Overview

- Masked Language Modeling and the Distributional Hypothesis: Order Word Matters Pre-training for Little
- Neural Path Hunter: Reducing Hallucination in Dialogue Systems via Path Grounding

Masked Language Modeling and the Distributional Hypothesis: Order Word Matters Pre-training for Little

Koustuv Sinha[†‡] **Robin Jia**[†] **Dieuwke Hupkes**[†] **Joelle Pineau**[†‡]

Adina Williams[†] **Douwe Kiela**[†]

[†] Facebook AI Research; [‡] McGill University / Montreal Institute of Learning Algorithms

{koustuvs,adinawilliams,dkiela}@fb.com

Introduction

Masked Language Modeling and the Distributional Hypothesis: Order Word Matters Pre-training for Little

- Masked language model (MLM) pretraining, as epitomized by BERT has proven wildly successful, but the precise reason for this success has remained unclear.
- It has been claimed that BERT has learned "the kind of abstractions that we intuitively believe are important for representing natural language" rather than "simply modeling complex co-occurrence statistics.
- This work tries to uncover how much of MLM's success comes from simple distributional information, as opposed to "the types of syntactic and semantic abstractions traditionally believed necessary for language processing.

Method

Masked Language Modeling and the Distributional Hypothesis: Order Word Matters Pre-training for Little

- The authors disentangle these two hypotheses by measuring the effect of removing word order information during pre-training.
- The authors use the same RoBERTa (base) architecture as the MLM model under investigation. The original 16GB BookWiki corpus is used.
- Two methods for permuting word order are used:
  - Sentence word order permutation and Corpus word order permutation
- Sentence word order permutation. M_1, M_2, M_3, M_4
- Corpus word order bootstrap resample. M_{UG}, M_{UF}
- Further ablations. M_N, M_{NP}, M_{RI}

Experiment-GLUE and PAWS

Masked Language Modeling and the Distributional Hypothesis: Order Word Matters Pre-training for Little

