Examining robustness of entity information processing in pre-trained language models

Anonymous NAACL-HLT 2021 submission

Abstract

Pre-trained LMs have shown impressive performance on downstream NLP tasks, but we have yet to get a clear understanding of how sophisticated these models are when it comes to processing, retaining, and applying information in text. Here we describe experiments testing models' handling of entity information, and robustness of these processes to contextual interference. We develop a controlled dataset and apply it to popular pre-trained LMs, finding that model accuracies on this task vary widely, but all models are susceptible to interference of "attractor" words in the context. Certain models, however, stand out in showing stronger impact of entity attribute information beyond superficial contextual cues.

1 Introduction

Pre-trained language models have burst onto the NLP scene with impressive performance across a wide variety of tasks. This has inspired growing interest in understanding what knowledge and linguistic capacities are conferred on these models during pre-training, with work showing these models to capture non-trivial syntactic information (Goldberg, 2019), world knowledge (Petroni et al., 2019; Jiang et al., 2020), and more.

In this paper we target and analyze pre-trained LMs' ability to process, retain, and use information described in text. We focus on the problem of linking an attribute to an entity, with this attribute information to be used subsequently in guiding a prediction about that entity. The capacity of models to encode and deploy information in this way is a fundamental component of what we might call "language understanding", and we are interested in particular in disentangling intelligent handling of this kind of textual information, from reliance on more superficial predictive cues. Artifacts and heuristics have been shown to inflate model performance (Kaushik and Lipton, 2018; Gururangan et al., 2018; McCoy et al., 2019), and in general, the task of assessing models' capacity for "understanding" is plagued by the challenge of disentangling sophisticated information processing from more superficial predictive capabilities. This is a challenge that we tackle here.

To do this, we design a dataset of contexts describing attributes of an entity (or entities), followed by a cloze-style statement about one entity, requiring models to reactivate information about the relevant entity and attribute. Within this context, we examine the robustness of models' handling of entity information, by introducing "attractor" elements, which are not relevant for determining the correct prediction, but which are likely to distract the model and impact predictions nonetheless. We also introduce additional distance manipulations between the key entity and the key attribute. Using the resulting dataset, we examine the impacts of attractors on models' ability to prefer the best completion over competitor completions. Then we take a closer look at the relationship between superficial contextual variables and model completion probabilities, and the role played by key entity-attribute information relative to these superficial cues.

Our contributions here are two-fold. First, we introduce a dataset for assessing a) models' handling of entity information, and b) sensitivity of models' predictions to different contextual variables. Second, we use this dataset to analyze behaviors of a number of pre-trained LMs. We find that the tested models vary dramatically in accuracy on this task, but all models are susceptible to interference by attractors. Model probabilities also appear to be more sensitive to distance in single-entity contexts. Finally, certain models show robust impact of the contextual variable associated with sophisticated entity information, while others show minimal impact of sophisticated entity attribution beyond use of superficial cues. We will make all code and datasets available for extensions to this work.

Base context	
Zero attractor	**Sebastian** works as a **florist**. For his job, Sebastian sells ____
Multiple-entity attractor setting	
One attractor	**Sebastian** works as a **florist**. Rowan works as a baker. For his job, Sebastian sells ____
Two attractors	**Sebastian** works as a **fisherman**. Rowan works as a baker. Daniel works as a butcher. For his job, Sebastian sells ____
One pre-$p*$ attribute and one attractor	**Sebastian** went to school in France and now works as a **painter**. Rowan works as a baker. For his job, Sebastian sells ____
Single-entity attractor setting	
One attractor	**Sebastian** works as a **baker**, and likes to eat meat. For his job, Sebastian sells ____
Two attractors	**Sebastian** works as a **carpenter**, and likes to eat meat and look at paintings. For his job, Sebastian sells ____
One pre-$p*$ attribute and one attractor	**Sebastian** went to school in France and now works as an **optician**, and likes to eat meat. For his job, Sebastian sells ____

Table 1: Example items from the entity profession synthetic dataset. Includes illustration of both multiple-entity and single-entity attractor settings, and addition of pre-$p*$ attributes.

2 Related Work

In focusing on models' ability to extract, retain, and deploy information conveyed in text, our work relates clearly to tasks in reading comprehension question answering (Rajpurkar et al., 2018; Kočiskỳ et al., 2018; Mostafazadeh et al., 2017; Yang et al., 2018; Richardson et al., 2013). We complement such tasks with a more precise and fine-grained picture of models' information processing, through use of controlled, synthetic data—both to enable focus on specific types of textual information, and to gauge impacts of different types of contextual variables on models' predictions.

