Machine Reading, Fast and Slow: When Do Models “Understand” Language?

Sagnik Ray Choudhury¹*, ², Anna Rogers², and Isabelle Augenstein²

¹University of Michigan
²University of Copenhagen
sagnikrayc@gmail.com, arogers@sodas.ku.dk, augenstein@di.ku.dk

Abstract

Two of the most fundamental challenges in Natural Language Understanding (NLU) at present are: (a) how to establish whether deep learning-based models score highly on NLU benchmarks for the ‘right’ reasons; and (b) to understand what those reasons would even be. We investigate the behavior of reading comprehension models with respect to two linguistic ‘skills’: coreference resolution and comparison. We propose a definition for the reasoning steps expected from a system that would be ‘reading slowly’, and compare that with the behavior of five models of the BERT family of various sizes, observed through saliency scores and counterfactual explanations. We find that for comparison (but not coreference) the systems based on larger encoders are more likely to rely on the ‘right’ information, but even they struggle with generalization, suggesting that they still learn specific lexical patterns rather than the general principles of comparison.

1 Introduction

Generally, human decisions may be based on deliberate, careful reasoning (‘slow thinking’) or quick heuristics (‘fast thinking’) (Kahneman, 2011). These two processes have parallels in the realm of reading comprehension (RC): a human reader would ideally fully process the text to answer questions, but in practice, we may deliberately skim rather than read to save effort. Even capable students may be misled by superficial cues (Ackerman et al., 2013).

The previous generations of NLP models have already achieved high performance on many RC benchmarks, but they were found to often ‘read fast’, i.e. rely on shallow patterns (Chen et al., 2016; Jia and Liang, 2017; Rychalska et al., 2018). Fine-tuned Transformer-based models (Devlin et al., 2019) still have similar shortcomings (Sugawara et al., 2020; Rogers et al., 2020; Sen and Saffari, 2020; Kassner and Schütze, 2020, inter alia) in RC, as well as other tasks (McCoy et al., 2019; Jin et al., 2020).

Consider the example in Figure 1. A human reader would ideally construct the coreference chain resolving the pronoun ‘he’ to ‘Leo Strauss’. A possible heuristic-based solution is entity type matching (Jia and Liang, 2017): a model could observe that a ‘where’ question can only be answered by a ‘location’ and among two such entities (‘Germany’ and ‘United States’) the correct answer (‘Germany’) is closer to ‘born’. Such heuristic reasoners will not generalize to unseen examples. Thus a key challenge in building trustworthy and explainable RC systems is to make sure their decisions are based on valid reasoning steps. However, it is difficult to establish: (a) what that reasoning should be; and (b) whether a blackbox system adheres to it.

The present study proposes a framework for the analysis of RC models that includes: (a) defining the expected reasoning; (b) analysing model performance using explainability techniques. In particular, we contribute a case study for RC questions involving coreference resolution and comparison: we define the expected ‘reasoning’ for them (§2) and use a combination of saliency-based and coun-
terfactual explanations (§3) to analyze RC systems based on BERT and RoBERTa encoders of various sizes (§4). Overall, we find that the larger models are more likely to rely on the ‘right’ information, but even they seem to learn specific lexical patterns rather than underlying linguistic phenomena.

2 When do RC Model ‘Understand’ A Text?

2.1 Understanding in Humans

The phenomenon of ‘natural language understanding’ is not yet sufficiently well defined even for human speakers, although it is pursued by at least three different fields: philosophy of mind (e.g. Grimm, 2021; Dellsén, 2020), psychology (e.g. Christianson, 2016; Zwaan, 2016), and pedagogy (e.g. Lander, 2010; Duffin and Simpson, 2000). We cannot do this topic justice within the scope of this paper, but let us briefly outline the key premises about human understanding that we rely on in our work:

- Understanding is not truth-connected: it is “a merely psychological state” (Grimm, 2012);
- Its objects are something like ‘connections’ or ‘relations’ of the phenomenon X to other phenomena (Grimm, 2021);
- It is not binary: teachers routinely talk of ‘levels of understanding’, ‘continuum of understanding’ or ‘partial understanding’ (Nurhuda et al., 2017);
- It is different from ‘knowledge’, i.a. since it is “not transmissible” in the same sense as knowledge is” (Burnyeat and Barnes, 1980).

If human understanding is about establishing connections between new and existing conceptualizations, its success depends on the pre-existence of a suitable set of conceptualizations, to which the connections can be established (this is why e.g. algebra is taught in schools before differential calculus). The set of conceptualizations that each of us possesses is unique, since it depends on our experience of the world (cf. Fillmore’s ‘semantics of understanding’ (Fillmore, 1985)). This, together with other factors like level of motivation, attention etc., explains the variation in human understanding: we may grasp different sets of possible connections between different aspects of the new phenomenon and our pre-existing worldview.

