_FACT: single instruction prompt, predicts factuality and deceptiveness
3. CONV_DECEPT: predicts deceptiveness in a conversational setting
4. CONV_DECEPT_FACT: predicts factuality and deceptiveness in a conversational setting
5. CONV_DECEPT_FACT_RETHINK: incorporates chain-of-thought in the conversational setting

4.2 Results

Table 2 shows the results of evaluating different models on the deception detection task. From the results on the development data, we see that none of the models perform better than the majority class prediction (.65) in terms of accuracy. However, the dataset is quite imbalanced and focusing on the F$_1$ score for deceptive instances would offer more insightful observations regarding their effectiveness in deception detection. GBERT model achieves an F$_1$ score of 0.79 on the deception label. This result is almost the same as majority class prediction, as the model labeled all instances as deceptive in 9 out of 10 folds. We observe that feature-based models seem to perform comparably well in identifying deceptive instances but struggle with non-deceptive ones. However, this could be also be an artifact of the class imbalance.

The results on prompting instruction-tuned LLMs show that the one-shot single prompt (DECEPT) seems to work slightly better than random in terms of accuracy (Δ 3pp acc.) and relatively good at predicting the deceptive instances (Δ 5pp in F$_1$). Unlike the pre-trained transformer model (GBERT), this is not an artifact of predicting instances as only deceptive. On prompting to predict both factuality and deceptiveness (DECEPT_FACT) the model performance drops. To evaluate whether a conversational setting provides better results, the same prompt was presented as a dialogue between the user and the assistant. The conversational setting in itself did not seem to be better than the single prompt. However, when the model is prompted with chain-of-thought setting, the performance of the model seems to improve in comparison to the single conversational prompt CONV_DECEPT(Δ 3pp acc.). This suggests that optimizing the prompt for a deception detection task could potentially result in better results. Out of all the models, Mistral with the simple prompt (DECEPT) seems to achieve the most reliable results on the deception label. To verify if the model follows instructions to disregard factuality, we modify the prompt to include reasoning for deception label predictions. However, we observe inconsistent adherence with this directive by the model (See Appendix B). Nevertheless, evaluating this "best" model on holdout data gives results similar to random prediction once again.

Our study presents contrasting results compared to previous research that reported over 80% accuracy using simple n-gram-based classifiers (Ott et al., 2011; Ott, 2014). We hypothesize that these models might leverage domain-specific traits rather than capturing genuine linguistic cues in deceptive text. Further, more recent studies employing LSTM and pre-trained language models (Fornaciari et al., 2021; Velutharambath and Klinger, 2023) have shown promising results in deception detection, albeit often limited to specific domains, challenging the broader applicability of these models. In contrast, Pérez-Rosas and Mihalcea (2014), using a topic-neutral dataset similar to the DEFABEL corpus, achieved 60-70% accuracy by employing LIWC categories as features. This superior performance could possibly be attributed to the fact that the dataset contains opinions on controversial issues which invokes stronger personal involvement and emotional valence compared to our neutral-topic corpus.

5 Is Fact Verification Confounded by Deception and Factuality? (RQ3)

5.1 Methods

The core task in fact verification is to assess if a claim is factual (Thorne and Vlachos, 2018). This process determines the relation between the content of a piece of evidence and the content of a claim. Fact checking is commonly modeled as an entailment task, where given a claim-evidence pair a model is trained to predict whether the evidence (premise) supports or refutes the claim. During this step, we assume the evidence is given, i.e., selected beforehand. The fact checking model should predict the entailment label accurately, regardless of underlying claim and evidence properties. We hypothesize, however, that models may inadvertently draw on non-propositional cues of deception or the implicit knowledge they store from pertaining, thereby influencing their predictions. This opaque entanglement between deception and factuality may lead to (a) factual claims being more reliably verified compared to non-factual ones and (b) cause models to perform worse for instances in which the evidence is corrupted by a deceptive intention.

To investigate the impact of these properties, we compare the performance of fact checking models for (a) factual vs. non-factual statements and (b) instances with deceptive vs. non-deceptive evidence. Further, we inspect these properties in particularly difficult instances to understand if they might be challenging instances for the models, particularly because they convey non-factual or deceptive content.

5.2 Experimental Setup

Task. We frame fact checking as an entailment/natural language inference (NLI) task. Each instance is a premise-hypothesis pair. The source statement from DEFABEL is the hypothesis, while the argument is the premise. The models predict whether the claim is ENTAILED or CONTRADICTED by the evidence premise or if there is a NEUTRAL relation between the two. As all arguments in DEFABEL were written to support a statement, the NLI label for all instances is ENTAILMENT.