A good deal of prior work has focused on investigating the linguistic knowledge in language models. Much of this work has focused on syntactic sensitivities in pre-trained LMs via agreement tests (Linzen et al., 2016; Gulordava et al., 2018). Others expand to broader sets of syntactic phenomena (Wilcox et al., 2018; Futrell et al., 2019; Warstadt et al., 2020). Other work has studied syntactic and semantic information in contextualized embeddings produced by these models (Hewitt and Manning, 2019; Tenney et al., 2018; Klafka and Ettinger, 2020). We take one step up from examination of these abstract linguistic capacities, with a focused examination of models' ability to use such linguistic information to process and recall attributes of entities being described in text.

A number of papers have also used cloze-style probes to examine higher-level capacities of pre-trained language models, such as world knowledge (Petroni et al., 2019; Jiang et al., 2020) and pragmatic/commonsense reasoning (Ettinger, 2020). We complement this previous work in focusing primarily on models' sensitivity to entity information provided directly in the text, and in applying various contextual manipulations to examine model robustness in the face of contextual variation.

Our use of attractors most closely mirrors Linzen et al. (2016), but we differ from that work in focusing on models' processing of entity information rather than syntactic dependencies. Some work has also introduced *primes* in context to examine impact on model predictions, experimenting with contextual factors like distance between prime and target (Kassner and Schütze, 2020), and contextual constraint (Misra et al., 2020). We build on this work in examining specifically how contextual interference impacts models' ability to retain and apply entity information from the text itself, and in executing a more systematic and comprehensive exploration of the impacts of contextual cues.

3 Entity profession dataset

To study models' information retention and robustness to contextual interference, we design a dataset of synthetically-generated cloze-style items. We fo-

cus on the problem of associating a key entity with a profession, and ask models to make predictions based on a property of that entity's profession. In total the dataset has 54,768 cloze-style contexts.

3.1 Base context

All contexts in the dataset are in English, and derived from the following base context:

Sebastian works as a <profession>. For his job, Sebastian sells ____

Each context requires a profession and a corresponding high-probability completion word. We select a small set of professions that involve selling of distinctive objects, ensuring that for all selected professions, the corresponding objects are high-probability completions in the base context for all of our tested models.[1] This results in a set of seven profession-object pairs that we use for our dataset.[2]

We will use the following terms to describe context components. Each context contains a key protagonist entity $e*$ and a profession $p*$ attributed to that entity.[3] The profession $p*$ is chosen from the set P of professions, and each profession $p_i \in P$ is associated with a corresponding completion object $o_i \in O$. Each context contains a target position t_{pred}, for which the ideal completion is object $o*$ associated with protagonist profession $p*$.

While the association of $p*$ with $o*$ involves world knowledge, we are interested in models' ability to attribute $p*$ to $e*$, and to reactivate this attribution at the second mention of $e*$, to predict $o*$. So we establish a baseline presence of the requisite world knowledge by ensuring that for all tested models, each o_i is among the top predictions in the base context with p_i. This allows us to set the world knowledge component aside, and to focus on the effects of the key contextual manipulations on models' handling of entity information.

3.2 Inserting attractors

We insert attractors between $p*$ and t_{pred}, to test models' robustness to mention of other professions/ profession-related-objects that are not relevant for identifying the best completion at t_{pred}. In inserting attractors, we take two different approaches.

Multiple entity In the first approach, which we call the *multiple entity* setting, attractors consist of sentences describing other entities and their professions. The middle segment of Table 1 shows examples in this setting. The maximum number of attractors that we use in this setting is six.

Single entity The second approach is the *single entity* setting. In this setting, attractors consist of other objects $o_i \in O$, such that $o_i \neq o*$, mentioned as additional attributes of the key entity $e*$. The bottom segment of Table 1 shows examples in this setting. The maximum number of attractors that we use in this setting is five.

3.3 Varying protagonist/profession distance

Since a key requirement in this task is for models to associate the protagonist entity with the correct profession, in both attractor settings we also introduce material of varying length between key entity $e*$ and key profession $p*$. This intervening material consists of other attributes of $e*$ (e.g, "went to school in France", "played basketball", and "sang in a choir"), but does not contain any attractors related to competing o_i completions. We refer to these as "pre-$p*$ facts" as they are facts about $e*$ that precede the mention of $p*$. Table 1 also illustrates insertion of these pre-$p*$ facts.