2.2 ‘Understanding’ in Machines

Much research on human understanding focuses on mechanisms that fundamentally do not apply to current NLP systems, such as the distinction between ‘knowledge’ and ‘understanding’ or the fact that humans will fail to understand if they don’t have suitable pre-existing conceptualizations (while an encoder will encode text even if its weights are random). Since the mechanism (and its results) is so fundamentally different, terms like ‘natural language understanding’ or ‘reading comprehension’ for the current NLP systems are arguably misleading. It would be more accurate to talk instead of ‘natural language processing’ and ‘information retrieval’.

While terms like ‘understanding’ are widely (mis)applied to models in AI research (Mitchell, 2021), their definitions are scarce. Turing famously posited that the question “can machines think?” is too ill-defined to deserve serious consideration, and replaced it with a behavioral test (conversation with a human judge) for when we would say that thinking occurs (Turing, 1950). Conceptually, this is still the idea underlying the ‘NLU’ benchmarks used today: we assume that for models to perform well on collections of tests such as GLUE (Wang et al., 2018, 2019), some capacity for language understanding is required, and hence if our systems get increasingly higher scores on such behavioral tests, this would mean progress on ‘NLU’. However, just like the Turing test itself turned out to be “highly gameable” (Marcus et al., 2016), so are our tests3 (Sugawara et al., 2020; Rogers et al., 2020; Sen and Saffari, 2020; Kassner and Schütze, 2020; McCoy et al., 2019; Jin et al., 2020, inter alia).

All this suggests that, at the very least, we need a better specification for the success criteria for such behavioral tests. Instead of asking “Does my RC model “understand” language?” we could ask: “Does my RC model produce its output based

---

1This is why, as any teacher knows from practice, simply presenting the students with definitions or principles does not necessarily result in understanding of those principles or definitions.

2Marcus and Davis (2019) dispute even the applicability of the term “reading”, declaring the current QA/RC systems “functionally illiterate” since they cannot draw the implicit inferences crucial for human reading.

3In fact, the larger the dataset, the more of likely spurious patterns are to occur (Gardner et al., 2021). This presents a fundamental problem for data-hungry deep learning systems: “the models, unable to discern the intentions of the data set’s designers, happily recapitulate any statistical patterns they find in the training data” (Linzen, 2020).
the ability to combine information in multi-step reasoning, knowing what kind of information is needed and where to find it, and interpreting/manipulating linguistic input. A single question may require the competency of several types of ‘skills’.

This study contributes an empirical investigation on two RC ‘skills’ in the broad category of ‘interpreting/manipulating linguistic input’: coreference resolution and comparison. Both of them rely on the contextual information and linguistic competence. Assuming that a human reader would first read the question and then read the context in order to find the answer, they would need to perform roughly three steps: (a) to interpret the ‘question’ (akin to its transformation to a formal semantic representation or a query); (b) to identify the relevant information in the context through establishing the referential equality between expressions in the question and in the context; (c) to use that information to perform the operation of comparison or coreference resolution (see Table 1).  

2.4 Reasoning an RC Model does Perform

Having established what reasoning steps an RC model should perform, the next step would be to ascertain whether that is the case for specific models. But generally, the interpretability of DL models is an actively developed research area (Belinkov and Glass, 2019; Molnar, 2022). In this study, we rely on a combination of two popular post-hoc explanation techniques, but we also discuss their limitations, and expect that new methods could soon be developed and used in the overall paradigm for the analysis of RC models that we propose.

Attribution/saliency-based methods Li et al. (2016); Sundararajan et al. (2017) provide a saliency score for each token in the input, which shows how ‘important’ a given token is for the model decision in this instance. Figure 2 illustrates that such scores may not necessarily map onto human rationales.

To establish whether a model performs a given reasoning step (see Table 1), we define the following partition of the token space: the tokens the model should find important (positive) vs the ones it should not (negative). For example, to know if the model ‘attends’ to the entities being compared, we can define the positive partition as (blind,
The basic reasoning steps for answering comparison and coreference questions.

**Example**

**Context:** Blind Shaft is a 2003 film about a pair of brutal coal artists operating in the illegal coal mines of present day northern China. The Mask Of Fu Manchu is a 1932 pre-Code adventure film directed by Charles Brabin.

