The King is Naked: on the Notion of Robustness for Natural Language Processing

Emanuele La Malfa, Marta Kwiatkowska
Department of Computer Science
University of Oxford
{emanuele.lamalfa, martakwiatkowska}@cs.ox.ac.uk

Abstract
There is growing evidence that the classical notion of adversarial robustness originally introduced for images has been adopted as a de facto standard by a large part of the NLP research community. We show that this notion is problematic in the context of NLP as it considers a narrow spectrum of linguistic phenomena. In this paper, we argue for semantic robustness, which is better aligned with the human concept of linguistic fidelity. We characterize semantic robustness in terms of biases that it is expected to induce in a model. We study semantic robustness of a range of vanilla and robustly trained architectures using a template-based generative test bed. We complement the analysis with empirical evidence that, despite being harder to implement, semantic robustness can improve performance on complex linguistic phenomena where models robust in the classical sense fail.

Introduction
In the last decade, deep learning has become the gold standard method to solve complex problems in Natural Language Processing (NLP) (Brown et al. 2020). The range of NLP applications encompasses text classification (Liu et al. 2019), language translation (Liu et al. 2020), and now also ranking systems and large-scale search engines (Wang et al. 2021). With models whose complexity – and consequently, size – has become gargantuan

1, there is an increasing concern about reproducibility (Liu et al. 2021b) and reliability of those models (Song et al. 2020), as it is known that, even for smaller networks, it is possible to exploit their brittleness with techniques of adversarial machine learning (Zhang et al. 2020). Consequently, concepts of robustness have been transferred from adversarial learning to NLP, resulting in techniques and tools (Li et al. 2020a) that typically check that the network’s decision is invariant to a simple bounded perturbation (word substitution or deletion) for a given input (local robustness), working in the (continuous) embedding space or the (discrete) word neighbourhood. However, NLP still lacks a definition of robustness that properly captures linguistic phenomena and is aligned with human common sense (Xu et al. 2020). There is currently a debate in the NLP community about the internal working of language models, with some believing they are the ‘foundation’ for the entire discipline (Bommasani et al. 2021) and others arguing that they mostly learn higher-order distributions of words frequency (Sinha et al. 2021).

In this work, we first review the classical notions of robustness adopted in NLP and identify their weaknesses, in terms of the lack of expressiveness and over-reliance on the neural model text representation. Next, to better align the perception of human robustness to that implemented by a neural model, we formalise (local) semantic robustness of NLP as a notion that generalizes local discrete robustness through measuring robustness to linguistic rules, rather than to word substitution or deletion. This allows us to define (global) semantic robustness for a linguistic task such as sentiment analysis, which can be extended to higher-order tasks. We contribute to the debate in the NLP community by performing a systematic comparison, complemented by an evaluation of different architectures, of the classical notions of robustness in NLP. We further show that with semantic robustness we can evaluate the performance of a model on cogent linguistic phenomena, which are of interest for both the NLP and the linguistics community. We achieve this by proposing an assessment framework and a simple, yet effective, test bed based on data augmentation. Last but not least, we wish to highlight the issue of NLP robustness, which for the last few years has over-focused on trivial and often machine-centric symbol manipulation. Using the terminology drawn from cyber-security, this work is a purple-team effort to align the key performance indicators of the red-team – whose role is to exploit NLP models with any kind of vulnerability – with those of the blue-team, a.k.a. the defenders, who aim to adopt a semantic notion of robustness that implies robustness to linguistic phenomena.

Related Work
Brittleness of neural network models is a serious concern, both theoretically (Biggio et al. 2013; Szegedy et al. 2014) and practically, including Natural Language Processing (NLP) (Belinkov and Bisk 2018; Ettinger et al. 2017; Gao et al. 2018; Jia and Liang 2017; Liang et al. 2017; Zhang et al. 2020) and more recently complex Masked Language Models (MLM) (Li et al. 2020b; Sun et al. 2020). In NLP, attacks are usually conducted either at character or word level (Ebrahimi et al. 2017; Cheng et al. 2018), or

Copyright © 2022, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

1GPT-3’s full version has 175 billion learning parameters.
at the embedding level, exploiting (partially or fully) vulnerabilities in the symbols’ representation (Alzantot et al. 2018; La Malfa et al. 2021). Brittleness of NLP does not pertain only to text manipulation, but also includes attacks and complementary robustness for ranking systems (Goren et al. 2018). Neural network robustness naturally complements the perspective offered by brittleness as it involves the certification of a model against a wide range of attacks (Huang et al. 2017). In NLP, similarly to computer vision (Akhtar and Mian 2018), the majority of works have adopted the narrow notion of robustness, in terms of invariance to minor perturbations of an input text (Gowal et al. 2018; Jia et al. 2019; Dong et al. 2021; La Malfa et al. 2020), while only a minority have contested these limitations, either implicitly (Ribeiro et al. 2020) or explicitly (Morris 2020; Morris et al. 2020a; Xu et al. 2020), mainly due to the difficulty of automatically generating semantically involved test beds (Feng et al. 2021). Although adversarial data augmentation in NLP is well established (Morris et al. 2020b), robustness to semantically coherent, yet possibly diverging, examples is still in its ‘adolescence’ (Ribeiro, Singh, and Guestrin 2018), as many highly accurate NLP models cannot recognize cogent linguistic phenomena even on low-order tasks such as binary classification (Barnes, Øvrelid, and Velldal 2019).