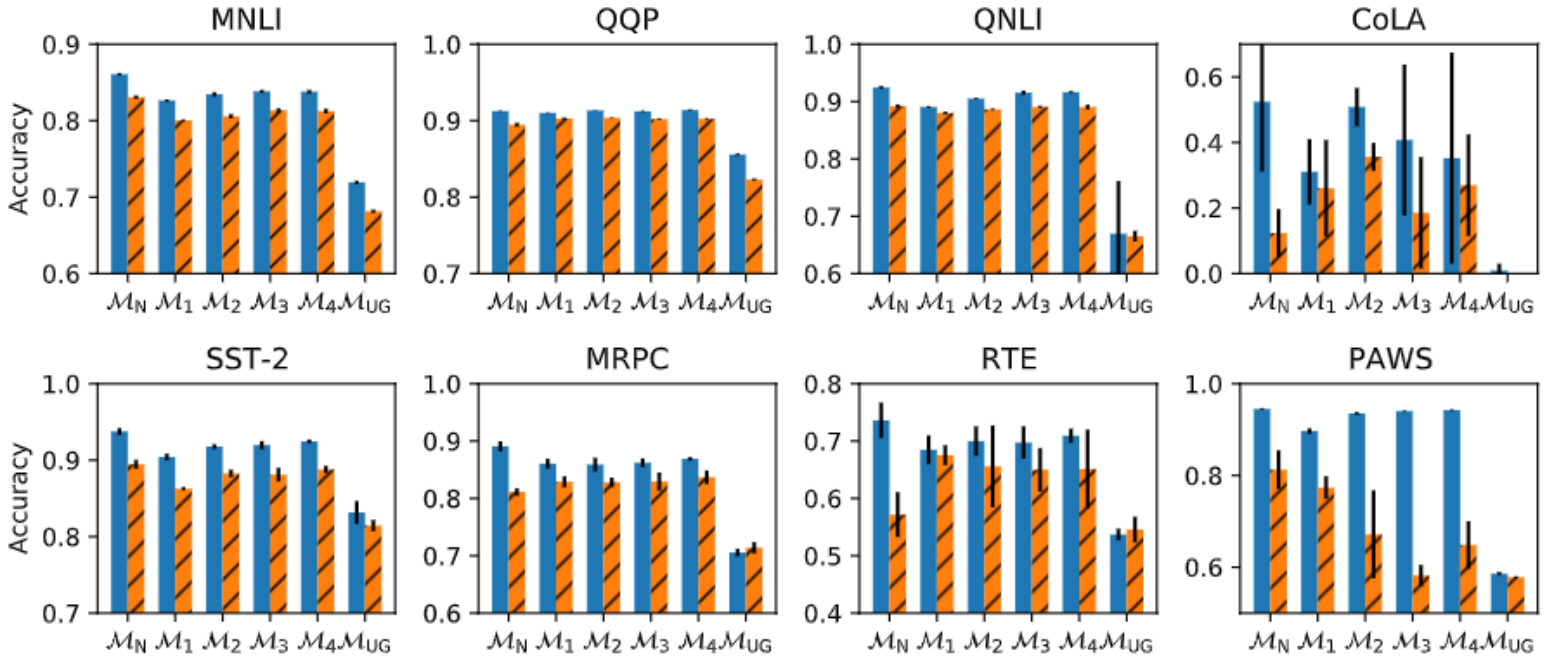
Model	QNLI	RTE	QQP	SST-2	MRPC	PAWS	MNLI-m/mm	CoLA
$\mathcal{M}_{\text{N}}$	92.45 +/- 0.2	73.62 +/- 3.1	91.25 +/- 0.1	93.75 +/- 0.4	89.09 +/- 0.9	94.49 +/- 0.2	86.08 +/- 0.2 / 85.4 +/- 0.2	52.45 +/- 21.2
$\mathcal{M}_{1}$	89.05 +/- 0.2	68.48 +/- 2.5	91.01 +/- 0.0	90.41 +/- 0.4	86.06 +/- 0.8	89.69 +/- 0.6	82.64 +/- 0.1 / 82.67 +/- 0.2	31.08 +/- 10.0
$\mathcal{M}_{2}$	90.51 +/- 0.1	70.00 +/- 2.5	91.33 +/- 0.0	91.78 +/- 0.3	85.90 +/- 1.2	93.53 +/- 0.3	83.45 +/- 0.3 / 83.54 +/- 0.3	50.83 +/- 5.80
$\mathcal{M}_{3}$	91.56 +/- 0.4	69.75 +/- 2.8	91.22 +/- 0.1	91.97 +/- 0.5	86.22 +/- 0.8	94.03 +/- 0.1	83.83 +/- 0.2 / 83.71 +/- 0.1	40.78 +/- 23.0
$\mathcal{M}_{4}$	91.65 +/- 0.1	70.94 +/- 1.2	91.39 +/- 0.1	92.46 +/- 0.3	86.90 +/- 0.3	94.26 +/- 0.2	83.79 +/- 0.2 / 83.94 +/- 0.3	35.25 +/- 32.2
$\mathcal{M}_{\text{RI}}$	62.17 +/- 0.4	52.97 +/- 0.2	81.53 +/- 0.2	82.0 +/- 0.7	70.32 +/- 1.5	56.62 +/- 0.0	65.70 +/- 0.2 / 65.75 +/- 0.3	8.06 +/- 1.60
$\mathcal{M}_{\text{NP}}$	77.59 +/- 0.3	54.78 +/- 2.2	87.78 +/- 0.4	83.21 +/- 0.6	72.78 +/- 1.6	57.22 +/- 1.2	63.35 +/- 0.4 / 63.63 +/- 0.2	2.37 +/- 3.20
$\mathcal{M}_{\text{UF}}$	77.69 +/- 0.4	53.84 +/- 0.6	85.92 +/- 0.1	84.00 +/- 0.6	71.35 +/- 0.8	58.43 +/- 0.3	72.10 +/- 0.4 / 72.58 +/- 0.4	8.89 +/- 1.40
$\mathcal{M}_{\text{UG}}$	66.94 +/- 9.2	53.70 +/- 1.0	85.57 +/- 0.1	83.17 +/- 1.5	70.57 +/- 0.7	58.59 +/- 0.3	71.93 +/- 0.2 / 71.33 +/- 0.5	0.92 +/- 2.10

Table 1: GLUE and PAWS-Wiki dev set results on different RoBERTa (base) models trained on variants of the

- The model without access distributional or word order information, M_{UG} (corpus randomization) performs much worse than M_N overall.
- We observe a significant improvement on all tasks when we give models access to sentence-level distributional information during pre-training.
- Overall, these results confirm our hypothesis that RoBERTa's good performance on downstream tasks can be largely explained by the distributional prior.

Experiment-GLUE and PAWS-Word order-permuted fine-tuning

Masked Language Modeling and the Distributional Hypothesis: Order Word Matters Pre-training for Little



- On QQP and QNLI, accuracy decreases only slightly when models are fine-tuned on shuffled data, suggesting that word order is not very important for these tasks.
- On the other hand, for the other six datasets, we see noticeable drops in accuracy when fine-tuning on shuffled data and testing on normal order.

Experiment-Parametric Probing-Pareto Probing

Masked Language Modeling and the Distributional Hypothesis: Order Word Matte

- We use the “difficult" probe: dependency parsing (DEP), as well as the “easy“ probes: dependency arc labeling (DAL) and POS tag prediction (POS).
- For DEP, we observe that the UAS scores also follow a linear trend as the fine-tuning results in that $M_{UG} \approx M_1 < M_2 < M_3 < M_4 < M_N$
- For POS and DAL, since these tasks are simpler than DEP, the gap between M_N and unnaturally pre-trained models reduces even more drastically.