**Question:** Which film came out earlier, Blind Shaft or The Mask Of Fu Manchu?

**Answer:** The Mask Of Fu Manchu.

**Example**

**Context:** Barack Obama was the 44th president of the US. He was born in Hawaii.

**Question:** Who was born in Hawaii?

**Answer:** Barack Obama.

---

**Table 1:** The basic reasoning steps for answering comparison and coreference questions.

<table>
<thead>
<tr>
<th>Example</th>
<th>Step</th>
<th>Relevant Spans</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Comparison</strong></td>
<td>Interpreting the question</td>
<td><em>came out</em> relation: ⟨film, release date⟩ &lt;br&gt; film entities: Blind Shaft, The Mask Of Fu Manchu &lt;br&gt; earlier: date comparison &lt;br&gt; target: min(release date Blind Shaft, release date The Mask Of Fu Manchu)</td>
</tr>
<tr>
<td><strong>Comparison</strong></td>
<td>Identifying relevant information through referential equality</td>
<td>Blind Shaft_q := Blind Shaft, &lt;br&gt; The Mask Of Fu Manchu_q := The Mask Of Fu Manchu, &lt;br&gt; came out_q := ⟨date, film⟩ construction, &lt;br&gt; release dates: ⟨Blind Shaft, 2003⟩, ⟨The Mask Of Fu Manchu, 1932⟩</td>
</tr>
<tr>
<td><strong>Comparison</strong></td>
<td>Value comparison</td>
<td>solution: earlier_q := min_q &lt;br&gt; min(1932, 2003) = 1932</td>
</tr>
<tr>
<td><strong>Coreference</strong></td>
<td>Interpreting the question</td>
<td>born relation: ⟨person, location⟩ &lt;br&gt; Hawaii: location &lt;br&gt; target: born: ⟨Hawaii, UNK⟩</td>
</tr>
<tr>
<td><strong>Coreference</strong></td>
<td>Identifying relevant information through referential equality</td>
<td>Hawaii_q := Hawaii, &lt;br&gt; born relation: ⟨he, Hawaii⟩</td>
</tr>
<tr>
<td><strong>Coreference</strong></td>
<td>Coref. resolution</td>
<td>solution: born ⟨Barack Obama, Hawaii⟩</td>
</tr>
</tbody>
</table>

---

**Figure 2:** IG saliency scores example. Green/red denotes positive/negative scores.

*CLS* which film came out earlier, blind shaft or the mask of fu manchu? [SEP] blind shaft is a 2003 film about a pair of brutal con artists operating in the illegal coal mines of present day northern china. the mask of fu manchu is a 1932 pre-code adventure film directed by charles bra ##bin. [SEP]

---

The basic reasoning steps for answering comparison and coreference questions.

**Counterfactual explanations** have the form: “had X not occurred, Y would not have occurred” (Molnar, 2022). In NLP, they are based on input perturbations (Kaushik et al., 2020; Gardner et al., 2020; Sen et al., 2021; Atanasova et al., 2022b). In our case, it translates to “had the model not relied on information X, it could not have answered both the original and the perturbed instance correctly”. Thus the perturbation has to change the correct label, unlike for contrast sets (Gardner et al., 2020).

Counterfactual (CF) explanations are considered to be more faithful, since they identify input features that impact predictions. However, they typically have to be manually generated (Kaushik et al., 2020), which makes large-scale CF generation prohibitively expensive (Khashabi et al., 2020).

We rely on both types of explanations as parallel sources of evidence about RC model reasoning, and define their alignment as follows:

**Definition 2** *Explanation Alignment.* A CF and saliency-based explanation align when: (a) both the original and the counterfactually modified instance are answered correctly; and (b) the positive partition has a statistical significantly higher average saliency score than the negative partition.

We define the alignment score as follows:

**Definition 3** *Alignment Score.* The Alignment Score for a dataset, model, reasoning step > triple is the proportion of instances in that dataset for which different kinds of explanations align (according to our Def. 2).

We interpret a high alignment score as evidence that both kinds of explanations are faithful, and the model indeed performs the expected reasoning steps.

---

i.e. there is an exact match between the predicted and the correct answer.
3 Methodology

3.1 Datasets and Models

For Coreference, we use the Quoref (Dasigi et al., 2019) dataset (20K training and 2.4K validation instances) where the annotators were asked to design questions for a given text so that answering those would require resolving anaphora. For Comparison, we sample questions from HotpotQA (Yang et al., 2018) and 2WikiMultiHopQA (Ho et al., 2020): two datasets with questions manually annotated with their reasoning type (bridge or comparison). We select the ‘comparison’ questions containing comparative adjectives or adverbs in them (23K training, 3K validation instances). These resources are based on Wikipedia and have multiple passages as contexts, but the sentences (typically 2-3) necessary to answer a question are marked as ‘supporting facts’. Since we are not focusing on the multi-hop information retrieval skill, we limit the contexts to these sentences.

We experiment with five pre-trained Transformer-based encoders of the BERT family: RoBERTalarge (Liu et al., 2019), BERTlarge-cased, BERTbase-cased (Devlin et al., 2019), BERTmedium, and BERTsmall (Turc et al., 2019; Bhargava et al., 2021). These BERT models differ mainly in the structure of architecture blocks and the number of parameters, while RoBERTa also has a different training corpus and optimization. Since larger models were shown to generalize better for some use cases (Hendrycks et al., 2020; Bhargava et al., 2021), we investigate whether they also are more likely to be right for the right reasons.

We fine-tune each encoder using the architecture in Devlin et al. (2019) (see the appendix for details) and evaluate them on the validation set (as the test sets are not public). We use the standard evaluation metrics in extractive QA: F1-Score (the percentage of token overlap between predicted and ‘gold’ answers, averaged over all data points), and Exact-match (the number of data points where the predicted answer matches the ‘gold’ answer).

3.2 Counterfactual Explanations

Our formulation of reasoning (Table 1) consists of three basic steps for both coreference and comparison: interpreting the question, identifying the relevant information through referential equality, and the target operation on the identified information (coreference resolution or value comparison). We focus on the final step, since: (a) it implicitly relies on correct semantic parsing of the question and the context; (b) referential equality in our data is in large part trivial: most entities have the same surface form in the question and the text.