**NLP Robustness: a Tale of Two Perspectives**

In this section we discuss the merits of local robustness in NLP, analysing existing concepts and highlighting the issues with local continuous robustness. We then introduce the notion of semantic robustness, aimed to better align the perception of human robustness to those implemented by neural models, together with an assessment framework. In the next section, we complement the methodology part with an experimental evaluation, where neural networks’ robustness is tested against linguistic phenomena. We complete the paper with a study of the inductive biases that different notions of robustness are expected to induce in a trained model.

**Notation.** We will refer to \( f(\cdot) \) as a generic neural network that solves a task \( T \) for an instance \( s \), which is represented as a piece of text written in natural language (e.g., sentiment analysis). W.l.o.g., we will assume that texts are represented as lists of words (features), namely \( s = (v_1, \ldots, v_l) \). We will denote with \( x \in \mathbb{R}^{d} \) a text \( s \) whose \( 1 \) features have been mapped to vectors of real numbers through an embedding, \( \mathcal{E} : \mathcal{V} \to \mathbb{R}^{d} \), where \( \mathcal{V} \) is a finite vocabulary of words. We refer to a component of \( x \) along a generic embedding axis as \( x^{(i)} \in \mathbb{R} \), \( i \in \{1, \ldots, d\} \). Since the majority of the embedding spaces are injective non-surjective functions, the notation \( x_v = (x_{v_1}, \ldots, x_{v_l}) \) will serve to denote an embedded text \( x \) that further admits, for each vector \( x_v \), a preimage in the vocabulary space, i.e., \( \forall x_v \in x_v \exists ! v \in \mathcal{V} : \mathcal{E}(v) = x_v \). With a slight abuse of notation, we will denote with \( \mathcal{E}(s) = (x_{v_1}, \ldots, x_{v_l}) \) a text whose words have each been embedded through \( \mathcal{E} \). Finally, we will assume that the first operation of the model \( f(\cdot) \) involves a transformation through an embedding representation \( \mathcal{E} \).

**Classical Notions of NLP Robustness**

We begin by discussing the concept of local continuous robustness, which is widely used in computer vision and has been applied to NLP (Huang et al. 2019a; Jia and Liang 2017; La Malfa et al. 2020). We then consider local discrete robustness, which manipulates symbols rather than embedding vectors (Alzantot et al. 2018). We show that the former notion can be reduced to the latter: nonetheless, both definitions only allow one to express robustness to a limited number of linguistic phenomena. We extensively discuss the advantages and drawbacks of those two notions.

**Definition 1 (Local Continuous Robustness).** A model \( f(\cdot) \) is locally robust to \( \epsilon \)-bounded perturbations when, given a task \( T \) and one of its instances \( x \), it holds that \( \forall x' \in Ball_{\epsilon}(x), f(x) = f(x') \), where \( Ball_{\epsilon}(x) = \{ x' : ||x - x'||_{p} \leq \epsilon \} \), \( || \cdot ||_p \) is an \( L_p \) norm of choice and \( \epsilon \geq 0 \) a (small) real number.

**Observation 1.** Natural language is discrete while local continuous robustness is defined over a dense representation. Standard embedding techniques (Mikolov et al. 2013; Pennington, Socher, and Manning 2020) define the word-to-vector mapping over a continuous space, with the input vocabulary discrete and finite (e.g., characters, words, sentences) and the output dense and uncountable. On the other hand, natural language is discrete and allows for finite, yet combinatorial, outcomes. In this hybrid setting, \( \epsilon \)-bounded robustness implies that any vector in this dense \( \epsilon \)-bounded region is safe. This assumption is linguistically inconsistent, as a network may present a decision boundary where an adversarial attack that is not a proper word limits the verification or severely reduces the safe region. We illustrate this issue in Figure 1.

**Definition 2 (Local Discrete Robustness).** A model \( f(\cdot) \) is locally robust to discrete perturbations when, given a task \( T \) and an instance \( x_v \) embedded from \( s \), it holds that \( \forall x'_v \in D-Ball_{\epsilon}(x_v), f(x_v) = f(x'_v) \), where \( D-Ball_{\epsilon}(x_v) = \mathcal{E}(V)^l \cap Ball_{\epsilon}(x_v) \).

We exemplify the differences between \( Ball_{\epsilon} \) and the corresponding \( D-Ball_{\epsilon} \) in Figure 2.