Models. We experiment with two off-the-shelf models, varying in architecture and size: mDeBERTa, a RoBERTa-based medium-sized model, trained for multilingual NLI[9] and Mistral7B-Instruct, an instruction-tuned LLM we use for few-shot prompting[10]. We provide all details for the experimental setup as well as prompt design in Appendix A.2. To understand the proficiency of these models for our use-case, we test their performance on DEFABEL. Table 3 presents the results. Both models show robust performance when evaluated on DEFABEL (full) with an F_1-score of .85 for mDeBerta and .86F_1 for the Mistral model on the target class ENTAILMENT[11].

5.3 Results

How do deception and factuality impact prediction performance? To understand how factuality and deception potentially confound the verification process, we compare the model performance on arguments supporting factual statements to their performance for non-factual statements. Analogously, we compare the performance for instances with deceptive evidence to instances with non-deceptive evidence.

Table 3 shows the results. To remove the effects of the imbalance of (non-)factual and (non-)deceptive instances in DEFABEL, we report results across all available instances and subsets of 50 claim-evidence pairs. See Appendix A.2

[9] https://huggingface.co/MoritzLaurer/mDeBERTa-v3-base-mnli-xnli

[10] https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.1

[11] We provide the full reports in Table 4.

sample	mDeBERTa			Mistral7B-Instruct		
	P	R	F_1	P	R	F_1
full	1.00	0.74	0.85	1.00	0.76	0.86
+fact	1.00	0.81	0.90	1.00	0.76	0.86
−fact	1.00	0.70	0.83	1.00	0.76	0.87
+fact^{50}	1.00	0.80	0.89	1.00	0.72	0.84
−fact^{50}	1.00	0.72	0.84	1.00	0.74	0.85
+decep	1.00	0.74	0.85	1.00	0.75	0.86
−decep	1.00	0.75	0.86	1.00	0.77	0.87
+decep^{50}	1.00	0.68	0.81	1.00	0.78	0.88
−decep^{50}	1.00	0.74	0.85	1.00	0.76	0.86

Table 3: Verification performance of `mDeBERTa` and `Mistral7B-Instruct`. We report results for the full DEFABEL dataset (full), for factual/non-factual (±fact), deceptive/non-deceptive (±decep) instances and subsets (50 instances) for each category.

for the sampling details. For mDeBERTa, factual instances are substantially more reliably verified compared to non-factual instances (Δ 7ppF_1). For Mistral, the performance is very similar across factual and non-factual instances (Δ 1ppF_1), with non-factual instances being slightly better verified. Regarding the deception property, both models show a slightly better performance for instances with non-deceptive evidence (Δ 1ppF_1 for mDeBERTa and Mistral, respectively). When comparing the performance on the subsets, we observe similar trends. Notably though, the Mistral model performs better on the instances with deceptive evidence. Based on the comparison of our results, we hypothesize that smaller models may be more susceptible to these confounding properties in the verification process, particularly for factuality.

What are the properties of particularly difficult instances? Finally, we want to understand if instances might get consistently mis-classified because they convey non-factual or deceptive content. We inspect the set of claim-premise pairs that get incorrectly classified by both models (120 instances). We find that the majority of instances are non-factual (72 %) and deceptive (62 %). While this is in line with the label distribution of deceptive instances in DEFABEL (62 %), the percentage of non-factual instances in the misclassified set is substantially higher (60 % in DEFABEL). This further corroborates that this property is a potential error source in the verification process.

We further investigate if errors in the prediction are in fact correlated with factuality and deception properties of the instances (see Appendix A.2), but do not find any meaningful connections.

6 Conclusion & Future Work

Belief-based deception, which disentangles factuality from deceptive intent, presents the possibility to investigate deceptive intent in isolation and gauge its impact on downstream applications, such as automatic fact verification.

In our correlation analysis, we found no clear links between linguistic cues of deception and expected patterns in German texts, highlighting the importance of considering cultural and language-specific differences in deception. Future research should investigate whether German texts exhibit unique linguistic patterns in deceptive contexts due to these factors. Furthermore, investigating belief-based data in the English language – currently unavailable to the best of our knowledge – could offer valuable insights into whether our negative findings are influenced by the framework. It is also important to note that the dataset contains labels for instances at the textual level rather than at the sentence level. Further investigation is necessary to explore potential variability in the deceptiveness of individual arguments.

Our automatic deception detection experiments show that computational models are not yet reliable. Further studies focusing on explainability using the current dataset could help understand the linguistic patterns the models are relying on for predicting deceptiveness. To enhance the robustness of deception detection tools, future work should prioritize incorporating context, cultural factors, and individual differences.

To gauge the impact of deception on fact checking, we explore if this property confounds verification models. A RoBERTa-based model shows lower performance for instances with non-factual claims and deceptive evidence documents, indicating that these instances are more difficult to verify. When using LLM prompting, performance is not substantially impacted by these properties. However, non-factual instances are a frequent error class for both models indicating that these properties might be challenging for the models. Future work should investigate which additional properties (non-)deceptive evidence or (non-)factual claims may exhibit to understand if the performance may be impacted e.g., by the choice of topic, argument structure or persuasive language.