An obvious semantically valid perturbation that should change the prediction (and thus test for the model’s understanding of the comparison operation) is to replace the comparative adjectives with their antonyms (Figure 3d). Since our sample only contains 6 tokens used as comparison operators, we define appropriate replacements manually.\textsuperscript{8}

For coreference questions, a competent RC model would at least resolve the coreference chain for the target entity. A context can have many coreference clusters, so we need to identify the relevant one. In the Quoref dataset, we use the instances where the relevant cluster itself contains the answer entity\textsuperscript{9} (see Figure 3a), and therefore, can be extracted automatically. This leaves us with 55% (1329/2418) of the validation instances. These are further subsampled to manually create 100 CF instances by inserting a new sentence, which includes the new and excludes the old answer (see Figure 3b). Similarly to the comparison questions, the original answer entity remains in the context. If the model uses the ‘shortcut’ of choosing the most frequent entity in the context (Wu et al., 2021), it should not be able to answer both the original and the perturbed instance correctly.

3.3 Saliency-based Explanations

We obtain token saliency scores from two families of attribution/saliency methods: Occlusion (DeYoung et al., 2020), a method based on perturbations, and Integrated Gradients (IG, Sundararajan et al., 2017), a method based on gradients.\textsuperscript{10}

**Design decisions:** RC models typically predict two scores ($t_s, t_e$) for each token $t$: the probability of $t$ being the start and the end of the answer span. Any attribution method produces two scores ($A_{start}^t, A_{end}^t$) for each token $t$, indicating how ‘im-

\textsuperscript{8}earlier↔later, first↔later, more recently↔earlier, older↔younger.

\textsuperscript{9}We extract the clusters using an off-the-shelf coreference resolver (Clark and Manning, 2016) implemented in Spacy.

\textsuperscript{10}Atanasova et al. (2020) shows that for Transformer based architectures, Occlusion is the best perturbation method by two evaluation criteria: agreement with human rationale and faithfulness. A recent paper by Ye et al. (2021) finds IG to be one of the most faithful gradient-based methods for extractive QA, only outperformed by Layerwise Attention Attribution (LAA), a method proposed in the paper itself. We leave LAA and other popular explainability methods such as LIME (Ribeiro et al., 2016) for future work.
Figure 3: Examples of CF perturbations used in this study.

(a) A coreference question from Quoref (the relevant coreference cluster tokens in green).

Context: Górecki said of the work... I had a grandfather who was in Dachau, an aunt in Auschwitz...
Question: What is the last name of the person who had an aunt at Auschwitz?
Answer: Górecki

(b) CF perturbation for 3a (added tokens are bold-faced)

Context: Blind Shaft is a 2003 film... The Mask Of Fu Manchu is a 1932 pre-Code adventure film...
Question: Which film came out earlier, Blind Shaft or The Mask Of Fu Manchu?
Answer: The Mask Of Fu Manchu

(c) A comparison question from 2WikiMultiHopQA

Comparison: Question: Which film came out more recently, Blind Shaft or The Mask Of Fu Manchu?
Coreference: Context: Barack Obama was the 44th president of the US. He was born in Hawaii.
Question: Who was born in Hawaii?

(d) CF perturbation for 3c

Context: Blind Shaft is a 2003 film... The Mask Of Fu Manchu is a 1932 pre-Code adventure film...
Question: Which film came out later, Blind Shaft or The Mask Of Fu Manchu?
Answer: Blind Shaft

Figure 4: Positive and negative partitions for saliency explanations.

important t is for predicting the start/end of the answer span. Following Kokhlikyan et al. (2020), we use $A_{start}$ in all our saliency experiments.11

For Occlusion, we calculate $A_{start}'$ by replacing t in the input with a baseline token (MASK) and measuring the change in $t_s$. DNNs map an input vector to a scalar value (loss/class probability). Gradient-based methods measure $A_{start}'$ using the gradient of the token $t_s$ w.r.t. this scalar function (we use $\text{argmax}(t_s)$). IG sums these gradient values along a linear path from a baseline to the current instance. Both Occlusion and IG need a baseline token, which for us is the MASK token.

Gradient-based methods in NLP do not produce a scalar saliency score, i.e., $A_{start}'$ is a vector because the input is an embedding matrix and not a vector. Two common ways to summarize this vector to a scalar are: (a) scalar product between the input and the gradient vector (Han et al., 2020); or (b) $l_p$ norm, where $p \in 1, 2$ (Atanasova et al., 2020). We use $l_2$ norm (see the discussion in §4.3).

Token partitions: Figure 4 shows the token partitions used for the same reasoning steps (comparison and coreference resolution) that we also target with the CF perturbations. For comparison the positive partition consists of the question token(s) expressing the comparison operation (e.g. ‘more recently’). The negative partition consists of question tokens that are not in the set of entities or values that need to be compared, or in the set of verbs (which could capture the relation between the entities and their values). For coreference resolution, the positive partition is the context tokens in the relevant coreference cluster (§3.2). The negative partition is the set of context tokens that are not in: (a) the positive partition; and (b) match the question tokens.