**Proposition 1.** Local continuous robustness implies local discrete robustness, but the converse is generally false.

**Proof.** From a mathematical perspective, \( D-Ball_{\epsilon}(x) \subset Ball_{\epsilon}(x) \) but the opposite is not true. For \( \epsilon = 0 \), both \( D-Ball \) and \( Ball \) are singletons. ■

**Observation 2.** Continuous and discrete robustness allow but limited syntax manipulations. As demonstrated empirically by many existing works in the literature (Alzantot et al. 2018; Jia et al. 2019; Huang et al. 2019a; Dong et al. 2021), both formulations of robustness only allow for robustness testing against symbol-to-symbol substitutions or deletions. The limited degree of freedom of an operator that locally substitutes a word with other words makes it hard, if not impossible, to test for robustness against paraphrases. As an example, if a model \( f(\cdot) \) is robust for the sentence “the movie was good”, which implies correct classification for
the texts “the film was good”, “the film was nice”, etc., we cannot say the same for the sentence “an enjoyable thriller”. From the linguistic perspective, this problem arises since the frequency of words in natural language follows the Zipf’s law (Zipf 2013), where rare terms and constructs – hence edge cases – occur more frequently than in other natural phenomena.

Observation 3. There is no guarantee that perturbations in both discrete and continuous settings do not violate the task under consideration. As the methods that implement both discrete and continuous robustness allow for weak supervision in the choice of the replacements, a perturbation can diverge from the task under consideration. It is well known that many recent embeddings have been developed to be faithful to a (static) version of the “distributional hypothesis” (Baroni, Dinu, and Kruszewski 2014), and thus it is not unusual to find words like “bad” and “good” close to each other in the representations. This could lead to potentially disastrous effects when balancing local robustness, e.g., (Gowal et al. 2018), with accuracy, especially for low-order tasks such as sentiment analysis.

A Semantic Notion of Robustness

We now introduce a notion of robustness that goes beyond word replacements, and thus permits an assessment of the brittleness to linguistic phenomena that are cogent to humans. To do so, we first need to introduce some notation.

Definition 3 (Oracle). An Oracle \( \Omega \) solves a task \( T \) for any input \( s \) that is compliant with \( T \), while it rejects all those instances that are not. We denote with \( \Omega \models s \) the act of solving a task, and with \( \Omega \not\models s \) the rejection. Solving and rejection are mutually exclusive.

Observation 4. An Oracle is an augmented, idealized linguistic model. There are two cogent differences between an Oracle \( \Omega \) and a standard model \( f(\cdot) \): (i) a model can be wrong on samples from \( T \) (e.g., misclassifications) while the Oracle is always right about its decisions; (ii) the Oracle (certainly) rejects inputs that are not compliant with the task \( T \).

Example 1. An Oracle for sentiment analysis. Given a sentiment analysis task \( T \) for movie reviews, an Oracle \( \Omega \) correctly classifies any text that expresses a judgment about the movie. As an example, “the movie was (not) good” will be classified as positive (negative). An Oracle alternatively rejects any piece of text that is inconsistent with \( T \), i.e., all those texts that do not explicitly (or implicitly) express a judgment about a movie. An example is the text “recipe of risotto with mushrooms: [...]”, which is rejected. From a practical perspective, a classifier \( f(\cdot) \) admits misclassifications in the sense that its accuracy may not be maximal (i.e., less than 1.) and it further cannot reject inputs that are not compliant with the task: in the case it does, the task is not fulfilled perfectly.

Definition 4 (Linguistic Rule). A linguistic rule is a symbolic function that manipulates a text \( s \) according to a linguistic phenomenon and a task \( T \), whose generated texts \( S' \), along with the original input, are not rejected by \( \Omega \). Formally, \( R : (s, T) \rightarrow S' \), \( \forall s' \in S', \Omega \models s \land s' \).

Observation 5. Linguistic rules are flexible symbolic methods. Since a linguistic variation of an input can be very different from the original text, a rule should be allowed to add/remove/replace words while remaining com-
pliant with the task $T$. As an example of a simple rule, one can think of verb negation that acts on a text and negates the action expressed by the subject (if any). While this task is often trivial for humans, fully algorithmic solutions to this problem are still limited in their capabilities (Guo et al. 2018). Hybrid methods, based on synthetic data augmentation, humans-in-the-loop and deep MLM (Feng et al. 2021; Lin et al. 2019; Huang et al. 2019b), constitute currently an active area of NLP research. One viable way to generate the replacements is to use template-based data-augmentation techniques (as employed in this paper and detailed in Experimental Evaluation). More complex approaches involve MLM with humans-in-the-loop who validate the generated perturbations. While for the generative process the MLM can be trained to be controlled through textual ‘seeds’ (e.g., in the spirit of the works by (Wu et al. 2021) or (Madaan et al. 2020)), humans play the role of the Oracle.

**Example 2.** A rule for shallow negation. For a sentiment analysis task $T$ with positive and negative instances and a positive instance $s$ “the movie was good”, the shallow negation rule $R$ negates the sentiment expressed by $s$, and hence valid perturbations generated by $R$ on $s$ are “the movie was not good”, “a bad film”, but also more involved examples like “it is false that the movie is good”, etc. We name this rule shallow negation as it does not allow for nested negations, regardless of their grammatical consistency (i.e., “it is false that the movie wasn’t good” cannot be generated by $R$ on $s$).