Model	UD EWT		PTB	
	MLP	Linear	MLP	Linear
$\mathcal{M}_N$	80.41 +/- 0.85	66.26 +/- 1.59	86.99 +/- 1.49	66.47 +/- 2.77
$\mathcal{M}_1$	69.26 +/- 6.00	56.24 +/- 5.05	79.43 +/- 0.96	57.20 +/- 2.76
$\mathcal{M}_2$	78.22 +/- 0.88	64.96 +/- 2.32	84.72 +/- 0.55	64.69 +/- 2.50
$\mathcal{M}_3$	77.80 +/- 3.09	64.89 +/- 2.63	85.89 +/- 1.01	66.11 +/- 1.68
$\mathcal{M}_4$	78.04 +/- 2.06	65.61 +/- 1.99	85.62 +/- 1.09	66.49 +/- 2.02
$\mathcal{M}_{UG}$	74.15 +/- 0.93	65.69 +/- 7.35	80.07 +/- 0.79	57.28 +/- 1.42

Table 2: Unlabeled Attachment Score (UAS) on the dependency parsing task (DEP) on two datasets, UD

Model	UD EWT		PTB	
	MLP	Linear	MLP	Linear
$\mathcal{M}_N$	93.74 +/- 0.15	88.82 +/- 0.42	97.07 +/- 0.38	93.1 +/- 0.65
$\mathcal{M}_1$	88.60 +/- 3.43	80.76 +/- 3.38	95.33 +/- 0.37	87.83 +/- 1.86
$\mathcal{M}_2$	93.39 +/- 0.45	87.58 +/- 1.06	96.96 +/- 0.15	91.80 +/- 0.50
$\mathcal{M}_3$	92.89 +/- 0.65	86.78 +/- 1.32	97.03 +/- 0.13	91.70 +/- 0.70
$\mathcal{M}_4$	92.83 +/- 0.61	87.23 +/- 0.77	96.96 +/- 0.12	92.08 +/- 0.39
$\mathcal{M}_{UG}$	89.10 +/- 0.21	79.75 +/- 0.5	94.12 +/- 0.01	84.15 +/- 0.51

Table 3: Accuracy on the part-of-speech labelling task

Model	UD EWT		PTB	
	MLP	Linear	MLP	Linear
$\mathcal{M}_N$	89.63 +/- 0.60	84.35 +/- 0.78	93.96 +/- 0.63	88.35 +/- 1.00
$\mathcal{M}_1$	83.55 +/- 3.31	75.26 +/- 3.08	91.10 +/- 0.38	82.34 +/- 1.37
$\mathcal{M}_2$	88.57 +/- 0.68	82.05 +/- 1.10	93.27 +/- 0.26	86.88 +/- 0.87
$\mathcal{M}_3$	88.69 +/- 1.09	82.37 +/- 1.26	93.46 +/- 0.29	87.12 +/- 0.72
$\mathcal{M}_4$	88.66 +/- 0.76	82.58 +/- 1.04	93.49 +/- 0.33	87.30 +/- 0.79
$\mathcal{M}_{UG}$	84.93 +/- 0.34	76.30 +/- 0.52	89.98 +/- 0.43	78.59 +/- 0.68

Table 4: Accuracy on the dependency arc labelling task

Experiment-Parametric Probing-SentEval

Masked Language Modeling and the Distributional Hypothesis: Order Word Matters Pre-training for Little

Model	Length (Surface)	WordContent (Surface)	TreeDepth (Syntactic)	TopConstituents (Syntactic)	BigramShift (Syntactic)	Tense (Semantic)	SubjNumber (Semantic)	ObjNumber (Semantic)	OddManOut (Semantic)	CoordInversion (Semantic)
$\mathcal{M}_N$	78.92 +/- 1.91	31.83 +/- 1.75	35.97 +/- 1.38	**78.26** +/- 4.08	**81.82** +/- 0.55	87.83 +/- 0.51	85.05 +/- 1.23	75.94 +/- 0.68	58.40 +/- 0.33	**70.87** +/- 2.46
$\mathcal{M}_1$	88.33 +/- 0.14	**64.03** +/- 0.34	40.24 +/- 0.20	70.94 +/- 0.38	58.37 +/- 0.40	87.88 +/- 0.08	83.49 +/- 0.12	**83.44** +/- 0.06	56.51 +/- 0.26	56.98 +/- 0.50
$\mathcal{M}_2$	**93.54** +/- 0.29	62.52 +/- 0.21	**41.40** +/- 0.32	74.31 +/- 0.29	75.44 +/- 0.14	**87.91** +/- 0.35	84.88 +/- 0.11	83.98 +/- 0.14	57.60 +/- 0.36	59.46 +/- 0.37
$\mathcal{M}_3$	91.52 +/- 0.16	48.81 +/- 0.26	38.63 +/- 0.61	70.29 +/- 0.31	77.36 +/- 0.12	86.74 +/- 0.12	83.83 +/- 0.38	80.99 +/- 0.26	57.01 +/- 0.21	60.00 +/- 0.26
$\mathcal{M}_4$	92.88 +/- 0.15	57.78 +/- 0.36	40.05 +/- 0.29	72.50 +/- 0.51	76.12 +/- 0.29	88.32 +/- 0.13	**85.65** +/- 0.13	82.95 +/- 0.05	**58.89** +/- 0.30	61.31 +/- 0.19
$\mathcal{M}_{UG}$	86.69 +/- 0.33	36.60 +/- 0.33	32.53 +/- 0.76	61.54 +/- 0.60	57.42 +/- 0.04	68.45 +/- 0.23	71.25 +/- 0.12	66.63 +/- 0.21	50.06 +/- 0.40	56.26 +/- 0.17

- The MN pre-trained model scored better than the unnatural word order models for only 1 out of 5 semantic tasks and in none of the lexical tasks.
- However, MN does score higher for 2 out of 3 syntactic tasks. Even for these two syntactic tasks, the gap among MUG and MN is much higher than M1 and MN.
- These results show that while natural word order is useful for at least some probing tasks, the distributional prior of sentence word order randomized models alone is enough to achieve a reasonably high accuracy on syntax sensitive probing.