4 Results & Analysis

4.1 Base Model Performance

As a sanity check, we fine-tune all models on the data described in §3.1 (Table 2). For coreference, the F1-Score of our best model (RoBERTa_large) is slightly better (82.10) than the previously reported score (79.64, Wu et al. (2021)). The comparison instances are sampled from parts of two datasets, and so a direct comparison is not possible.12

The size and the model family matter: RoBERTa performs better than BERT for two models of the same size, and the larger models do better. Interestingly, the difference is more pronounced for the Quoref dataset, where the instances have longer contexts and the questions are more complex.

11We also briefly experimented with $(A_{start} + A_{end})/2$ for Occlusion but it yielded very similar saliency ranking of the tokens on a 100 sample subset of the Comparison dataset.

12The best model (RoBERTa_large) has an F1-Score of 92%, slightly better than the highest score reported on the HotpotQA leaderboard (89.14%) and much better than the baseline model for the 2WikiMultiHopQA dataset (65.02, (Ho et al., 2020)).
Table 2: Average (3 runs) results of different models on Comparison and Coreference datasets. The STD varies between 0.01 – 0.72%. Green indicates the best scores.

<table>
<thead>
<tr>
<th></th>
<th>Comparison</th>
<th>Coreference</th>
<th>Comparison</th>
<th>Coreference</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>F1</td>
<td>EM</td>
<td>F1</td>
<td>EM</td>
</tr>
<tr>
<td>RoBERTa$_{large}$</td>
<td>92.08</td>
<td>91.07</td>
<td>82.10</td>
<td>79.39</td>
</tr>
<tr>
<td>BERT$_{large}$-cased</td>
<td>89.23</td>
<td>88.57</td>
<td>71.91</td>
<td>68.47</td>
</tr>
<tr>
<td>BERT$_{base}$-cased</td>
<td>89.38</td>
<td>88.37</td>
<td>64.62</td>
<td>59.38</td>
</tr>
<tr>
<td>BERT$_{medium}$</td>
<td>86.45</td>
<td>85.96</td>
<td>60.16</td>
<td>54.82</td>
</tr>
<tr>
<td>BERT$_{small}$</td>
<td>71.44</td>
<td>69.87</td>
<td>50.94</td>
<td>43.39</td>
</tr>
</tbody>
</table>

Table 3: F1-Score for the original and the CF perturbations. Red denotes significant drop.

<table>
<thead>
<tr>
<th></th>
<th>Coreference</th>
<th>Comparison</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>og</td>
<td>cf</td>
</tr>
<tr>
<td>RoBERTa$_{large}$</td>
<td>92.0</td>
<td>70.7</td>
</tr>
<tr>
<td>BERT$_{large}$-cased</td>
<td>86.2</td>
<td>50.8</td>
</tr>
<tr>
<td>BERT$_{base}$-cased</td>
<td>82.5</td>
<td>39.2</td>
</tr>
<tr>
<td>BERT$_{medium}$</td>
<td>74.0</td>
<td>35.8</td>
</tr>
<tr>
<td>BERT$_{small}$</td>
<td>67.2</td>
<td>29.4</td>
</tr>
</tbody>
</table>

4.2 Counterfactual Explanations

Table 3 compares the F1-Score of the original (‘og’) vs counterfactual (‘cf’) instances. For the comparison questions, the performance on the original and CF instances are very close for all models except BERT$_{small}$. Bigger models consistently perform better, but in most cases the difference with the next larger model is relatively small.

For coreference questions, CF instances are much more difficult for all models. Even the best model RoBERTa$_{large}$ experiences a 24% drop. All BERT models perform poorly: even the larger ones have a 40% performance drop (BERT$_{large}$-cased). Thus, the CF tests show that the models are more likely to follow the expected reasoning strategy for comparison, but not for the coreference questions.

4.3 Alignment Score

For statistical significance testing in ‘Expectation Alignment’ (Def. 2), we use a one-tailed independent $t$-test ($p = 0.05$) with the null hypothesis that the positive partition does not have a higher average saliency score. Table 4 shows the ‘Alignment Score’ (Def. 3) results for comparison and coreference resolution (§3.3), using saliency scores from IG and Occlusion.

Ideally, for a random partition of tokens in any instance, the positive and the negative partitions should have similar saliency scores. For a dataset, they should be significantly different in $\approx 0\%$ cases. For Occlusion, the saliency scores are significantly different in only $5.6 – 8.2\%$ instances for a random partition. Recall that in §3.3 we discussed 3 summarizers for IG. Among all of them, the $l_2$ norm is the only one where this happens in $5.2 – 7.3\%$ cases, for the other two the numbers are between $11.3 – 28.9\%$.