**Definition 5 (Local Semantic Robustness).** Formally, given a model $f(\cdot)$, a linguistic rule $R$, an input $s$ from a task $T$, a measure of the performance of $f(\cdot)$ on $T$, namely $p \in [0,1]$ (with 0. denoting a random guess and 1. perfect accuracy), a small number $\tau \geq 0$, and a measure of performance $p'$ on samples $S'$ generated by $R(s,T)$, we say $f(\cdot)$ is $\tau$-semantically robust for $R$ and $s$ if it holds that $E_{s' \sim S'}[p'] \geq p - \tau$, with $\min(0, p - \tau) = 0$.

We further say that a model is bounded invariant to a rule $R$ when it holds that $p - \tau \leq E_{s' \sim S'}[p'] \leq p + \tau$.

Informally, a model $f(\cdot)$ that correctly classifies an instance $s$ of a task $T$ is semantically robust to a linguistic rule $R$ when it exhibits at least the same performance on the set $S'$ of perturbations $s'$ generated by applying $R$ to $s$. We further observe that this formulation allows for the performance $p'$ to even surpass $p$, so this notion entails that $f(\cdot)$ is no worse at correctly solving $T$ for $S'$ than it is at solving any other task, and is hence a stronger notion than bounded invariance.

**Observation 6.** Local semantic robustness is linguistically meaningful. The notion is local as linguistic rules act on a single text $s$. It is further inherently linguistic as the transformation $R$ of an input text acts at the syntax level but then the Oracle’s reject phase guarantees it has preserved the semantics of each $s'$ w.r.t. $T$.

**Observation 7.** Local semantic robustness is entailed by linguistic generalization, but not the other way round. Linguistic robustness is different from generalization on unseen test cases. The former is entailed by the latter, while the other way round is not necessarily true. Semantic robustness is defined over a rule while generalization is a more general and hard to obtain/optimize objective.

**Proposition 2.** Local semantic robustness can be reduced to local discrete robustness, but not to local continuous robustness.

**Proof.** For local discrete robustness, it is straightforward to define a rule that generates perturbations according to the definition of local discrete robustness. In this sense the semantic rule $R$ involves extracting the replacements in the embedding’s neighborhood of each input word.

As regards local continuous robustness, the invariance over all the input texts $s'$ in a $\epsilon$-ball cannot be mapped back to the embedding (a.k.a. input) vocabulary $V$ by any combination of linguistic rules as they act, by definition, at the symbol level. Since the majority of continuous embeddings are injective non-surjective functions, almost all the vectors in any non-empty region of the space cannot be mapped back to a proper entry of $V$.

**Definition 6 (Semantic Robustness).** This notion extends local semantic robustness beyond the single instance and to a specific task $T$. A model $f(\cdot)$ exhibits semantic robustness (or in general semantic robustness) to a rule $R$ and a task $T$ when it is locally semantically robust for any input $s'$ generated by applying $R$ to a test set.

**Assessment Framework for Semantic Robustness** A sufficient condition for quantifying the semantic robustness of a model on an NLP task is that it is possible to measure the performance of such a model on unseen input texts. In this sense, we can measure the semantic robustness of a model $f(\cdot)$ that solves a task $T$ by comparing its performance $p$ with the performance $p'$ of the model on an unseen test bed that contains one or more semantic phenomena.

We now describe some illustrative examples of measuring semantic robustness, firstly for sentiment analysis and then for more involved NLP tasks.

**Example 3.** Robustness to shallow negation in sentiment analysis. Given a sentiment analysis task with positive and negative instances, a model $f(\cdot)$ trained on a dataset $(S, Y)$ and validated on $(S_{test}, Y_{test})$ is robust to shallow negation when $\forall s \in S_{test}, \forall s' \in S' = R(s, T), (\Omega \models s \land s') \Rightarrow E_{s' \sim S'}[p'] \geq p - \tau$ for some $\tau \geq 0$, with $R$ the negation rule that acts on a specific text and negates the sentiment expressed by $s$. In this sense, $p$ represents the accuracy of the trained model on $(S_{test}, Y_{test})$, while $p'$ is the accuracy measured on a subset of samples that contain specific linguistic phenomena. We remark that a test bed can be handcrafted, as we show in our paper, or distilled from existing datasets, as described in (Barnes, Øvrelid, and Veldal 2019).

**Example 4.** Semantic robustness in high-order NLP tasks. We now briefly sketch how we would approach the measurement of semantic robustness for higher-order NLP tasks. For Question and Answer (QA) tasks, a measure of robustness can be quantified as the gap between the ‘unexpectedness’ of an Answer when the Question does/doesn’t contain a lin-
guistic phenomenon. In Natural Language Inference (NLI), directly applying our framework would be straightforward since NLI is reducible to a classification task. In the same way, when Read and Comprehension (RC) is pursued in the form of a classification task, the evaluation of semantic robustness would be similar to sentiment analysis or NLI, whereas when the answer requires re-elaborating the input, the measurement of semantic robustness would be similar to QA (with possibly a different evaluation metric for T).