3.4 Contextual annotations

In addition to testing how attractors impact models' robustness in predicting the correct completion $o*$, we also want to better understand the impact of different contextual variables on completion probabilities in general. To facilitate this investigation, our dataset includes contextual annotations for each object $o_i \in O^C$, where O^C is the set of all objects with associated mentions in the context. (For instance, in the context "Sebastian works as a florist. Rowan works as a painter. For his job, Sebastian sells ____", we have contextual annotations for *flowers* and *paintings*). These contextual annotations are completion-specific in that they involve factors like distance and strength of association between a candidate completion and its associated mention in the context. Since these associated mentions may be professions or objects, and may be $p*$ or attractors, we simply denote as m_i the contex-

[1] For BERT and RoBERTa models, all $o_i \in O$ are among the top two completions in the relevant base context. For GPT models, all $o_i \in O$ are among the top ten completions, with the exception of *glasses* in the case of GPT2$_{\text{SMALL}}$.

[2] The set of profession-object pairs is as follows: *florist-flowers*, *baker-bread*, *butcher-meat*, *fisherman-fish*, *painter-paintings*, *optician-glasses*, and *carpenter-furniture*.

[3] The key entity in our dataset is always set to the name *Sebastian*. Though the choice of different names may impact model probabilities, examination of this particular contextual effect is beyond the scope of the present paper.

tual mention associated with completion candidate o_i to which a given annotation applies.

Contextual annotations in the dataset are detailed below. In total, the dataset has contextual annotations for 328,776 completion candidates.

Correctness label This is a binary variable indicating whether completion candidate o_i is the correct completion for the context (whether $o_i = o*$). If models primarily rely on true entity information in the context to inform predictions, this should be the key factor in determining probabilities.

Profession/completion index This annotation simply identifies the completion candidate (and by extension, its associated profession).

Entity/associate distance This measures the distance (in number of words) between the key entity $e*$ and contextual mention m_i associated with o_i. We predict that models may be sensitive to proximity between entity and attribute mentions for establishing links between the two. Specifically, we might predict that the closer an attribute mention is to the key entity $e*$ (whether or not it is the correct profession $p*$), the higher the probability of its corresponding object o_i might be at t_{pred}.

Associate/target distance This measures distance (in number of words) between the context mention m_i associated with completion o_i, and the target position t_{pred}. We hypothesize that the closer a mention is to the target position, the greater its effect on probabilities at that position may be—and as a result, a completion associated with that mention may receive a boost in probability.

Baseline completion probability This refers to the probability of completion o_i when it is the correct completion $o*$ in the base context (e.g., the probability of *flowers* in "Sebastian works as a florist. For his job, Sebastian sells ____"). This is intended to capture the baseline strength of o_i in the presence of a contextual associate. This is model-specific, with probabilities used for analyzing a given model computed based on that model.

Maximum competitor baseline probability This refers to the maximum baseline completion probability among all $o_i \in O^C$. This annotation is not specific to the completion, but rather to the context. We hypothesize that when certain o_i have particularly strong baseline probability in the presence of an associate, this will impact the probability distribution, and by extension the probability of any candidate completion o_i. This is also a model-specific computation.

4 Experimental Setup

We apply our synthetic dataset for a number of analyses on several popular pre-trained LMs.

4.1 Analyses

We examine model "accuracy", defined as the proportion of contexts for which the correct completion $o*$ is assigned a higher probability than any alternative completions $o_i \in O^C$. Then, we use correlation and regression to explore the influence of contextual variables on completion probabilities.

4.2 Models

We experiment with three classes of pre-trained LMs, and various size settings within each class. For the models analyzed in this paper, we use the implementation of (Wolf et al., 2020).

BERT (Devlin et al., 2019) We experiment with two variants: $\text{BERT}_{\text{BASE}}$ (110M parameters), and $\text{BERT}_{\text{LARGE}}$ (340M parameters). For both, we use the uncased version.

RoBERTa (Liu et al., 2019) We experiment with $\text{RoBERTa}_{\text{BASE}}$ (125M parameters) and $\text{RoBERTa}_{\text{LARGE}}$ (355M parameters).