Acknowledgments

This research has been conducted as part of the FIBISS project which is funded by the German Research Foundation (DFG, project number: KL 2869/5- 1 – 438135827). It was further supported by the CEAT project (DFG, KL 2869/1-2 – 380093645). We also acknowledge the use of the computing resources provided by the *bwUniCluster* funded by the Ministry of Science, Research and the Arts Baden-Württemberg and the Universities of the State of Baden-Württemberg, Germany, within the framework program bwHPC.

Limitations

Our study is limited to investigating previously reported linguistic cues of deception. While we consider it important to include extra-linguistic factors such as socio-cultural factors, and individual differences, it is not addressed in this study. Previous work studied mostly English, meaning cues could be exclusive to this language and transferable to German only in a limited way. For fact verification, we focus on the connection of factuality and deception with the fact checking label. These properties may also be tied to topics, which we did not investigate in this study. As outlined in Section 6 we see this as an opportunity for future work. Additionally, we do not include fake news as a form of deception in this study, as it represents a broader phenomenon involving the intentional spread of misinformation, distinct from individual acts of lying.

Ethical Considerations

Understanding deception and factuality from both linguistic and computational perspectives is vital for combating misinformation. Our work can therefore contribute to more robust and reliable efforts in detecting deceptive content and counter-acting the spread of false information. However, we are aware that the same insights could be misused – for example, to create more convincing disinformation or unfairly profile individuals. We therefore emphasize the need for responsible use of these models. Developing a system that detects lies solely based on textual content raises important questions about its feasibility and ethics. Since lying is not inherently a criminal act, labeling someone as a liar based on text analysis requires careful consideration of the implications.

References

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A next-generation hyperparameter optimization framework. In *The 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2623–2631.

Ángela Almela, Rafael Valencia-García, and Pascual Cantos. 2012. Seeing through deception: A computational approach to deceit detection in written communication. In *Proceedings of the Workshop on Computational Approaches to Deception Detection*, pages 15–22, Avignon, France. Association for Computational Linguistics.

Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. 2022. Fact checking with insufficient evidence. *Transactions of the Association for Computational Linguistics*, 10:746–763.

Katarina Boland, Pavlos Fafalios, Andon Tchechmedjiev, Stefan Dietze, and Konstantin Todorov. 2022. Beyond facts–a survey and conceptualisation of claims in online discourse analysis. *Semantic Web – Interoperability, Usability, Applicability*, 13(5):793–827.

Gary D. Bond and Adrienne Y. Lee. 2005. Language of lies in prison: linguistic classification of prisoners' truthful and deceptive natural language. *Applied Cognitive Psychology*, 19(3):313–329.

Carlo Bonferroni. 1936. Teoria statistica delle classi e calcolo delle probabilita. *Pubblicazioni del R Istituto Superiore di Scienze Economiche e Commericiali di Firenze*, 8:3–62.

Pasquale Capuozzo, Ivano Lauriola, Carlo Strapparava, Fabio Aiolli, and Giuseppe Sartori. 2020. DecOp: A multilingual and multi-domain corpus for detecting deception in typed text. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 1423–1430, Marseille, France. European Language Resources Association.

Bob de Ruiter and George Kachergis. 2018. The mafiascum dataset: A large text corpus for deception detection. *ArXiv*, abs/1811.07851.

Bella M. DePaulo, James J. Lindsay, Brian E. Malone, Laura Muhlenbruck, Kelly Charlton, and Harris Cooper. 2003. Cues to deception. *Psychological bulletin*, 129(1):74–118.

Yibing Du, Antoine Bosselut, and Christopher D. Manning. 2022. Synthetic disinformation attacks on automated fact verification systems. In *Proceedings of the Thirty-Sixth AAAI Conference on Artificial Intelligence*, volume 10.

Nicholas D. Duran, Charles Hall, Philip M. McCarthy, and Danielle S. McNamara. 2010. The linguistic correlates of conversational deception: Comparing natural language processing technologies. *Applied Psycholinguistics*, 31(3):439–462.

Tommaso Fornaciari, Federico Bianchi, Massimo Poesio, and Dirk Hovy. 2021. BERTective: Language models and contextual information for deception detection. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2699–2708, Online. Association for Computational Linguistics.

Tommaso Fornaciari, Leticia Cagnina, Paolo Rosso, and Massimo Poesio. 2020. Fake opinion detection: how similar are crowdsourced datasets to real data? *Language Resources and Evaluation*, pages 1–40.

Tommaso Fornaciari and Massimo Poesio. 2014. Identifying fake Amazon reviews as learning from crowds. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 279–287, Gothenburg, Sweden. Association for Computational Linguistics.

G.V. Glass and K.D. Hopkins. 1996. *Statistical Methods in Education and Psychology*. Allyn and Bacon.

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.

Zhijiang Guo, Michael Schlichtkrull, and Andreas Vlachos. 2022. A survey on automated fact-checking. *Transactions of the Association for Computational Linguistics*, 10:178–206.