Table 4 shows that, counter-intuitively, for both comparison and coreference questions the larger models overall have lower IG alignment scores, meaning that they do not pay as much ‘attention’ to the tokens we defined as important. This is despite the fact that for comparison the above CF experiment suggests that the models do perform the expected reasoning operations. One possible explanation is that IG simply does not reliably capture the model’s reasoning process, and Occlusion does better at that because its trend in alignment is the opposite of IG: bigger models tend to have significantly higher alignment scores. Another possible explanation is that IG explanations are in fact faithful, but, having more ‘attention’ to the tokens we defined as important is counter-productive. Consider that the BERT$_{small}$ model achieves an Exact-match of 87% on the original questions containing the comparative tokens ‘earlier’, ‘first’ and ‘older’ (which are 2.1 times more frequent in the training data than all others), and an Exact-match of 28% on the other original questions. Yet overall the model performs poorly, and thus the reliance on these highly frequent comparative adjectives could be a bug rather than a feature. As this hypothesis brings into question the overall utility of saliency-based explanations for testing for the ‘correct’ reasoning steps, we hope it will be investigated in more depth in future work.

---

$^{13}$Aggregation of local explanations such as saliency scores are not guaranteed to produce faithful global explanations (Setzu et al., 2021), but this is a convincing evidence.

$^{14}$The lack of alignment between the two techniques is consistent with the findings of Atanasova et al. (2020).
Table 5: F1-Score for the original (OG) comparison questions and their counterfactual perturbations in (CF) and out (CF-ood) of the training distribution. The models are provided either a smaller context of supporting facts or full paragraphs. Red indicates a significant drop in performance.

<table>
<thead>
<tr>
<th></th>
<th>Supporting Facts</th>
<th>Paragraphs</th>
<th>Coreference</th>
<th>SQuAD</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>OG</td>
<td>CF</td>
<td>CF-ood</td>
<td>OG</td>
</tr>
<tr>
<td>RoBERTa_{large}</td>
<td>99.4</td>
<td>98.9</td>
<td>77.2</td>
<td>98.7</td>
</tr>
<tr>
<td>BERT_{large-cased}</td>
<td>98.9</td>
<td>93.1</td>
<td>68.7</td>
<td>98.0</td>
</tr>
<tr>
<td>BERT_{base-cased}</td>
<td>98.4</td>
<td>91.8</td>
<td>58.1</td>
<td>97.0</td>
</tr>
<tr>
<td>BERT_{medium}</td>
<td>97.4</td>
<td>96.5</td>
<td>64.4</td>
<td>96.2</td>
</tr>
<tr>
<td>BERT_{small}</td>
<td>68.2</td>
<td>45.3</td>
<td>57.1</td>
<td>68.3</td>
</tr>
</tbody>
</table>

Table 6: Results for different heuristic methods on the coreference and SQuAD datasets. Green indicates the best score.

4.4 Generalization Tests

Table 3 shows that when measured with CF tests, most models do not follow the expected coreference resolution strategy, but they do so for comparison. Still, based on our success criteria (Def. 1), we cannot yet conclude that they ‘understand’ comparison. A human would be able to disassociate the logical operation of comparison from the surface realizations, i.e., they would be able to answer a question correctly with either of the surface forms ‘younger’ and ‘more junior’.

For the CF experiments reported up until this point the perturbations were in-distribution, i.e., the training data had both the original question “who is younger” and the CF “who is older”. Now we replace the comparative adjectives with antonyms that are not in the training data (see the appendix for details). We also increase the context size by using full paragraphs instead of just the sentences marked as ‘supporting facts’, to see if the models would be ‘distracted’ by more information.

Table 5 shows a considerable drop in performance for CF-ood condition for all models. The larger models generalize better: RoBERTa_{large} and BERT_{large-cased} perform 2% and 8% worse for CF questions, whereas BERT_{small} exhibits a 29% reduction. The ‘supporting facts only’ condition is overall easier than the ‘paragraphs’ condition.

4.5 Heuristics for Coreference Questions

Since the CF tests (§4.2) do not show that BERT models can cope with the altered coreference chains, we have to conclude that they do not follow the expected reasoning steps. Though given the above-chance performance they must follow some other strategy. We test the hypothesis that many of the coreference questions can be answered by simple heuristics and that the models resort to those.

Specifically, we define an unsupervised dataset-independent heuristic method consisting of two steps: sentence selection and phrase extraction.

Sentence Selection: Among all the context sentences \{c_i\}, select the one that is the ‘closest’ to the question q. We experiment with 4 options for similarity: token-overlap (number of common tokens in q and c_i), sentence encoder (cosine similarity between the sentence embeddings of q and c_i created by a sentence encoder (Reimers and Gurevych, 2019)), LCS (number of tokens in the Longest Common Subsequence between q and c_i), and position (simply taking the first sentence in the context following Ko et al. (2020)).