GPT-2 (Radford et al., 2019) We test $\text{GPT2}_{\text{SMALL}}$ (117M parameters), $\text{GPT2}_{\text{MEDIUM}}$ (345M parameters), $\text{GPT2}_{\text{LARGE}}$ (774M parameters) and GPT2_{XL} (1558M parameters).

5 Model accuracy results

In this section we report the percentage of contexts for which models assign a higher probability to the correct completion $o*$ than to any of the competitor completions $o_i \in O^C$ with attractors in the context.

5.1 Overall accuracy

Table 2 shows the accuracy across our tested models for both the multiple entity and single entity settings. These accuracies are averaged over all numbers of attractors and pre-$p*$ facts.

Immediately apparent from this table is that there is substantial variation between models. For the multiple entity setting, accuracies range from only 11% in $\text{RoBERTa}_{\text{BASE}}$ to 73% in GPT2_{XL}. In the single entity setting, accuracies range from 7% in $\text{GPT2}_{\text{SMALL}}$ to 97% in $\text{RoBERTa}_{\text{LARGE}}$.

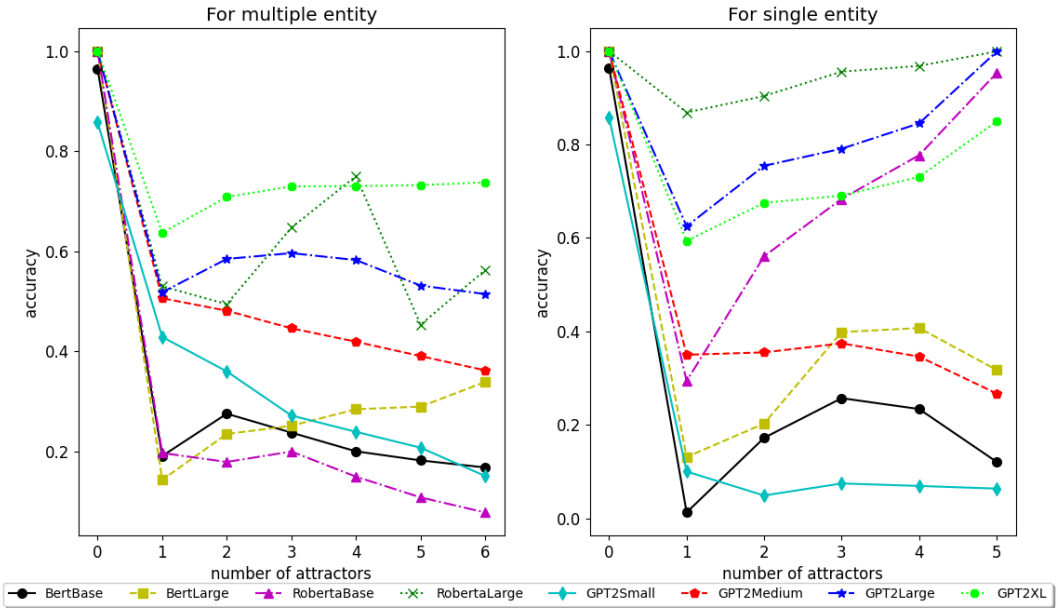

Figure 1: Impact of number of attractors on accuracy

Model	Multiple entity	Single entity
$\text{BERT}_{\text{BASE}}$	0.18	0.2
$\text{BERT}_{\text{LARGE}}$	0.30	0.35
$\text{RoBERTa}_{\text{BASE}}$	0.11	0.77
$\text{RoBERTa}_{\text{LARGE}}$	0.56	**0.97**
$\text{GPT2}_{\text{SMALL}}$	0.20	0.07
$\text{GPT2}_{\text{MEDIUM}}$	0.39	0.34
$\text{GPT2}_{\text{LARGE}}$	0.54	0.86
GPT2_{XL}	**0.73**	0.74

Table 2: Accuracy in preferring correct completion

We also see noteworthy differences between multiple entity and single entity settings. For most models, accuracy is higher for the single entity setting—and for RoBERTa and $\text{GPT2}_{\text{LARGE}}$, this difference is quite substantial. $\text{GPT2}_{\text{SMALL}}$, by contrast, shows comparatively poor accuracy in both settings, but more than doubles its accuracy in the multiple entity setting relative to single entity.

We see consistent advantages for larger models over smaller models of a given class, though for the single entity setting, the largest BERT model falls far short of the smallest RoBERTa model, and $\text{RoBERTa}_{\text{LARGE}}$ soundly outperforms GPT2_{XL}.[4]

[4]The appendix shows accuracies plotted against number of model parameters. While there is an overall upward trend, it is clear that larger is not strictly better for this task.