Swati Gupta, Kayo Sakamoto, and Andrew Ortony. 2013. Telling it like it isn't: A comprehensive approach to analyzing verbal deception. Online.

Jeffrey T. Hancock. 2009. Digital deception: Why, when and how people lie online. In *Oxford Handbook of Internet Psychology*. Oxford University Press.

Casper Hansen, Christian Hansen, and Lucas Chaves Lima. 2021. Automatic fake news detection: Are models learning to reason? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 80–86, Online. Association for Computational Linguistics.

Momchil Hardalov, Arnav Arora, Preslav Nakov, and Isabelle Augenstein. 2022. A survey on stance detection for mis- and disinformation identification. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1259–1277, Seattle, United States. Association for Computational Linguistics.

Valerie Hauch, Iris Blandón-Gitlin, Jaume Masip, and Siegfried L. Sporer. 2015. Are computers effective lie detectors? a meta-analysis of linguistic cues to deception. *Personality and Social Psychology Review*, 19(4):307–342.

Sanchaita Hazra and Bodhisattwa Prasad Majumder. 2024. To tell the truth: Language of deception and language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8506–8520, Mexico City, Mexico. Association for Computational Linguistics.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of experts. *Preprint*, arXiv:2401.04088.

Ian Kelk, Benjamin Basseri, Wee Lee, Richard Qiu, and Chris Tanner. 2022. Automatic fake news detection: Are current models "fact-checking" or"gut-checking"? In *Proceedings of the Fifth Fact Extraction and VERification Workshop (FEVER)*, pages 29–36, Dublin, Ireland. Association for Computational Linguistics.

J. Peter Kincaid, Robert P. Fishburne Jr, Richard L. Rogers, and Brad S. Chissom. 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel. Technical Report 8-75, University of Central Florida, Institute for Simulation and Training.

Bennett Kleinberg, Yaloe Van Der Toolen, Aldert Vrij, Arnoud Arntz, and Bruno Verschuere. 2018. Automated verbal credibility assessment of intentions: The model statement technique and predictive modeling. *Applied cognitive psychology*, 32(3):354–366.

Bennett Kleinberg, Isabelle van der Vegt, Arnoud Arntz, et al. 2019. Detecting deceptive communication through linguistic concreteness. *PsyArXiv*.

Maximilian Köper and Sabine Schulte im Walde. 2016. Automatically generated affective norms of abstractness, arousal, imageability and valence for 350 000 German lemmas. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2595–2598, Portorož, Slovenia. European Language Resources Association (ELRA).

Paige E. Lloyd, Jason C. Deska, Kurt Hugenberg, Allen R. McConnell, Brandon T. Humphrey, and Jonathan W. Kunstman. 2019. Miami university deception detection database. *Behavior Research Methods*, 51:429–439.

Riccardo Loconte, Roberto Russo, Pasquale Capuozzo, Pietro Pietrini, and Giuseppe Sartori. 2023. Verbal lie detection using large language models. *Scientific reports*, 13:22849.

James Edwin Mahon. 2007. A definition of deceiving. *International Journal of Applied Philosophy*, 21(2):181–194.

Sandra Metts. 1989. An exploratory investigation of deception in close relationships. *Journal of Social and Personal Relationships*, 6(2):159–179.

Matthew L. Newman, James W. Pennebaker, Diane S. Berry, and Jane M. Richards. 2003. Lying words: Predicting deception from linguistic styles. *Personality and Social Psychology Bulletin*, 29(5):665–675.

Myle Ott. 2014. Linguistic models of deceptive opinion spam. In *Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, page 31, Baltimore, Maryland. Association for Computational Linguistics.

Myle Ott, Claire Cardie, and Jeffrey T. Hancock. 2013. Negative deceptive opinion spam. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 497–501, Atlanta, Georgia. Association for Computational Linguistics.

Myle Ott, Yejin Choi, Claire Cardie, and Jeffrey T. Hancock. 2011. Finding deceptive opinion spam by any stretch of the imagination. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 309–319, Portland, Oregon, USA. Association for Computational Linguistics.

James W. Pennebaker, Ryan Boyd, Kayla Jordan, and Kate Blackburn. 2015. *The development and psychometric properties of LIWC2015*. University of Texas at Austin.

Verónica Pérez-Rosas and Rada Mihalcea. 2014. Cross-cultural deception detection. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 440–445, Baltimore, Maryland. Association for Computational Linguistics.

Verónica Pérez-Rosas and Rada Mihalcea. 2015. Experiments in open domain deception detection. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1120–1125, Lisbon, Portugal. Association for Computational Linguistics.

Denis Peskov, Benny Cheng, Ahmed Elgohary, Joe Barrow, Cristian Danescu-Niculescu-Mizil, and Jordan Boyd-Graber. 2020. It takes two to lie: One to lie, and one to listen. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3811–3854, Online. Association for Computational Linguistics.