**Experimental Evaluation**

We next conduct an extensive experimental evaluation\(^2\) designed to answer the following research questions: (i) whether models robust in the classical sense are also semantically robust; (ii) whether robustness to specific linguistic phenomena is a by-product of training accurate NLP classifiers; (iii) whether, for different architectures, augmented supervised training – with texts that contain a specific linguistic phenomenon – induces generalization on unseen test samples that contain the same phenomenon; (iv) whether it is possible to train models that are both accurate and semantically robust, and (v) to what extent unsupervised learning contributes to semantic robustness.

We conduct the experiments on models trained – or fine-tuned through data augmentation – on the Stanford Sentiment Treebank dataset (SST-2) (Socher et al. 2013) and on the dataset collected by (Barnes, Øvrelid, and Velldal 2019). The advantages of this approach are two-fold. Firstly, human experts have collected/handcrafted sentences whose syntax/semantics is rich and the level of noise restrained. Secondly, since in NLP spurious patterns and over-fitting play a crucial role during training whose influence is hard to estimate and quantify, cogent compactness of those datasets makes it relatively easy to assess the results. To further estimate the robustness on linguistic phenomena, in the spirit of the evaluation done in (Huang et al. 2019b), we utilise a template-based method, whose details are given below, for generating augmented samples for a selection of linguistic phenomena. In Natural Language Inference (NLI), the form of a classification task, the evaluation of semantic robustness would be similar to sentiment analysis or NLI, whereas when the answer requires re-elaborating the input, the measurement of semantic robustness would be similar to QA (with possibly a different evaluation metric for T). In a template, part of the text is fixed while the remaining part is symbolically represented by tokens which are iteratively replaced by combinations of words from candidate perturbation sets. The augmentation preserves the semantics of the sentence while introducing a linguistic phenomenon (such as shallow negation). In our implementation, the rules, a perturbation cannot change the template’s label: in this sense, the rejection phase (see Definition 3) is embedded in the generative pipeline, while a process that involves an MLM and generations that are possibly label-changing might be supervised by a human. Examples of templates for each linguistic rule are included in Table 3, along with candidate replacements for each token in Table 2.

**Comparative Study**

We compare architecturally different models on the three linguistic phenomena we previously introduced. We conduct an extensive evaluation on four neural architectures, namely fully connected (FC), convolutional (CNN) (Zhang, Zhao, and LeCun 2015), Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber 1997) and self-attention (Vaswani et al. 2017). We choose the number of hidden units of each layer so that the number of parameters is approximately the same and in the order of 40K\(^3\). For each

---

\(^2\)The code for full reproducibility of the experiments is available at https://github.com/EmanueleLM/the-king-is-naked. All of the experiments have been conducted on a Fedora 32 mid-end laptop equipped with 16GB of RAM and an Intel-i5 CORE 8\(^{th}\)-generation. All the neural network models have been built, trained and tested with Keras (Chollet and others 2018), while for experiments that involved BERT (Devlin et al. 2018) we relied on the PyTorch implementation (Paszke et al. 2019).

\(^3\)Each input text is 25 words long (eventually padded or cut), while each word is mapped to a vector of real numbers through a 50-dimensional embedding, pre-trained on the SST-2 task (Chollet and others 2018). Each network is composed of 3 layers, where the topology of the last two is shared, i.e., respectively a 32 hidden units ReLU and a 2 hidden units softmax layer (both are dense). The first layer depends on the specific topology under examina-
linguistic phenomenon, we analyse and compare the robustness of 20 models trained on plain SST-2 dataset (i.e., no semantic data augmentation of any kind) and then on a semantically augmented version of the same dataset (details of the augmentation are provided in the relevant subsection).

**Vanilla Models.** For each linguistic phenomenon introduced in the previous section, we analyse and compare the robustness of 20 models trained on the SST-2 dataset without augmentation. For FCs and CNNs the average accuracy on the SST-2 test set is 0.8993 ± 0.0029 and 0.9077 ± 0.0038 respectively, while the accuracies of LSTMs and self-attention are 0.9101 ± 0.0033 and 0.8963 ± 0.0015. We report in Table 1 the results of each population for the three linguistic phenomena (shallow negation, mixed sentiment and sarcasm). All the networks have been trained on the SST-2 dataset. Augmented models are vanilla models fine-tuned on the linguistic rules of interest. Symbol *, when present, means that the improved performance (from vanilla to augmented, or the other way round) is statistically significant. Interestingly, sarcasm is harder to learn and models fine-tuned on this phenomenon perform as well as their vanilla counterparts (when not worse).