5.2 Accuracy by number of attractors

Linzen et al. (2016) and Gulordava et al. (2018) demonstrate that increasing the number of attractors weakens performance on a syntactic number agreement task. Here we test the effects of increasing attractors on models' retention of entity profession information. Figure 1 shows the results.

For both settings and for all models, we see that the addition of the first attractor causes a substantial dip in accuracy relative to the zero-attractor (base context) condition.[5] This drop is particularly dramatic for the BERT models, and for $\text{GPT2}_{\text{SMALL}}$ in the single entity setting. Most robust to the first attractor is $\text{RoBERTa}_{\text{LARGE}}$ in the single entity setting, where we see a slight drop, but overall surprising stability across all numbers of attractors.

When we turn to changes in accuracy as we add more attractors, we see that in the multiple entity setting some models do show a trend such that accuracy reduces as the number of attractors increases—particularly $\text{GPT2}_{\text{SMALL}}$ and $\text{GPT2}_{\text{MEDIUM}}$. However, GPT2_{XL} and $\text{BERT}_{\text{LARGE}}$ show *increases* in accuracy as the number of attractors increases, and $\text{RoBERTa}_{\text{LARGE}}$ shows a marked increase in accuracy at 4 attractors, followed by a drop at higher numbers of attractors. For the single entity setting, most models in fact show better accuracy with more attractors—at least up to an intermediate number of

[5]Zero-attractor accuracy is calculated by comparing probability of correct completion against all other $o_i \in O$.

	Multiple entity				Single entity			
	ATD	EAD	BCP	MCP	ATD	EAD	BCP	MCP
$\text{BERT}_{\text{BASE}}$	-0.14	0.05	0.46	-0.28	0.094	-0.25	0.24	-0.0068
$\text{BERT}_{\text{LARGE}}$	-0.14	0.025	0.43	-0.42	-0.015	-0.12	0.2	-0.25
$\text{RoBERTa}_{\text{BASE}}$	-0.26	0.19	0.26	-0.037	0.32	-0.29	0.34	-0.095
$\text{RoBERTa}_{\text{LARGE}}$	0.16	-0.17	0.26	-0.29	0.46	-0.37	0.013	-0.13
$\text{GPT2}_{\text{SMALL}}$	-0.22	0.12	0.24	-0.04	0.041	-0.084	0.38	-0.088
$\text{GPT2}_{\text{MEDIUM}}$	0.069	-0.098	0.54	-0.36	0.11	-0.18	0.61	-0.4
$\text{GPT2}_{\text{LARGE}}$	0.12	-0.1	0.41	-0.38	0.43	-0.36	0.37	-0.18
GPT2_{XL}	0.24	-0.18	0.55	-0.27	0.34	-0.21	0.54	-0.29

Table 3: Spearman correlation with log model completion probabilities. ATD = associate/target distance, EAD = entity/associate distance, BCP = baseline completion probability, MCP = max. competitor baseline probability.

attractors. Only $\text{GPT2}_{\text{MEDIUM}}$ shows a slight overall downward trend. This suggests that for some models under some circumstances, adding more attractors actually facilitates deployment of earlier entity information for the current prediction.

6 Examining contextual effects on completion probabilities

We find above that models are sensitive across the board to the addition of a first attractor—however, the relationship between number of attractors and model accuracy varies across models and depends on the attractor setting. In this section we take advantage of our contextual annotations from Section 3.4 to examine the relationships between these different contextual variables and the probabilities that models assign to candidate completions.

6.1 Correlations with superficial variables

We first compute Spearman rank correlations between contextual variables for all candidate completions $o_i \in O^C$, and log model probabilities for those completions. We focus for now on the four continuous contextual variables: entity/associate distance, associate/target distance, baseline completion probability, and maximum competitor baseline probability. All of these continuous variables are what we would characterize as "superficial" contextual variables, since any of them may be used as cues by LMs, but none are reliable indicators of whether a completion is actually appropriate.

Table 3 shows the resulting correlations. For the multiple entity setting, the variable that shows the strongest correlation across the board is the baseline completion probability. The correlations are particularly strong for the BERT models and the larger GPT2 models. This suggests that completion probabilities in these models are driven in large part by the baseline probability that this completion receives when triggered by an associate in the context, regardless of whether that completion is plausible in the target position. Maximum competitor baseline probability shows the next strongest overall correlations—particularly for $\text{BERT}_{\text{LARGE}}$, suggesting that that model is especially susceptible to effects of attractors on competitor completions (consistent with patterns seen in Figure 1).