Dina Pisarevskaya, Tatiana Litvinova, and Olga Litvinova. 2017. Deception detection for the Russian language: Lexical and syntactic parameters. In *Proceedings of the 1st Workshop on Natural Language Processing and Information Retrieval associated with RANLP 2017*, pages 1–10, Varna, Bulgaria. INCOMA Inc.

Hannah Rashkin, Eunsol Choi, Jin Yea Jang, Svitlana Volkova, and Yejin Choi. 2017. Truth of varying shades: Analyzing language in fake news and political fact-checking. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2931–2937, Copenhagen, Denmark. Association for Computational Linguistics.

Gunning Robert. 1968. *The Technique of Clear Writing*. McGraw-Hill, New York.

Franco Salvetti, John B. Lowe, and James H. Martin. 2016. A tangled web: The faint signals of deception in text - boulder lies and truth corpus (BLT-C). In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3510–3517, Portorož, Slovenia. European Language Resources Association (ELRA).

Justyna Sarzynska-Wawer, Aleksandra Pawlak, Julia Szymanowska, Krzysztof Hanusz, and Aleksander Wawer. 2023. Truth or lie: Exploring the language of deception. *PLOS ONE*, 18(2):1–17.

Stephen Cameron Skalicky, Nicholas D. Duran, and Scott Andrew Crossley. 2020. Please, please, just tell me: The linguistic features of humorous deception. *Dialogue Discourse*, 11:128–149.

Felix Soldner, Verónica Pérez-Rosas, and Rada Mihalcea. 2019. Box of lies: Multimodal deception detection in dialogues. 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 1768–1777, Minneapolis, Minnesota. Association for Computational Linguistics.

Lyn M. Van Swol, Michael T. Braun, and Deepak Malhotra. 2012. Evidence for the pinocchio effect: Linguistic differences between lies, deception by omissions, and truths. *Discourse Processes*, 49(2):79–106.

Irina Temnikova, Silvia Gargova, Ruslana Margova, Veneta Kireva, Ivo Dzhumerov, Tsvetelina Stefanova, and Hristiana Krasteva. 2023. New bulgarian resources for studying deception and detecting disinformation. In *10th LANGUAGE AND TECHNOLOGY CONFERENCE: Human Language Technologies as a Challenge for Computer Science and Linguistics*. Adam Mickiewicz University Press.

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.

Ronny E. Turner, Charles Edgley, and Glen Olmstead. 1975. Information control in conversations: Honesty is not always the best policy. *The Kansas Journal of Sociology*, 11(1):69–89.

Aswathy Velutharambath and Roman Klinger. 2023. UNIDECOR: A unified deception corpus for cross-corpus deception detection. In *Proceedings of the 13th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis*, pages 39–51, Toronto, Canada. Association for Computational Linguistics.

Aswathy Velutharambath, Amelie Wührl, and Roman Klinger. 2024. Can factual statements be deceptive? the DeFaBel corpus of belief-based deception. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 2708–2723, Torino, Italia. ELRA and ICCL.

Ben Verhoeven and Walter Daelemans. 2014. CLiPS stylometry investigation (CSI) corpus: A Dutch corpus for the detection of age, gender, personality, sentiment and deception in text. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3081–3085, Reykjavik, Iceland. European Language Resources Association (ELRA).

Juraj Vladika and Florian Matthes. 2023. Scientific fact-checking: A survey of resources and approaches. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 6215–6230, Toronto, Canada. Association for Computational Linguistics.

Aldert Vrij. 2008. *Detecting lies and deceit: Pitfalls and opportunities*. John Wiley & Sons.

William Yang Wang. 2017. "liar, liar pants on fire": A new benchmark dataset for fake news detection. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 422–426, Vancouver, Canada. Association for Computational Linguistics.

Wenlin Yao, Zeyu Dai, Ruihong Huang, and James Caverlee. 2017. Online deception detection refueled by real world data collection. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 793–802, Varna, Bulgaria. INCOMA Ltd.

Lina Zhou, Judee K. Burgoon, Jay F. Nunamaker, and Doug Twitchell. 2004. Automating linguistics-based cues for detecting deception in text-based asynchronous computer-mediated communications. *Group Decision and Negotiation*, 13(1):81–106.

Miron Zuckerman, Bella M. DePaulo, and Robert Rosenthal. 1981. Verbal and nonverbal communication of deception. In Leonard Berkowitz, editor, *Advances in Experimental Social Psychology*, volume 14, pages 1–59. Academic Press.

A Modeling Details

A.1 Deception Detection

GBERT. Fine-tuning is conducted using the `BertForSequenceClassification`[12] implementation from Hugging Face. During fine-tuning, we set the number of epochs to 8, the learning rate to 10^{-5}, and the batch size to 16. To prevent overfitting, we monitor the training loss and stop if it does not decrease for 5 consecutive batches. We retain default values for all other hyperparameters unless specified.