Experiment-Non-Parametric Probing

Masked Language Modeling and the Distributional Hypothesis: Order Word Matters Pre-training for Little

- For each, the objective is for a pre-trained model to provide higher probability to a correct word than to an incorrect one.
- We observe the highest difference between probabilities of the correct and incorrect focus words for the model pretrained on the natural word order (MN).
- With each step from M1 to M4, the difference between probabilities of correct and incorrect focus words increases, showing that pre-trained models with fewer n-gram words perturbed capture more syntax.
- MUG, with the distributional prior ablated, performs the worst, as expected

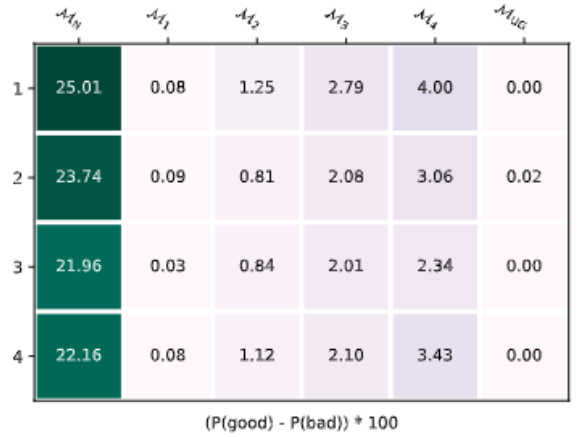


Figure 3: Linzen et al. (2016)

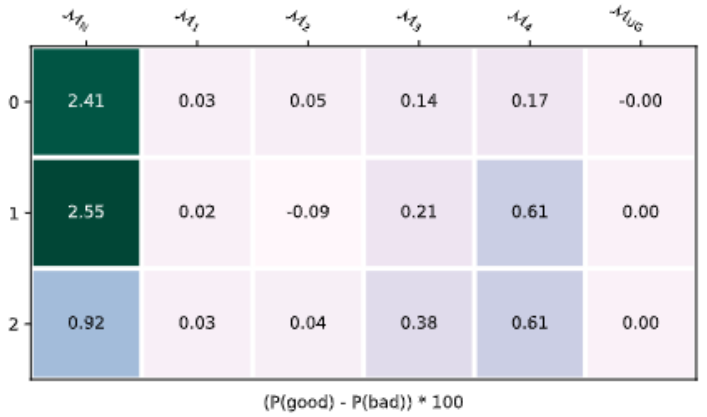


Figure 4: Gulordava et al. (2018)

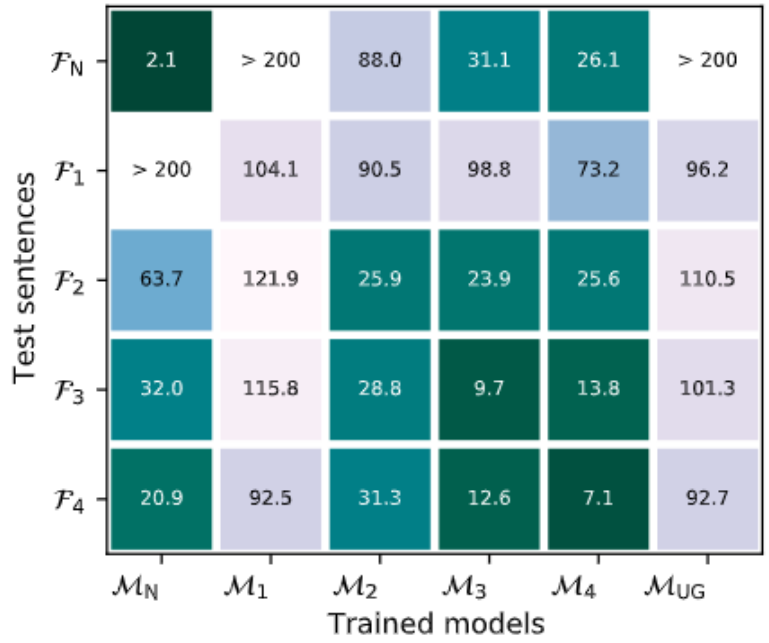
Experiment-Perplexity analysis

Masked Language Modeling and the Distributional Hypothesis: Order Word Matters Pre-training for Little

- We measure Pseudo Perplexity:

$$\mathrm{PLL}(S) = \frac{1}{|S|} \sum_{w \in S} \log P_{\mathrm{MLM}}(w | S_{\setminus w}; \theta)$$

- The pre-trained model MN has the lowest perplexity on the sentences with natural word order.
- Pre-trained models with random word order exhibit significantly higher perplexity than the normal word order sentences (top row).
- Interestingly, with the exception of M1, the models pretrained on randomized data (M2, M3 and M4) all display the lowest perplexity for their respective n = 2; 3; 4.