The distance-based predictors show substantially weaker correlations. For associate/target distance, BERT models, $\text{RoBERTa}_{\text{BASE}}$ and $\text{GPT2}_{\text{SMALL}}$ do tend to assign weaker probabilities to completions with associates that are more distant from t_{pred}—but $\text{RoBERTa}_{\text{LARGE}}$, $\text{GPT2}_{\text{LARGE}}$ and GPT2_{XL} show the opposite trend. Correlations for the entity/attractor distance are mostly even weaker.[6]

In the single entity setting, the trends shift. For some models the baseline completion probability is still the strongest correlation—this is especially true for $\text{GPT2}_{\text{MEDIUM}}$ and GPT2_{XL}. However, the correlation of the maximum competitor probabilities has dropped. The distance measures, on the other hand, now show stronger correlations for many models—RoBERTa in particular now shows solid correlations for both distance measures, as do the larger GPT models. $\text{BERT}_{\text{BASE}}$ additionally now shows a greater correlation with the entity/associate distance.

These trends suggest that with multiple entity and profession mentions, models put more weight on baseline completion strengths, and less weight on distances between key elements. However,

[6]These two distance measures are negatively correlated, as one might expect, but since they are varied independently, they do capture importantly different information.

	Multiple entity			Single entity		
	R^2 full model	R^2 w/o correctness label	Correctness correlation	R^2 full model	R^2 w/o correctness label	Correctness correlation
$\text{BERT}_{\text{BASE}}$	0.3	0.29	0.04	0.34	0.33	0.16
$\text{BERT}_{\text{LARGE}}$	0.46	0.3	0.28	0.17	0.11	0.23
$\text{RoBERTa}_{\text{BASE}}$	0.36	0.34	0.01	0.54	0.43	0.5
$\text{RoBERTa}_{\text{LARGE}}$	0.54	0.44	0.41	0.66	0.41	0.76
$\text{GPT2}_{\text{SMALL}}$	0.27	0.23	0.07	0.33	0.29	-0.02
$\text{GPT2}_{\text{MEDIUM}}$	0.69	0.65	0.22	0.6	0.6	0.19
$\text{GPT2}_{\text{LARGE}}$	0.46	0.35	0.37	0.63	0.45	0.69
GPT2_{XL}	0.6	0.46	0.48	0.62	0.47	0.57

Table 4: R^2 for model with all six predictors ("full model"), and with correctness label removed ("w/o correctness label"), plus point-biserial correlations between log probabilities and correctness label ("Correctness correlation").

when a single entity is being linked to a list of attributes, distances between elements appear to play a more substantial role for many models.[7]

6.2 Predicting probabilities

To examine the collective predictive power of our annotated contextual variables, we fit ordinary least squares regression models to the log probabilities of all candidate completions $o_i \in O^C$. For this regression analysis we include not only the four continuous variables analyzed in Section 6.1, but also our two discrete variables: correctness label and profession/completion index.[8] We will first examine how much variance is accounted for in general by these predictors, and in the next section we will examine the contribution of the key predictor of correctness label.

Table 4, in the "R^2 full model" columns, shows the R^2 values for the model with all six predictors. The predictors account for the most variance in $\text{RoBERTa}_{\text{LARGE}}$, $\text{GPT2}_{\text{MEDIUM}}$, and GPT2_{XL} across both settings, with additionally high R^2 in the single entity setting for $\text{RoBERTa}_{\text{BASE}}$ and $\text{GPT2}_{\text{LARGE}}$. In general, for models of a given class, larger models have more variance accounted for, suggesting that larger models make heavier use of the identified predictors.

6.3 Examining role of correctness label

Now we turn to disentangling models' use of superficial cues from a more sophisticated ability to process and retain entity information, and to reactivate this information appropriately for predictions. The correctness label is the key predictor associated with this more sophisticated capability, so in this section we compare regression models with and without this predictor, to gauge its contribution relative to the more superficial predictors.