Additionally, we conduct hyperparameter optimization using `Optuna` (Akiba et al., 2019). The optimization process involves defining a search space for several hyperparameters, including learning rate, number of epochs, train batch size, and eval batch size. The objective function evaluates different combinations of these hyperparameters by training the model and minimizing the training loss. Optuna's `study.optimize` method is employed to run multiple trials, automatically selecting the best set of hyperparameters based on the lowest training loss observed during the trials.

In the main body of the paper, we present the results obtained with the default hyperparameters, as both the best and default parameters yielded similar results. The model consistently predicted most instances as *deceptive*, making the default configuration a pragmatic choice for reporting results.

Computational details. The deception detection experiments were conducted on a cluster with distributed memory. The node we used featured an Intel Xeon Gold 6230 processor and a NVIDIA Tesla V100 accelerator. The GBERT model required approximately 30 minutes of GPU runtime to complete 10-fold cross-validation and evaluation on the holdout set, and approximately 5.5 hours for hyperparameter optimization. The prompt-based Mistral model consumed approximately 2.5 hours of GPU runtime to evaluate 5 different prompt settings. The prompts used for these evaluations are provided as part of the supplementary material.

A.2 Fact Verification

mDeBERTa. We use `mDeBERTa`[13], a RoBERTa-based medium-sized model, trained for multilingual NLI. We use the `transformers` library and provide the model with tokenized premise-hypothesis pairs. We convert the model output into probabilities for each class (ENTAILMENT, NEUTRAL, CONTRADICTION) represented by the logits using Softmax. We run the experiments on a single Nvidia GeForce RTX A6000 GPU.

Mistral. We use `Mistral-7B-instruct`[14] to prompt for NLI labels in the fact verification setting. In a conversational one-shot prompt setup, a fictional user describes setting, task, formatting instructions and an example instance. We imitate a one-turn conversation in which the LLM assistant provides the correct answer for the example instance. Subsequently, the user provides the assistant with the actual instance. Refer to Table 6 for the prompt template. We provide the initialized prompts as part of the supplementary material. For each input prompt, we apply the respective chat template[15] and generate the output sequence with following parameters: max_new_tokens=1024, temperature=0.3, do_sample=True, top_p=0.95, top_k=50, repetition_penalty=1.2.

We instruct the model to provide the output in JSON format. If the model does not generate a label within the label space, or does not provide JSON formatted text, we randomly assign the instance to one of the three target classes. Note that this concerns a total of 31 instances.

We run our experiments on a GPU server with 4 Nvidia GeForce GTX 1080 Ti GPU nodes. Generating the responses for all instances in DEFABEL takes approx. 5 hours.

Subsampling. DEFABEL consists of an unbalanced number of factual vs. non-factual source statements. To understand if this impacts the results, we draw a sample to evaluate on. We further want to ensure that the source statements do not overlap between the subsets. Out of the 30 source statements (11 factual, 19 non-factual), we sample 10 statements and subsequently sample 5 arguments per statement, leaving us with 50 pairs each for evaluation.

Are prediction errors correlated with deception & factuality? We investigate if prediction errors are correlated with a) factuality of statements and b) deceptive intent in evidence premises. We cal-

[12] `https://huggingface.co/transformers/model_doc/bert.html`

[13] `https://huggingface.co/MoritzLaurer/mDeBERTa-v3-base-mnli-xnli`

[14] `https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.1`

[15] `https://huggingface.co/docs/transformers/main/en/chat_templating`

culate Pearson's correlation between the two binary variables ((in)correctly predicted and (non-)factual/(non-)deceptive). For mDeBerta, we observe a correlation of 0.12 with the factuality property (p-value$<$0.05). We do not find significant correlations for the other property and model, indicating that there might be more complex textual properties indirectly linked to the deception label that impact the verification process.

B Prompt design

Table 5 shows details on the prompts for deception detection and 6 for fact checking. Please refer to Sections 4 and 5 for explanations how these are used in our experiments.

Is factuality ignored consistently? To make sure that the model follows instructions and ignore the factual accuracy of statements, we modify the last line of `DECEPT` prompt to `{“Is the text deceptive?”': Yes or No, "Reason": }`. On manually inspecting the reasons, we see that while the model sticks to the instructions in most cases, it is not consistent in ignoring the factual accuracy. For instance, it uses arguments like the following:

- “*the author ignores basic facts about marine life and fish behavior to deceive readers*”
- “*the statement is not supported by factual evidence.*”
- “*the text is factually accurate and includes evidence to support the claim*”
- “*the text presents a clear, factual statement*”

C Linguistic cues

We show details on the operationalization of the linguistic cues in Table 7. Please see Section 3 for details on the use of these features.