Summary

Masked Language Modeling and the Distributional Hypothesis: Order Word Matters Pre-training for Little

- The authors revisited the hypothesis that masked language modelling's impressive performance can be explained in part by its ability to learn classical NLP pipelines, using targeted pre-training on sentences with various degrees of randomization in their word order.
- Instead, the experiments suggest that MLM's success can be mostly explained by it having learned higher-order distributional statistics that make for a useful prior for subsequent finetuning.
- These results should hopefully encourage the development of better, more challenging tasks that require sophisticated reasoning, and harder probes to narrow down what exact linguistic information is present in learned representations.

Neural Path Hunter: Reducing Hallucination in Dialogue Systems via Path Grounding

Nouha Dziri[*], **Andrea Madotto**[†], **Osmar Zaiane**[*§], **Avishek Joey Bose**[‡]
[*]University of Alberta, [‡]Mila, McGill University, [§]Canada CIFAR AI Chair
[†]The Hong Kong University of Science and Technology
dziri@cs.ualberta.ca

Introduction

Neural Path Hunter: Reducing Hallucination in Dialogue Systems via Path Grounding

- Dialogue systems powered by large pretrained language models (LM) exhibit an innate ability to deliver fluent and natural looking responses.
- Despite their impressive generation performance, these models can often generate factually incorrect statements impeding their widespread adoption.
- In this work, the authors focus on addressing the open problem of hallucination of factually invalid statements in knowledge grounded dialogue systems where the source of knowledge is a KG.

Introduction

Neural Path Hunter: Reducing Hallucination in Dialogue Systems via Path Grounding

- The authors focus on factoids of knowledge represented as heterogenous G = (V; E; R), termed Knowledge Graphs (KG).
- Each KG consists of a set of directed edge triples t = [SBJ], [PRE], [OBJ]
- Hallucination can take form as either intrinsic or extrinsic:
  - **Definition of Extrinsic Hallucination.** An extrinsic Hallucination corresponds to an utterance that brings a new span of text that does not correspond to a valid triple in G_c^k.
  - **Definition of Intrinsic Hallucination.** An intrinsic hallucination corresponds to an utterance that misuses either [SBJ] or [OBJ] in G_c^k such that there is no direct path between the two entities.

Introduction

Neural Path Hunter: Reducing Hallucination in Dialogue Systems via Path Grounding

GPT2	Hallucination			Faithfulness	Generic	Coherence	Fluency
	Ex	In	B				
Greedy	**17.66** ± 2.6	**2.00** ± 3.5	**1.66** ± 0.5	**73.00** ± 3.2	9.5 ± 2.7	81.66 ± 3.2	95.67 ± 1.6
Beam Search	18.33 ± 2.8	3.33 ± 3.8	4.00 ± 1.8	71.00 ± 3.9	6.33 ± 2.7	83.33 ± 1.6	97.00 ± 1.9
Nucleus 0.9	20.33 ± 2.1	4.00 ± 3.6	2.33 ± 3.6	69.66 ± 2.3	**3.66** ± 3.2	83.66 ± 2.4	**99.10** ± 0.6
Nucleus 0.5	23.33 ± 2.2	5.33 ± 3.1	4.33 ± 0.8	59.90 ± 2.5	7.00 ± 2.6	**87.66** ± 2.1	98.34 ± 0.4
Top20	25.33 ± 1.5	7.00 ± 2.6	5.00 ± 1.5	49.00 ± 0.6	15.66 ± 1.8	80.33 ± 1.6	97.34 ± 0.5

Table 1: Human assessment of random 300 GPT2 dialogue responses generated based on the test OpenDialkg data (Moon et al., 2019). "Ex", "In" and "B" mean extrinsic hallucination, intrinsic hallucination and Both respectively. Each cell in the table represents the percentage of responses with a specific dialogue property (mean preferences ±90% confidence intervals).

- **Observation** 1. Humans notice that most hallucinations in KG-grounded dialogue systems are extrinsic hallucinations.
- **Observation** 2. A hallucination occurs the least in dialogue responses generated using a greedy decoding scheme. Conversely, top-k sampling results in the highest hallucination percentage (37.33%).