Table 4, in the "R^2 w/o correctness label" columns, shows the R^2 values for the model without the correctness label. Comparing these values against those for the full model, we see quite small differences for $\text{BERT}_{\text{BASE}}$, $\text{GPT2}_{\text{SMALL}}$, and $\text{GPT2}_{\text{MEDIUM}}$ across both conditions, and for $\text{RoBERTa}_{\text{BASE}}$ in the multiple entity setting, and $\text{BERT}_{\text{LARGE}}$ in the single entity setting. Other models, however, show substantial increases in the R^2 values when the correctness label is incorporated as a predictor, suggesting that for these models, the key entity information has non-trivial impact beyond that of the more superficial variables.

To complement this analysis, we also report point-biserial correlations between model log probabilities and correctness label, shown in the Table 4 "Correctness correlation" columns. The magnitudes of these correlations align with what we see in the regression analysis: models with the strongest impact of completion correctness are $\text{RoBERTa}_{\text{LARGE}}$, $\text{GPT2}_{\text{LARGE}}$, and GPT2_{XL}, with $\text{BERT}_{\text{LARGE}}$ showing stronger sensitivity to correctness in the multiple entity setting, and $\text{RoBERTa}_{\text{BASE}}$ stronger sensitivity in the single-entity setting. Sensitivity to correctness is generally stronger for the single entity setting, with $\text{RoBERTa}_{\text{LARGE}}$ showing particularly striking correlation in this setting, consistent with the results from Section 5.[9]

[7] All correlations tested here were statistically significant at $p < .0001$, except ATD–$\text{BERT}_{\text{LARGE}}$, p=0.0042 ; BCP–$\text{RoBERTa}_{\text{LARGE}}$, p=0.01601 ; MCP–$\text{BERT}_{\text{BASE}}$, p=0.197.

[8] Profession/completion index is not meaningful for correlations, but comparison of regression models with and without it show that it accounts for significant variance over that of the most related variable, baseline completion probability.

[9] All likelihood ratio tests and point-biserial correlations

	Single sentence	Original
$\text{BERT}_{\text{BASE}}$	0.18	0.18
$\text{BERT}_{\text{LARGE}}$	0.38	0.30
$\text{RoBERTa}_{\text{BASE}}$	0.16	0.11
$\text{RoBERTa}_{\text{LARGE}}$	**0.75**	0.56
$\text{GPT2}_{\text{SMALL}}$	0.50	0.20
$\text{GPT2}_{\text{MEDIUM}}$	0.39	0.39
$\text{GPT2}_{\text{LARGE}}$	0.46	0.54
GPT2_{XL}	0.67	**0.73**

Table 5: Accuracy of single-sentence multiple entity setting, compared to original multiple entity accuracy

7 Revisiting multiple versus single entity

What is the source of performance differences between single and multiple entity settings? Since these settings differ both in number of entities mentioned, and in number of sentences used, we run a follow-up test with a modified multiple entity setting, in which attractors still consist of new entities and their professions, but the list of entities and professions is contained within a single sentence. The results of this test are shown in Table 5. We see that BERT and RoBERTa models, as well as $\text{GPT2}_{\text{SMALL}}$, improve in accuracy when multiple-entity attributes are described in a single sentence—while the larger GPT models degrade in performance or do not change. The changes are large enough that $\text{RoBERTa}_{\text{LARGE}}$ takes the lead with the highest accuracy over GPT2_{XL}. This suggests that stronger performance in the single entity setting is facilitated by packaging of attribute descriptions in a single sentence. However, the RoBERTa accuracies in this follow-up test still fall far short of the single entity accuracies (see Table 2), as do $\text{GPT2}_{\text{LARGE}}$ and GPT2_{XL}, suggesting that the need to track multiple entities at once is also a factor creating challenges for these models.

8 Discussion

The above experiments allow us to explore the dynamics of pre-trained LMs' sensitivity to contextual variables, in the context of a task of processing and deploying entity attribute information. There are several key takeaways from the analyses.

Models vary dramatically in accuracy. GPT2_{XL} performs best in a multiple entity/multiple-sentence context, and $\text{RoBERTa}_{\text{LARGE}}$ performs best with a single entity, but accuracies vary widely. In general, performance is stronger for single entity contexts—a fact that seems to be related both to the number of sentences and to the number of entities.

All models are susceptible to attractors. All models show a drop in accuracy with the addition of the first attractor, though $\text{RoBERTa}_{\text{LARGE}}$ is quite robust to attractors in the case of a single entity. After the first attractor, while some models degrade in performance as number of attractors increases, other models' accuracies actually improve.