		mDeBERTa				Mistral7B-Instruct			
		P	R	F$_1$	S	P	R	F$_1$	S
full	Neutral	0.00	0.00	0.00	0	0.00	0.00	0.00	0
	Contradict	0.00	0.00	0.00	0	0.00	0.00	0.00	0
	Entailed	1.00	0.74	0.85	1031	1.00	0.76	0.86	1031
	micro avg			0.74	1031			0.76	1031
	macro avg	0.33	0.25	0.28	1031	0.33	0.25	0.29	1031
	weighted avg	1.00	0.74	0.85	1031	1.00	0.76	0.86	1031
+fact	Neutral	0.00	0.00	0.00	0	0.00	0.00	0.00	0
	Contradict	0.00	0.00	0.00	0	0.00	0.00	0.00	0
	Entailed	1.00	0.81	0.90	376	1.00	0.76	0.86	376
	micro avg			0.81	376			0.76	376
	macro avg	0.33	0.27	0.30	376	0.33	0.25	0.29	376
	weighted avg	0.33	0.27	0.30	376	1.00	0.76	0.86	376
−fact	Neutral	0.00	0.00	0.00	0	0.00	0.00	0.00	0
	Contradict	0.00	0.00	0.00	0	0.00	0.00	0.00	0
	Entailed	1.00	0.70	0.83	655	1.00	0.76	0.87	655
	micro avg			0.70	655			0.76	655
	macro avg	0.33	0.23	0.28	655	0.33	0.25	0.29	655
	weighted avg	1.00	0.70	0.83	655	1.00	0.76	0.87	655
+decep	Neutral	0.00	0.00	0.00	0	0.00	0.00	0.00	0
	Contradict	0.00	0.00	0.00	0	0.00	0.00	0.00	0
	Entailed	1.00	0.74	0.85	643	1.00	0.75	0.86	643
	micro avg			0.74	643			0.75	643
	macro avg	0.33	0.25	0.28	643	0.33	0.25	0.29	643
	weighted avg	1.00	0.74	0.85	643	1.00	0.75	0.86	643
−decep	Neutral	0.00	0.00	0.00	0	0.00	0.00	0.00	0
	Contradict	0.00	0.00	0.00	0	0.00	0.00	0.00	0
	Entailed	1.00	0.75	0.86	388	1.00	0.77	0.87	388
	micro avg			0.75	388			0.77	388
	macro avg	0.33	0.25	0.29	388	0.33	0.26	0.29	388
	weighted avg	1.00	0.75	0.86	388	1.00	0.77	0.87	388

Table 4: Verification performance (Precision, Recall, F$_1$, Support) of `mDeBERTa` and `Mistral7B-Instruct`. We report results for the full DEFABEL dataset (full), for factual/non-factual (±fact) and deceptive/non-deceptive (±decep) instances.

Name	Prompt structure
DECEPT	‘**system_prompt**’: "You are analyzing German text to identify deceptive texts based on linguistic cues only.", ‘**user_prompt**’: “The task is to evaluate a text and determine whether they exhibit signs of deception. Focus solely on the linguistic properties of the text and disregard any considerations of factual accuracy. Look for patterns of language, logic, or presentation that may indicate an attempt to mislead or manipulate the reader. In each text, pay attention to the cues of deception like: **<list of cues>** Keep in mind that factual inaccuracy alone cannot be considered a sign of deception. Some individuals may genuinely believe misinformation. Therefore, focus solely on linguistic cues to identify deception. After analyzing each text for these cues, determine whether they exhibit signs of deception or not. Provide your assessment based solely on the linguistic properties of the text, disregarding any considerations of factual accuracy. Look at the following example: **<example>** This text is deceptive because of the following reason: **<reasons>** Now judge the following text referring to the information above: “**<argument>**”. Judge the presence of different linguistic cues of deception. Think also whether the arguments are supporting a factual statement. Make sure factual accuracy of the statement is not affecting your judgement about the deceptive nature of the text. Strictly format your answer exactly as the following json and do not include any explanations in the output. {“Is the text deceptive?”’: Yes or No}”
DECEPT_FACT	DECEPT + {“Are the arguments in the text factual?”: Yes or No, “Is the text deceptive? ”: Yes or No}”
CONV_DECEPT	“**role**”: “**user**”, “**content**”: ’You are an AI agent who can analyse German text to identify deceptive texts based on linguistic cues only.’, “**role**”: “**assistant**”, “**content**”: ’As an AI analyzing German text, I confidently spot deception using linguistic cues like inconsistencies and evasion. Yet, I also recognize the importance of considering context and evidence for precise evaluation.’, “**role**”: “**user**”, “**content**”: ’Your task is to evaluate a text and determine whether they exhibit signs of deception. Focus solely on the linguistic properties of the text. It is very important that you strictly discard factual accuracy of the statement as there are instances where the author believes in a factually inaccurate statement and is arguing in favor of it, which is non-deceptive. Do you understand? ’, “**role**”: “**assistant**”, “**content**”: ’Yes, I understand. I will focus solely on the linguistic properties of the text to evaluate whether it exhibits signs of deception, disregarding the factual accuracy of the statement.’, “**role**”: “**user**”, “**content**”: “In each text, pay attention to the cues of deception like: **<list of cues>** Keep in mind that factual inaccuracy alone cannot be considered a sign of deception. Some individuals may genuinely believe misinformation. Therefore, focus solely on linguistic cues to identify deception. After analyzing each text for these cues, determine whether they exhibit signs of deception or not. Provide your assessment based solely on the linguistic properties of the text, disregarding any considerations of factual accuracy.”, “**role**”: “**assistant**”, “**content**”: ’Understood. Please provide the texts for analysis, and I will evaluate them based solely on the linguistic cues you have outlined.’, “**role**”: “**user**”, “**content**”: “Judge whether or not the following text is deceptive. Return result as a json “Is the text deceptive”: **Yes or No** text: “**<example>**” , “**role**”: “**assistant**”, “**content**”: Answer: “Is the text deceptive”: “Yes” This text is deceptive because of the following reason: **<reasons>** “**role**”: “**user**”, “**content**”: Judge whether or not the following text is deceptive. text: “**<argument>**” Return result as a json {“Is the text deceptive”: **Yes or No**}. Keep explanation short.
CONV_DECEPT_FACT	CONV_DECEPT + {“Are the arguments in the text factual?”: Yes or No, “Is the text deceptive? ”: Yes or No}”
CONV_DECEPT_RETHINK	CONV_DECEPT + “**role**”: “**user**”, “**content**”: “Are the arguments in the text supporting a factual statement?”, “**role**”: “**assistant**”, “**content**”: “Yes. They are arguing that the earth is round. This statement is factually accurate.”, “**role**”: “**user**”, “**content**”: “Is the text deceptive irrespective of the factuality of the statement it is supporting? Strictly format your answer exactly as the following json and do not include any explanations in the output “Answer: “Is the text deceptive?”: Yes or No”, “**role**”: “**assistant**”, “**content**”: Answer: “Is the text deceptive?”: “Yes””, “**role**”: “**user**”, “**content**”: Here is another text. Judge the presence of different linguistic cues of deception. Think also whether the arguments are supporting a factual statement. Make sure factual accuracy of the statement is not affecting your judgment about the deceptive nature of the text. ...