Introduction

Neural Path Hunter: Reducing Hallucination in Dialogue Systems via Path Grounding

GPT2	Hallucination			Faithfulness	Generic	Coherence	Fluency
	Ex	In	B				
Greedy	**17.66** ± 2.6	**2.00** ± 3.5	**1.66** ± 0.5	**73.00** ± 3.2	9.5 ± 2.7	81.66 ± 3.2	95.67 ± 1.6
Beam Search	18.33 ± 2.8	3.33 ± 3.8	4.00 ± 1.8	71.00 ± 3.9	6.33 ± 2.7	83.33 ± 1.6	97.00 ± 1.9
Nucleus 0.9	20.33 ± 2.1	4.00 ± 3.6	2.33 ± 3.6	69.66 ± 2.3	**3.66** ± 3.2	83.66 ± 2.4	**99.10** ± 0.6
Nucleus 0.5	23.33 ± 2.2	5.33 ± 3.1	4.33 ± 0.8	59.90 ± 2.5	7.00 ± 2.6	**87.66** ± 2.1	98.34 ± 0.4
Top20	25.33 ± 1.5	7.00 ± 2.6	5.00 ± 1.5	49.00 ± 0.6	15.66 ± 1.8	80.33 ± 1.6	97.34 ± 0.5

Table 1: Human assessment of random 300 GPT2 dialogue responses generated based on the test OpenDialkg data (Moon et al., 2019). "Ex", "In" and "B" mean extrinsic hallucination, intrinsic hallucination and Both respectively. Each cell in the table represents the percentage of responses with a specific dialogue property (mean preferences ±90% confidence intervals).

- **Observation** 3**.** Increased diversity in response generation —i.e.(less generic), is positively correlated with an increase in hallucination.
- **Observation** 4. Responses from all models tend to be highly relevant and fluent, which reflects the characteristic of powerful pre-trained LMs in generating human-like responses.

Method

Neural Path Hunter: Reducing Hallucination in Dialogue Systems via Path Grounding

- NEURAL PATH HUNTER (NPH) a lightweight refinement strategy that can be easily applied to any generated response without retraining the model.
- NPH is composed of two modules: A token-level hallucination critic and an autoregressive entity mention retriever.
- The first module flags and masks out hallucinated entities in an existing response and can be trained offline.
- The second module builds a contextual representation of these problematic tokens which are then used to retrieve more faithful entities.

Each instance in the dataset is composed of a dialogue history $\mathcal{D} = (x_1, \ldots, x_n)$, a set of j triples at turn n, $\mathcal{K}_n = (t_1, t_2, \ldots t_j)$ which together with $\mathcal{D}$ must be used towards generating the response. Here each individual triple $t_i = \langle \texttt{[SBJ]}, \texttt{[PRE]}, \texttt{[OBJ]} \rangle$ is extracted from a provided KG. Thus the task is to generate a response x_{n+1} that is faithful to a non-empty subset $M_n \subset$

Method

Neural Path Hunter: Reducing Hallucination in Dialogue Systems via Path Grounding

Token-level hallucination critic

- To flag entities of concern we design a token-level hallucination critic C. C is taken to be a pretrained LM with an additional classification.
- To train C, we choose to cast the problem as a sequence labelling task where a binary label is predicted at each word position.
- We create a synthetic dataset consisting of ground truth dialogue samples and corrupted negative samples.

1. **Extrinsic Negatives**. We replace each m_i in x_{n+1} with entities of the same type (e.g., person, location, etc...) but crucially not within $\mathcal{G}_c^k$ and the dialogue history $\mathcal{D}$.

2. **Intrinsic Negatives**. We simply swap every pair `[SBJ]` and `[OBJ]` mentions in x_{n+1}. For example, the response "Crescendo was written by Becca Fitzpatrick" $\rightarrow$ "Becca Fitzpatrick was written by Crescendo" transforms into an intrinsic hallucination as the abstract relation `[PRE]` is not bi-directional.

Method

Neural Path Hunter: Reducing Hallucination in Dialogue Systems via Path Grounding

Entity Mention Retriever

- we feed the dialogue history, the triple set at turn as well as flagged set of entities to obtain contextual hidden state representations

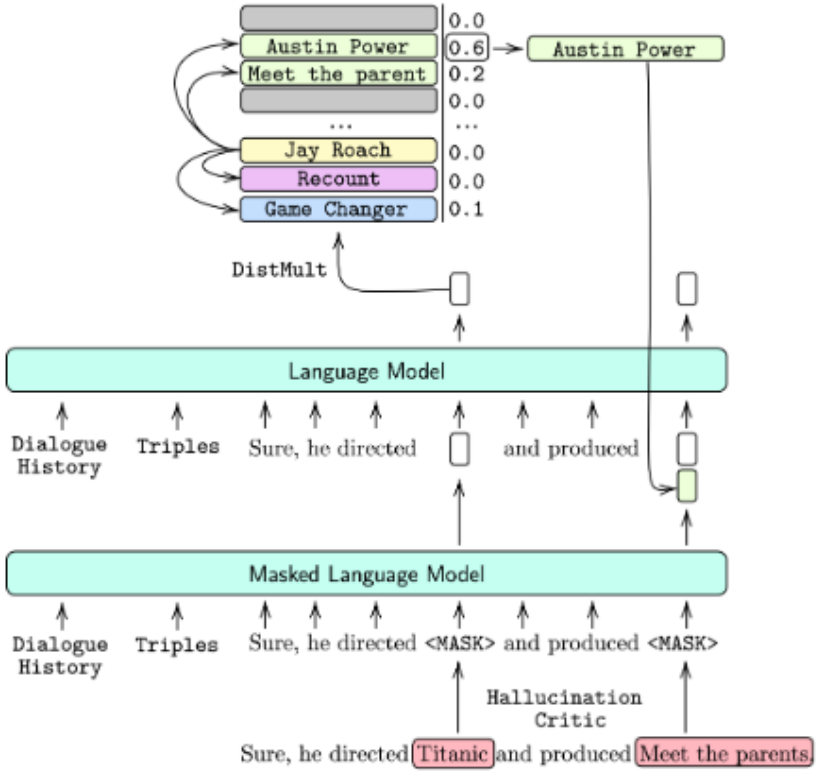
$$H = \mathrm{MLM}(\mathcal{D}, \mathcal{K}_n, M_c)$$