<table>
<thead>
<tr>
<th>Phenomena</th>
<th>FCs Training</th>
<th>CNNs Training</th>
<th>LSTMs Training</th>
<th>Self-attention Training</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shallow Negation</td>
<td>Vanilla</td>
<td>0.4034 ± 0.0214</td>
<td>0.4032 ± 0.0124</td>
<td>0.4771 ± 0.0143</td>
</tr>
<tr>
<td></td>
<td>Augmented</td>
<td>0.4062 ± 0.0167</td>
<td>0.4249 ± 0.0255</td>
<td>0.6387 ± 0.0387*</td>
</tr>
<tr>
<td>Mixed Sentiment</td>
<td>Vanilla</td>
<td>0.4707 ± 0.0360</td>
<td>0.4986 ± 0.0415</td>
<td>0.5110 ± 0.0251</td>
</tr>
<tr>
<td></td>
<td>Augmented</td>
<td>0.4912 ± 0.0339</td>
<td>0.5271 ± 0.0387</td>
<td>0.6357 ± 0.0317*</td>
</tr>
<tr>
<td>Sarcasm</td>
<td>Vanilla</td>
<td>0.5136 ± 0.0504</td>
<td>0.4681 ± 0.0327</td>
<td>0.5578 ± 0.0128*</td>
</tr>
<tr>
<td></td>
<td>Augmented</td>
<td>0.5297 ± 0.0657</td>
<td>0.4678 ± 0.0317</td>
<td>0.4807 ± 0.0197</td>
</tr>
</tbody>
</table>

Table 1: Comparison of accuracy of vanilla and augmented models obtained for four different architectures (FCs, CNNs, LSTMs and self-attention), on three linguistic phenomena (shallow negation, mixed sentiment and sarcasm). All the networks have been trained on the SST-2 dataset. Augmented models are vanilla models fine-tuned on the linguistic rules of interest. Symbol *, when present, means that the improved performance (from vanilla to augmented, or the other way round) is statistically significant. Interestingly, sarcasm is harder to learn and models fine-tuned on this phenomenon perform as well as their vanilla counterparts (when not worse).

**Table 2:** Candidate perturbation sets used to generate combinations of replacements in template-based texts (Table 3).

<table>
<thead>
<tr>
<th>Tokens</th>
<th>Replacements</th>
</tr>
</thead>
<tbody>
<tr>
<td>@NEGATIVE@</td>
<td>‘bad’, ‘poor’, ‘boring’, [...]</td>
</tr>
<tr>
<td>@POSITIVE@</td>
<td>‘good’, ‘nice’, ‘fantastic’, [...]</td>
</tr>
<tr>
<td>@NAME@</td>
<td>‘Uma’, ‘Bruce’, ‘Sandra’, [...]</td>
</tr>
<tr>
<td>@SURNAME@</td>
<td>‘Thurman’, ‘Wills’, ‘Bullock’, [...]</td>
</tr>
<tr>
<td>@CATEGORY@</td>
<td>‘thriller’, ‘horror’, ‘comedy’, [...]</td>
</tr>
<tr>
<td>@BOOLFALSE@</td>
<td>‘false’, ‘wrong’, ‘incorrect’, [...]</td>
</tr>
<tr>
<td>@AUGMENT@</td>
<td>‘very’, ‘extremely’, [...]</td>
</tr>
</tbody>
</table>

Semantic robustness through data augmentation. In this section, we study how – and to what extent – data augmentation, along with architectural inductive biases, can be used to inject semantic robustness to different linguistic phenomena. We re-trained the models of the previous section by adding samples from (Barnes, Øvrelid, and Velldal 2019) that contain one of the specific rules used previously to the training set, up to a multiplicative factor to balance the large number of samples of the SST-2 dataset. While for a multiplicative factor of 500 none of the models exhibit any improvement in the semantic tasks, for a multiplicative factor of 750 we observe some improvement in LSTMs and self-attention. While the experiments suggest that FCs and CNNs cannot learn any of the three linguistic phenomena we studied, LSTMs and self-attention networks benefit from data augmentation. With reference to Table 1, both LSTMs and self-attention improve considerably on shallow negation. On mixed sentiment, augmented LSTMs substantially improve over the vanilla counterpart, while self-attention does not seem to exploit the additional information (despite a slight improvement over the vanilla case). Finally, data augmentation allows self-attention to improve significantly on sarcasm, though the same regime is detrimental for LSTMs, where the vanilla networks consistently outperform those trained on augmented data. Finally, for a multiplicative factor of 1K or superior, we observe a detrimental effect on the robustness of each model that is comparable to the vanilla SST-2 training.

**Classic Robustness is Linguistically Brittle**

We have compared robust models trained with IBP (Interval Bound Propagation) (Gowal et al. 2018) with their vanilla counterparts. For different values of $\epsilon = (0.001, 0.01)$ in the $L_{\infty}$-norm, which makes the model-to-model results easy to compare (La Malfa et al. 2020), and an embedding diameter

---

*With the SST-2 train set that accounts for approximately 112K input samples and each semantic rule that generates roughly 500 – 1000 new samples, semantic data augmentation with a multiplicative factor of 1 accounts for additionally 1K samples, etc.*
Shallow Negation

- "This @CATEGORY@ movie is not @AUGMENT@ @NEGATIVE@.
- "It is @BOOLFALSE@ that this @CATEGORY@ movie is @AUGMENT@ @POSITIVE@.