Table 5: Structure of prompts used for the deception detection experiments.

user	You're going to be working on a natural language inference (NLI), also known as textual entailment task. As you probably know that this is a subfield of natural language processing that deals with determining if one sentence or text implies another sentence or text. Your goal is to determine whether a hypothesis statement is entailed or refuted by premise sentence(s). You will be presented with a premise and a hypothesis pair. Note that the texts are in German. Then your task is to categorize pair into one of three categories: 1) Entailment: The hypothesis is entailed by the premise, 2) Neutral: The hypothesis is neutral to the premise, or 3) Contradiction: The hypothesis contradicts the premise. Provide the output in a json format with the key being 'label' and the value being the category number as an integer. For example, if you believe the post should be categorized as 3) Contradiction , your json output should be: {'label': 3} Now consider the following pair: Premise: «Die Beweise für die Rundum-Theorie der Erde sind überwältigend. Astronomische Beobachtungen, Raumfahrt und Phänomene der Zeitzonen bestätigen alle die kugelförmige Gestalt der Erde. Bilder von Weltraummissionen zeigen deutlich die Krümmung unseres Planeten und bestätigen damit jahrhundertelangen wissenschaftlichen Konsens. Fringe-Theorien, die eine flache Erde behaupten, stützen sich jedoch auf ausgewählte Daten und emotionale Appelle, anstatt auf empirische Beweise. Obwohl es wichtig ist, alternative Standpunkte zu berücksichtigen, widerspricht die Leugnung der Realität einer runden Erde etabliertem wissenschaftlichem Wissen.»Hypothesis: «Die Erde ist eine Kugel.» What is the correct label for this pair?
assistant	Here is the correct answer formatted as json: {'label': 1}
user	Perfect! Here is another post: Premise: **<premise>** Hypothesis: **<hypothesis>**What is the correct label for this pair? Only output the json, no additional text or explanation.

Table 6: Prompt template for generating NLI-based fact verification labels. We provide the instantiated prompts in the supplementary material.

Linguistic cues	**Features**	**Operationalization**
Complexity	Token count	# tokens
	Sentence count	# sentences
	Unique token count	# unique tokens
	Type-token ratio / lexical diversity	# unique tokens / # tokens
	Average postag count	# postag / # tokens calculated per postag
	Flesch–Kincaid readability	calculated using `textstat` python library
	Gunning–Fox readability	calculated using `textstat` python library
Concreteness	Abstractness score	from Köper and Schulte im Walde (2016)
	Imageability	from Köper and Schulte im Walde (2016)
Sentiment	Sentiment score	Positive, negative, and neutral scores using Guhr et al. (2020)
	Arousal	from Köper and Schulte im Walde (2016)
	Valence	from Köper and Schulte im Walde (2016)
LIWC	99 psychological categories	Relative frequency for from DE-LIWC2015

Table 7: Linguistic cues and their operationalization