- we simply apply a pooling operation to obtain a single representation for each entity. To obtain the actual query q_i we use an autoregressive LM:

$$q_i = \mathrm{LM}(W(\mathrm{concat}[e_{i-1}, h_i])),$$

- Finally, to retrieve the correct entity for query q_i we simply use a scoring function s to score every KG-Entity memory triple. The NCE loss is used to train the component:

$$\mathcal{L}_{\mathrm{NCE}} = -\log\left(s(t)\right) - \log\left(s(t) + \sum_{j=1}^{n} s(t')\right)$$



Experiment

Neural Path Hunter: Reducing Hallucination in Dialogue Systems via Path Grounding

- The model is evaluate on the OpenDialKG.
- Standard classification metrics such as F1-score, precision and recall are used to evaluate the hallucination critic.
- Hits@k, Mean Rank (MR), and Mean Reciprocal Rank (MRR) to evaluate ability of the Entity Mention Retriever to return a faithful entity.

Experiment-Q1: Identifying Hallucinations

Neural Path Hunter: Reducing Hallucination in Dialogue Systems via Path Grounding

- We ask humans to identify the span of text that is hallucinated w.r.t. to the given triples. In total, we annotate 500 responses generated greedily from GPT2-KG.
- To explore the robustness of our corruption strategies as discussed, we fine-tune the critic C on three different synthetic datasets
- This result highlights that a token-level classifier can indeed detect both extrinsic and intrinsic hallucinations, but also that our corruption strategies are effective.

Model	Precision	Recall	F1
RoBERTa-Intrin	44.9	32.54	37.73
RoBERTa-Extrin	68.65	46.94	55.76
RoBERTa-Intrin-Extrin	**83.05***	**61.02***	**70.35***

Table 3: Performance of the token-level hallucination critic on the 500 human-annotated test data generated based on GPT2-KG (greedy). (* indicates statistical significance with p-value < 0.001)

Experiment-Q2: Reducing Hallucinations

Neural Path Hunter: Reducing Hallucination in Dialogue Systems via Path Grounding

Model	FeQA (↑)	Hallu. Critic (↓)	E. prec. (↑)	BLEU	Rouge-L
GPT2-KG	26.54	19.04	61.21	11.10	30.0
GPT2-KG (+ NPH)	**28.98***	**11.72***	**69.34***	11.29	31.0
GPT2-KG (+ NPH-W/O NCE)	26.02	17.91	65.74	10.98	24.0
GPT2-KG (+ NPH-W. COMPGCN)	26.89	15.41	67.76	**11.79***	**45.0***
GPT2-KG (+ NPH-W/O MLM)	27.01	15.02	68.91	10.88	34.0
AdapterBot	23.11	26.68	42.38	10.08	31.0
AdapterBot (+ NPH)	**27.21***	**18.51***	**62.41***	**10.64***	**32.0***
AdapterBot (+ NPH-W/O NCE)	24.02	25.02	55.12	9.98	28.0
AdapterBot (+ NPH-W. COMPGCN)	25.83	20.23	61.05	10.11	30.0
AdapterBot (+ NPH-W/O MLM)	26.02	21.04	59.98	10.56	29.0
GPT2+KE	19.54	28.87	15.91	5.49	0.19
GPT2+KE (+ NPH)	**26.21***	**20.34***	47.98	**6.06***	21.0
GPT2+KE (+ NPH-W/O NCE)	20.34	24.32	44.58	5.89	20.0
GPT2+KE (+ NPH-W. COMPGCN)	23.23	21.21	**48.17***	6.01	**27.0***
GPT2+KE(+ NPH-W/O MLM)	24.01	22.40	47.56	5.99	27.0
Gold response	30.34	5.2	52.48	-	-

- NPH consistently performs favorably in reducing hallucination across all metrics.
- The strongest iteration of each baseline model is the original model paired with the full NPH module

Experiment–Q3: Query Generation

Neural Path Hunter: Reducing Hallucination in Dialogue Systems via Path Grounding