Mixed Sentiment

- "Despite @NAME@ @SURNAME@ acted well, this @CATEGORY@ movie is @AUGMENT@ @NEGATIVE@.
- "A @AUGMENT@ @NEGATIVE@ plot for a @AUGMENT@ @POSITIVE@ movie."

Sarcasm

- "Starring @NAME@ @SURNAME@ I'd prefer to die rather than watching this @CATEGORY@ movie."
- "Please throw this @AUGMENT@ long @CATEGORY@ movie into the ocean, and thank me later."

Table 3: Examples of template-based reviews, along with the ground truth label, used to generate sentences that contain the linguistic phenomena studied in the paper.

<table>
<thead>
<tr>
<th>Phenomenon</th>
<th>Accuracy (Barnes et al., 2019)</th>
<th>Accuracy (Our Benchmark)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shallow Negation</td>
<td>0.8552</td>
<td>0.7928</td>
</tr>
<tr>
<td>Mixed Sentiment</td>
<td>0.6024</td>
<td>0.6974</td>
</tr>
<tr>
<td>Sarcasm</td>
<td>0.7111</td>
<td>0.8455</td>
</tr>
</tbody>
</table>

Table 4: Summary of BERT semantic robustness on different linguistic phenomena, tested on samples from (Barnes, Øvrelid, and Velldal 2019) (left column) and from our template-based benchmark (right column). For these results, a BERT model has been fine-tuned on the SST-2 dataset.

Table 5: Examples of synthetic augmentations used to generate template-based reviews.

- Of approximately 3.17, we assess IBP-induced robustness on semantic rules. Interestingly, their performance is comparable (when not worse) to the brittle counterparts for all the linguistic phenomena we analyse, thus validating our previous Observation 2, i.e., that models robust in the classical sense have an extremely limited syntax/semantic manipulation capability. Results are reported in Table 5.

Accuracy is a Red Herring: the BERT Case

We analyse the relationship between semantic robustness and accuracy of a Masked Language Model (MLM): while it is known that MLMs have an improved accuracy on out-of-distribution (OOD) data (Hendrycks et al. 2020), there is no clear agreement on the nature of the semantic phenomena, i.e., whether they are linguistic outliers or OODs. Although in deep learning a trade off has been observed between the classical notions of robustness and accuracy (Tsipras et al. 2018), semantic robustness does not seem to exacerbate this phenomenon. We fine-tuned the BERT language model (Devlin et al. 2018) on the SST-2 dataset and tested its robustness on the linguistic phenomena we introduced in the previous section.

Despite an accuracy of 0.90, which is in line with the accuracy of the (simpler) architectures we tested previously, BERT’s semantic robustness is considerably higher than the "shallow" counterparts (BERT has 16 hidden layers, the models in our benchmark 3). BERT has an accuracy of 0.7928 on shallow negation, 0.6974 on mixed sentiment, and 0.8445 on sarcasm. The linguistic phenomenon where BERT performs worst is mixed sentiment, as: (i) a few recent works point out the limitations of MLM models such as BERT when learning complex syntactic/semantic constructs (Sinha et al. 2021); (ii) we have shown in our previous evaluation that self-attention (along with any other model) is especially brittle to that linguistic construct, despite the layer’s name suggesting the opposite. In general, we interpret this linguistic performance as a result of the huge amount of unsupervised training (i.e., the masked language prediction) to which BERT is subjected before being fine-tuned on our supervised task: in this sense, the phase of pre-training, which shapes the dynamics of BERT’s contextual embeddings, enables it to considerably outperform shallow models on the linguistic phenomena.

We finally validate the results of (Barnes, Øvrelid, and Velldal 2019), proving that on their challenging dataset, which contains texts from other non-movie-review datasets (so certainly out of distribution samples), BERT has an accuracy of 0.8552, 0.6024 and 0.7111 on respectively shallow negation, mixed sentiment and sarcasm. This therefore justifies that the task that we set up with our synthetic augmentation through templates is a solid alternative benchmark for semantic robustness. We summarize the results in Table 4.

Ablation Study of BERT. We performed an ablation study of BERT to assess the role of the stacked embeddings to semantic robustness. We hence trained different semantic classifiers on top of a decreasing number of BERT embedding layers. We then measured the semantic robustness on shallow negation, mixed sentiment and sarcasm on samples from (Barnes, Øvrelid, and Velldal 2019): we found that, despite the accuracy on the task (SST-2) being strongly correlated with the depth of the BERT embedding, semantic robustness is not, as depicted in Figure 3. While the best performing layer is the penultimate? (an interesting phenomenon that is already known in the literature (Rogers, Kovaleva, and Rumshisky 2020)), we could not find a layer that performed the best on all the tasks, a result that leads us to conclude that stacked attention embeddings are fundamental but their internal representation w.r.t. linguistic phenomena (i.e., the 'semantics of BERT') is still poorly understood. To complement the analysis, we tried to disentangle the role of pre-training from that of the embedding depth and attention (which are considered in the design of each BERT hidden layer) by training a very deep LSTM, with 100 input words.
against conducted on 30 networks trained to be semantically robust.