Phrase Extraction: We assume that the model would also learn to look for a named entity in the selected sentence. The question dictates the type of this entity (e.g. ‘where’ → location, ‘who’ → person name). The type could be determined by a simple mapping between ‘wh’ question words and entity types, but this can fail (e.g. for the question “Who won the World Cup in 2002?” the expected answer is a location, not a person). Therefore, we fine-tune a Transformer model to predict the answer type from the question.\(^\text{15}\)

Table 6 shows the best heuristic has an F1-Score of 21.5% on the coreference dataset, and 26.68% on SQuAD (Rajpurkar et al., 2016), which we use for validation. The SQuAD score is comparable to the previously reported result of 26.7% in Sen and Saffari (2020) for an algorithm predicting entity types heuristically, and choosing the entity from the whole context instead of the best possible sentence. Ray Choudhury et al. (2022) uses a similar approach to find Quoref questions that can be answered heuristically, but our algorithm has

\(^{15}\)The accuracy for this model is 85.7%. See the appendix for results from multiple models and loss functions.
more sentence selection strategies, and unlike ours, Ray Choudhury et al. (2022) only uses one loss function in the phrase extraction model.

Nevertheless, the best heuristic algorithm performs considerably worse than the smallest BERTsmall model (51%, Table 2). Performance alone cannot reveal whether this strategy is used in the instances where it would be sufficient, but this result shows that even the smaller models must either rely on a more successful (but still imperfect) strategy, or at least rely on more than one heuristic. The problem with discovering potential ‘shortcuts’ in low-performing models is complicated as these strategies are not necessarily human-interpretable: González et al. (2021) show that humans struggle to predict the answer chosen by poorly performing RC models, even when the saliency explanations for that answer is shown, because these answers simply do not align with human RC strategies.

5 Discussion and Related Work

Our work continues the emerging trend of research on being ‘right for the right reasons’ (McCoy et al., 2019; Chen and Durrett, 2019; Min et al., 2019; Atanasova et al., 2022b, inter alia). We contribute stricter success criteria for behavioral tests of NLP models (Def. 1), and, for the RC task, develop the methodology of: (a) defining what information the model should rely on for a given linguistic, logical, or world knowledge ‘skill’; (b) systematically testing the behavior of RC models with interpretability techniques for whether they rely on that information. This is most closely related to the work on ‘defining comprehension’ by Dunietz et al. (2020), though their testing is limited to probing the models with RC questions. Another related study is the QED framework (Lamm et al., 2021), annotating Natural Questions (Kwiatkowski et al., 2019) with ‘explanations’ of the expected reasoning process. Their expected reasoning process also contains 3 steps, partly similar to ours: selecting a relevant sentence, referential equality, and deciding on whether this sentence entails the predicate in the question. However, the goals of QED are to: (a) predict both the answer and the explanation for a question; and (b) understand if explanations help QA models. Such explanation annotations are unavailable for most datasets, and few QA models produce explanations. Therefore, our approach of: (a) defining expected reasoning steps; and (b) using model interpretations to validate such steps applies to a broader class of models.

This study is also related to the overall efforts to define what kinds of ‘skills’ RC models can be expected to exhibit (Sugawara et al., 2018; Schlegel et al., 2020; Rogers et al., 2022). While these works focus on the high-level taxonomies of ‘skills’, we contribute practical definitions for two linguistic ‘skills’ (comparison and coreference resolution) which could be used for analyzing model performance. Implicitly, research proposing RC resources that target various specific ‘skills’ (e.g. TempQuestions (Jia et al., 2018) for temporal order, MathQA (Amini et al., 2019) for numerical reasoning, etc.) also contributes to this area, but they typically rely on broad linguistic definitions rather than on steps for machine reasoning.

The saliency techniques we rely on have previously been used for extractive QA (Madsen et al., 2021), but we are among the first (Ye et al., 2021) to investigate their correlation with counterfactual explanations. For counterfactual perturbations, we also ensure that the perturbations are human-interpretable and change the prediction, which is not the case for adding incomprehensible text (Kaushik and Lipton, 2018), removing words from questions, shuffling the context (Sen and Saffari, 2020), or replacing context tokens with random tokens (Sugawara et al., 2020).

6 Conclusion

Making progress towards trustworthy NLP models requires specific definitions for the behavior expected of these models in different situations. We propose a framework for RC model analysis that involves: (a) the definition of the expected ‘reasoning’ steps; (b) analysis of model behavior. We contribute such definitions for two linguistic ‘skills’ (comparison and coreference resolution), and use parallel explainability techniques to investigate whether RC models based on BERT family encoders answer such questions correctly for the right reasons. We find that to be the case for comparison, but not for coreference. Moreover, we find that, even for comparison, the models ‘break’ when encountering out-of-distribution counterfactual perturbations, suggesting that they memorize specific lexical patterns rather than learn more general reasoning ‘skills’. As such, more research is needed on developing definitions and tests for specific ‘skills’ expected of NLU models, as well as on more faithful interpretability techniques.
7 Acknowledgements

We would like to thank Pepa Atanasova, Gary Marcus, Mark Steedman, and Bonnie Webber for the discussion of various aspects of this work. We also thank the anonymous reviewers for their time and insightful comments.