Distance is more of a factor in single entity contexts. When we examine correlation of model probabilities with superficial contextual cues, we find that baseline probabilities overall correlate more strongly with completion probabilities than do distances between key context elements. However, distance between context elements appears to play a larger role in single entity contexts.

Certain LMs are more attuned to sophisticated entity information. Examining predictors of model probabilities, we find in particular that $\text{RoBERTa}_{\text{LARGE}}$, $\text{GPT2}_{\text{LARGE}}$, and GPT2_{XL} show the strongest impact of completion correctness over and above the more superficial predictors, consistent with the overall stronger performance of these models in the tests of completion accuracy.

9 Conclusion

We have presented a new dataset and a series of experiments exploring the capacity of pre-trained LMs to integrate and correctly reactivate entity information in the presence of contextual manipulations. The analyses reveal a number of insights about the processing dynamics of the tested models, with models showing clear sensitivity to contextual interference, but with certain models showing noteworthy robustness to attractors under certain conditions, and strong impact of target entity information beyond the effects of our tested superficial variables. Future work can extend this investigation to understand processing dynamics of additional models, and can expand beyond the focused range of entity information tested here, to continue to improve our understanding of how robustly these models process and deploy information in text.

are statistically significant at $p < .0001$, except the $\text{GPT2}_{\text{SMALL}}$ single entity point-biserial correlation, with $p = .00149$.

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT (1)*.

Allyson Ettinger. 2020. What BERT is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for Computational Linguistics*, 8:34–48.

Richard Futrell, Ethan Wilcox, Takashi Morita, Peng Qian, Miguel Ballesteros, and Roger Levy. 2019. Neural language models as psycholinguistic subjects: Representations of syntactic state. 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 32–42.

Yoav Goldberg. 2019. Assessing BERT's syntactic abilities. *arXiv preprint arXiv:1901.05287*.

Kristina Gulordava, Piotr Bojanowski, Edouard Grave, Tal Linzen, and Marco Baroni. 2018. Colorless green recurrent networks dream hierarchically. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1195–1205. Association for Computational Linguistics.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel R Bowman, and Noah A Smith. 2018. Annotation artifacts in natural language inference data. In *NAACL-HLT (2)*.

John Hewitt and Christopher D. Manning. 2019. A structural probe for finding syntax in word representations. 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 4129–4138, Minneapolis, Minnesota. Association for Computational Linguistics.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know?

Nora Kassner and Hinrich Schütze. 2020. Negated and misprimed probes for pretrained language models: Birds can talk, but cannot fly. Association for Computational Linguistics.

Divyansh Kaushik and Zachary C Lipton. 2018. How much reading does reading comprehension require? a critical investigation of popular benchmarks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5010–5015.

Josef Klafka and Allyson Ettinger. 2020. Spying on your neighbors: Fine-grained probing of contextual embeddings for information about surrounding words. *arXiv preprint arXiv:2005.01810*.

Tomáš Kočiskỳ, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2018. The narrativeqa reading comprehension challenge. *Transactions of the Association for Computational Linguistics*, 6:317–328.

Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. 2016. Assessing the ability of LSTMs to learn syntax-sensitive dependencies. *Transactions of the Association for Computational Linguistics*, 4:521–535.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448.

Kanishka Misra, Allyson Ettinger, and Julia Taylor Rayz. 2020. Exploring BERT's sensitivity to lexical cues using tests from semantic priming. *Findings of EMNLP*.

Nasrin Mostafazadeh, Michael Roth, Annie Louis, Nathanael Chambers, and James Allen. 2017. Lsdsem 2017 shared task: The story cloze test. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics*, pages 46–51.

Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H. Miller, and Sebastian Riedel. 2019. Language models as knowledge bases?

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789.

Matthew Richardson, Christopher JC Burges, and Erin Renshaw. 2013. Mctest: A challenge dataset for the open-domain machine comprehension of text. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 193–203.

Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R Bowman, Dipanjan Das, et al. 2018. What do you learn from context? probing for sentence structure in contextualized word representations. In *International Conference on Learning Representations*.

Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R Bowman. 2020. Blimp: The benchmark of linguistic minimal pairs for english. *Transactions of the Association for Computational Linguistics*, 8:377–392.

Ethan Wilcox, Roger Levy, Takashi Morita, and Richard Futrell. 2018. What do RNN language models learn about filler–gap dependencies? In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 211–221.

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

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *EMNLP*.