E. Mem	Model	Negative candidates	PPL	Hits@1	Hits@3	Hits@10	MR	MRR
GPT2	NPH	SANS (Ahrabian et al., 2020)	8.67	**0.73**	**0.92**	**0.99**	**1.76**	**0.83**
		In-Batch Negatives	**8.56**	0.42	0.75	0.94	3.08	0.68
	NPH-W/O NCE	-	9.64	0.02	0.05	0.1	35.49	0.07
	NPH-W/O MLM	SANS (Ahrabian et al., 2020)	9.73	0.47	0.76	0.96	2.83	0.64
		In-Batch Negatives	9.70	0.20	0.43	0.75	9.22	0.36
CompGCN	NPH	SANS (Ahrabian et al., 2020)	8.99	0.13	0.26	0.52	14.27	0.25
		In Batch Negatives	10.04	0.08	0.17	0.43	15.75	0.16
	NPH-W/O NCE	-	10.61	0.04	0.12	0.27	26.50	0.12
	NPH-W/O MLM	SANS (Ahrabian et al., 2020)	9.64	0.08	0.21	0.47	15.52	0.20
		In-Batch Negatives	9.64	0.02	0.05	0.16	80.52	0.07

- Key metrics such as Hits@3 and Hits@10 are nearly saturated when using the complete NPH module. All retrieval metrics drop dramatically, e.g. Hits@1 drops by 70 points compared when LNCE is omitted from the training objective
- SANS negatives are lead to higher both lower perplexity and better retrieval performance across the board. This is not surprising since local negative samples are known to be harder and thus provides a richer learning signal to the retrieval model.

Experiment-Q4: Impact of MLM

Neural Path Hunter: Reducing Hallucination in Dialogue Systems via Path Grounding

E. Mem	Model	Negative candidates	PPL	Hits@1	Hits@3	Hits@10	MR	MRR
GPT2	NPH	SANS (Ahrabian et al., 2020)	8.67	**0.73**	**0.92**	**0.99**	**1.76**	**0.83**
		In-Batch Negatives	**8.56**	0.42	0.75	0.94	3.08	0.68
	NPH-W/O NCE	-	9.64	0.02	0.05	0.1	35.49	0.07
	NPH-W/O MLM	SANS (Ahrabian et al., 2020)	9.73	0.47	0.76	0.96	2.83	0.64
		In-Batch Negatives	9.70	0.20	0.43	0.75	9.22	0.36
CompGCN	NPH	SANS (Ahrabian et al., 2020)	8.99	0.13	0.26	0.52	14.27	0.25
		In Batch Negatives	10.04	0.08	0.17	0.43	15.75	0.16
	NPH-W/O NCE	-	10.61	0.04	0.12	0.27	26.50	0.12
	NPH-W/O MLM	SANS (Ahrabian et al., 2020)	9.64	0.08	0.21	0.47	15.52	0.20
		In-Batch Negatives	9.64	0.02	0.05	0.16	80.52	0.07

- Instead of constructing a contextual representation via the MLM, we instead initialize them randomly during training. The performance degrades substantially
- These findings suggest that the MLM facilitates the learning of rich masked representations by fusing both the left and the right context

Experiment-Q5: Impact of global graph structure

Neural Path Hunter: Reducing Hallucination in Dialogue Systems via Path Grounding

E. Mem	Model	Negative candidates	PPL	Hits@1	Hits@3	Hits@10	MR	MRR
GPT2	NPH	SANS (Ahrabian et al., 2020)	8.67	**0.73**	**0.92**	**0.99**	**1.76**	**0.83**
		In-Batch Negatives	**8.56**	0.42	0.75	0.94	3.08	0.68
	NPH-W/O NCE	-	9.64	0.02	0.05	0.1	35.49	0.07
	NPH-W/O MLM	SANS (Ahrabian et al., 2020)	9.73	0.47	0.76	0.96	2.83	0.64
		In-Batch Negatives	9.70	0.20	0.43	0.75	9.22	0.36
CompGCN	NPH	SANS (Ahrabian et al., 2020)	8.99	0.13	0.26	0.52	14.27	0.25
		In Batch Negatives	10.04	0.08	0.17	0.43	15.75	0.16
	NPH-W/O NCE	-	10.61	0.04	0.12	0.27	26.50	0.12
	NPH-W/O MLM	SANS (Ahrabian et al., 2020)	9.64	0.08	0.21	0.47	15.52	0.20
		In-Batch Negatives	9.64	0.02	0.05	0.16	80.52	0.07

- we notice a dramatic difference in both perplexity and retrieval performance in favor of using simply the output of a pre-trained GPT-2 model.
- any specific turn in dialogue local information—as conversation topics may drift—are significantly more important to generate a faithful response.
- Thus enriching entity embeddings with global structure in G is less beneficial than aligning G_c^k with the representation space of the autoregressive LM.

Summary

Neural Path Hunter: Reducing Hallucination in Dialogue Systems via Path Grounding

- This work investigates the open problem of hallucination in KG-grounded dialogue systems and demonstrate that these models are more susceptible to extrinsic hallucinations which predominantly manifest as the injection of erroneous entities.
- NEURAL PATH HUNTER is proposed to enforce faithfulness in KG-grounded dialogue systems by identifying and refining hallucinations via queries over a k-hop subgraph.
- Disadvantage
  - considered a paired KG that was aligned with dialogue.
  - but in many other applications, such dialogue to KG alignment may be difficult to easily obtain.

Thanks