In Figure 4 (top), we show average accuracy of 30 trained FC models on the SST-2 dataset, compared for different values of $\epsilon$-robustness (w.r.t. the $L_\infty$-norm). For $\epsilon$ equal to 0., a model is not robustly trained, and otherwise it is through IBP (Gowal et al. 2018). There is clear a trade-off between robustness and accuracy. On the bottom plot, the average norm of the models’ parameters indicates that robust models tend to have lower variance, and hence arguably lower complexity.

Robustness Induced Biases

In this section we examine the relationship between common inductive biases that have inspired the design of machine learning algorithms for the past decades (Mitchell 1980), and recently also neural networks (Kharitonov and Chaabouni 2020), connecting them to the notions of robustness we dissected in the previous section. In particular, we compare local continuous to local semantic robustness.

Minimum Cross-validation Error. There is empirical evidence in the literature (Huang et al. 2019a; Jia et al. 2019) that continuous robustness does not naturally induce better performance on trained models. Indeed, most of the models that are trained to be robust are less accurate than the brittle counterparts. This side-effect is caused by the margin that is propagated through the network to the output to induce invariance to nearest neighbours of a given input. “Shielding” the model with a thick margin of possibly unrelated terms leads to an inconsistent treatment of different sentences (as noted in Observation 2, human language abounds in edge cases). This is testified by further experiments shown in Figure 4 (top). Concerning semantic robustness, generalization on cogent linguistic rules does not necessarily benefit a model’s performance, as demonstrated by experiments we conducted on 30 networks trained to be semantically robust against shallow negation vs. their vanilla counterpart: both populations have been trained on the SST-2 dataset (Socher et al. 2013). Robustness is enabled through simple data augmentation on the dataset provided by Barnes et al. (Barnes, Øvrelid, and Velldal 2019), whereas the test is performed on unseen sentences that exhibit the same linguistic phenomenon. While the vanilla networks have an average accuracy of 0.9036 ± 0.0019 on the test set and 0.4916 ± 0.0074 on the shallow negation test set, those that have been robustly trained have an accuracy of 0.8838 ± 0.0049 and 0.5491 ± 0.0124, respectively.

Minimum Description Length. Local continuous robustness is known to be a strong regularizer (Gowal et al. 2018). In fact, classical methods used to induce local robustness for NLP (such as IBP), which propagate through all the embedding dimensions and thus amplify the noise, nonetheless play an important role as they smooth out the network’s hidden activations. We report the results of experiments that we conducted that support this hypothesis in Figure 4 (bottom). As regards semantic robustness, we cannot conclude anything definitive but the evidence suggests that semantically robust models are not necessarily smoother than the vanilla counterparts. We compared the weights’ norm of 30 networks trained to be robust against shallow negation vs. their vanilla counterparts (see previous paragraph for details). While the difference between the performance of the two networks on unseen texts that contain that linguistic phenomenon is substantial, there is very little difference in the norm of the two populations, which are respectively 0.0017 ± 0.0019 (vanilla) and 0.0064 ± 0.0032 (robust).

Nearest Neighbours. Local robustness induces a strong bias towards nearest neighbours, by definition. This assumption is critical as robust training underestimates the effect of making a model robust, treating all the dimen-
sions in the embedding as equally important. We hypothesize this causes the deterioration of the performance of robust models in NLP. The induced invariance along any dimension reduces the effectiveness of the embedding representation on cogent syntactic/semantic tasks such as word-sense-disambiguation, polysemy, etc. **Semantic robustness** takes a different approach and is expected not to be robust to nearest neighbours in the embedding space, but rather to perturbations that are generated by the linguistic rules for which they have been robustly trained. For an increasing number of embedding dimensions, semantic robustness does not suffer in principle from the trade-off between the performance on linguistic tasks (Chen et al. 2013) and robustness guarantees (La Malfa et al. 2020).

### Conclusions

In this paper we formalise the concept of semantic robustness, which generalizes the notion of NLP robustness by explicitly considering the measurement of robustness on cogent linguistic phenomena. We propose a template-based generative test bed to evaluate semantic robustness. We conduct an empirical analysis that demonstrates that, despite being harder to implement, semantic robustness provides stronger guarantees for complex linguistic phenomena where models robust in the classical sense fail. In future, we aim to automate, when possible, the generation of semantic test beds, aided by powerful Masked Language Models such as GPT. We further plan to introduce a validation step for the newly generated texts by involving humans to assess the quality of the semantic perturbations (and consequently of the newly generated texts by involving humans to assess the quality of the semantic perturbations (and consequently of the newly generated texts by involving humans to assess the quality of the semantic perturbations (and consequently of the newly generated texts). Finally, we aim to extend our analysis to high-order NLP tasks and study the relationship of linguistic phenomena with out-of-distribution and outlier samples.

### Acknowledgements

This project has received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (FUN2MODEL, grant agreement No. 834115).

### References


Brown, T. B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell,