References


A Appendix

A.1 QA Model Training

For training the QA models in §3.1 The questions and contexts are concatenated, and a linear layer on top of the encoder is used to predict the probability of a context token $i$ being the start ($P_{i,s}$) or end ($P_{i,e}$) of an answer. The score ($S_{i,j}$) for a span with start token $i$ and end token $j$ is computed as $P_{i,s} + P_{j,e}$. For all valid combination of $i$ and $j$, the span with the highest score is chosen as the answer. A cross entropy loss between the actual and predicted start/end positions is minimized.

The models were trained for 10 epochs with a batch size of 16 using the Adam optimizer (Kingma and Ba, 2015) ($\beta_1 = 0.9, \beta_2 = 0.99, \epsilon = 1e-8$, weight_decay = 0.01) and gradient clipping. The learning rate (LR) was kept at $1e-05$ with a linear warm-up schedule (staring LR=0). The models were evaluated on a subset of the validation data every 500 mini-batches with early stopping on 100 evaluations (Pruksachatkun et al., 2020). The LR and batch size was determined by a small grid search on the coreference dataset: LR=$\{1e-05, 1e-04, 1e-03\}$, batch size = $\{8, 16, 32\}$.

A.2 Antonym Replacements for CF Generation

The antonym replacements for the generalization test (§4.4) are described below:

- first $\rightarrow$ less recently
- older $\rightarrow$ less old, more junior, less mature, less grown-up
- earlier $\rightarrow$ subsequently, thereafter, less recently
- later $\rightarrow$ less recently
- younger $\rightarrow$ more old, less junior, more mature, more grown-up
- more recently $\rightarrow$ less recently, longer ago

A.3 Supervised Entity Type Predictor

Our goal is to build a classifier to predict the answer entity type from the question (§4.5). A sample data point is shown in Figure 5. The entity types are defined in the Ontonotes-5 dataset (Pradhan et al., 2013). The answer entity type is detected from the context using an off-the-shelf entity detector implemented in Spacy.\footnote{\url{https://spacy.io}} When the answer is not a named entity, or the entity detector fails to determine its type, that question is discarded.

The classification models are trained on the training portion of Quoref and SQuAD which is further divided into train/dev/test (70/20/10) split for training and evaluation. The distribution of the class labels is very skewed.

**Models:** We use two types of models: 1) a fine-tuned 12 layer 768 dimensional BERT\textsubscript{base-cased} model; and 2) a popular word convolutional model for sentence classification (Kim, 2014) using three parallel filters (size 3, 4, and 5) and 300 dimensional Google News Word2Vec representations (Mikolov et al., 2013).

BERT model: This model is trained for 5 epochs, with Adam optimizer (Kingma and Ba, 2015) with a weight decay of 1.0e-08 and a learning rate of 1.0e-05. The sequence max length is kept at 128. We search for two hyper-parameters: 1) number of epochs: 3-7, increasing by 1; and 2) learning rate: 1.0e-05, 5.0e-05, 1.0e-04.

WordConv model: This model is trained for 40 epochs, with Adadelta optimizer (Zeiler, 2012) with a learning rate of 1.0e-05. The sequence max length is again kept at 128.

For both models, accuracy was used as the early stopping metric. We minimized the cross entropy (CE) loss in general, but for the WordConv model, a weighted CE loss was also implemented to account for the training data class-imbalance in Quoref. That did not improve the results significantly and was not used in the BERT\textsubscript{base-cased} model. Table 7 shows the detailed results. Finally, we choose the fine-tuned BERT\textsubscript{base-cased} model as the entity detector as it performs the best. Ray Choudhury et al. (2022) also proposes a model to determine the answer entity type from a question, but the major difference is the label space. The model in Ray Choudhury et al. (2022) is trained to predict a label of “UNKNOWN_ENTITY” when the an-

---

Figure 5: A sample instance for answer entity type classifier.

Text: What is the full name of the person who is the television reporter that brings in a priest versed in Catholic exorcism rites?
Label: PER
Table 7: Models for supervised entity type selection. Green indicates the best results.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Model</th>
<th>Accuracy</th>
<th>Macro F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>SQuAD</td>
<td>BERT\textsubscript{base-cased}</td>
<td>76.4</td>
<td>56.2</td>
</tr>
<tr>
<td></td>
<td>WordConv</td>
<td>72.4</td>
<td>44.9</td>
</tr>
<tr>
<td>Coref</td>
<td>BERT\textsubscript{base-cased}</td>
<td>85.7</td>
<td>73.9</td>
</tr>
<tr>
<td></td>
<td>WordConv</td>
<td>85.0</td>
<td>67.6</td>
</tr>
<tr>
<td></td>
<td>WordConv</td>
<td>85.3</td>
<td>69.7</td>
</tr>
<tr>
<td></td>
<td>Weighted BCE</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

swar span is a) not a named entity or b) the entity detector can not find its type. However, an “UN- KNOWN\_ENTITY” label does not help the final algorithm (heuristic answer selection) to find the correct answer span. Therefore, our model never predicts this label, and consequently, has a better accuracy than Ray Choudhury et al. (2022). It potentially makes a mistake on the test data points that fall in the previous two categories, but the final algorithm is no worse than Ray Choudhury et al. (2022).