A Token-level Reference-free Hallucination Detection Benchmark for Free-form Text Generation

Tianyu Liu[1 2*] **Yizhe Zhang**[3] **Chris Brockett**[4] **Yi Mao**[4]
Zhifang Sui[1] **Weizhu Chen**[4] **Bill Dolan**[4]
[1] Peking University [2] Tencent Cloud Xiaowei [3] Meta AI [4] Microsoft Corporation
{tianyu0421,szf}@pku.edu.cn , yizhe.zhang@hotmail.com
{chrisbkt,maoyi,wzchen,billdol}@microsoft.com

Abstract

Large pretrained generative models like GPT-3 often suffer from hallucinating non-existent or incorrect content, which undermines their potential merits in real applications. Existing work usually attempts to detect these hallucinations based on a corresponding oracle reference at a sentence or document level. However ground-truth references may not be readily available for many free-form text generation applications, and sentence- or document-level detection may fail to provide the fine-grained signals that would prevent fallacious content in real time. As a first step to addressing these issues, we propose a novel *token-level, reference-free* hallucination detection task and an associated annotated dataset named HADES (**HA**llucination **DE**tection data**S**et) [1]. To create this dataset, we first perturb a large number of text segments extracted from English language Wikipedia, and then verify these with crowd-sourced annotations. To mitigate label imbalance during annotation, we utilize an iterative model-in-loop strategy. We conduct comprehensive data analyses and create multiple baseline models.

1 Introduction

Automatic text generation using neural natural language generation (NLG) systems is increasingly fluent and thus seemingly plausible in many real-world applications. Large-scale pretrained models like GPT-3 (Brown et al., 2020) are proven to be powerful in understanding and performing free form text generation tasks at human-quality level with a few in-context examples, which dramatically reduces the manual labor needed in many text-based applications and services. Despite their great success, however, neural NLG systems using very large pre-trained models struggle to generate factually accurate and trustworthy text (Devlin et al., 2019; Radford et al., 2019), and exhibit a propensity to hallucinate non-existent or incorrect content that is unacceptable in most user-oriented applications. This poses a major challenge for deploying production NLG systems with realtime generation, where post-examination is impossible.

Existing work has sought to detect hallucination and quantitatively measure generation consistency against a provided reference. Such *reference-based* hallucination detection has been proposed for abstractive summarization (Maynez et al., 2020), machine translation (Wang and Sennrich, 2020), data-to-text generation (Rebuffel et al., 2021), and image caption generation (Rohrbach et al., 2018). For many free-form text generation tasks, however, references are not readily available. For example, in a production NLG system such as a social chatbot using real-time response generation or a document auto-completion system, the generation model often cannot pair its outputs with sufficient reference information, rendering reference-based methods less applicable: i) It may be difficult to even know where to obtain the reference, as obtaining it may be as hard as generating consistent information in the first place; ii) Generation may be at a real-time online setting that demands leveraging only existing context to create new content.

One common setup for qualitatively measuring the level of hallucination is performed at *sentence- or document-level* (Dhingra et al., 2019; Scialom et al., 2019). Related tasks such as fake news detection (Zellers et al., 2019) or fact checking (Thorne and Vlachos, 2018) also adopt this strategy. However, sentence- or document-level detection may not always provide high-resolution signals sufficient to pinpoint the hallucinated text, or can only judge whether a generated sentence or a document

[*] Work was done when Tianyu (intern) and Yizhe was at Microsoft.

[1] Code and data are provided in https://github.com/microsoft/HaDes

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 6723 - 6737
May 22-27, 2022 ©2022 Association for Computational Linguistics

Input: …. She had a large family and lived with her **grandparents** …. In 1933 she gave birth to her first child …. In July 1926, many of her friends attended her **funeral** …

Label1: grandparents → Not Hallucination
Label2: funeral → Hallucination

Figure 1: Overview for reference-free token-level hallucination detection task.

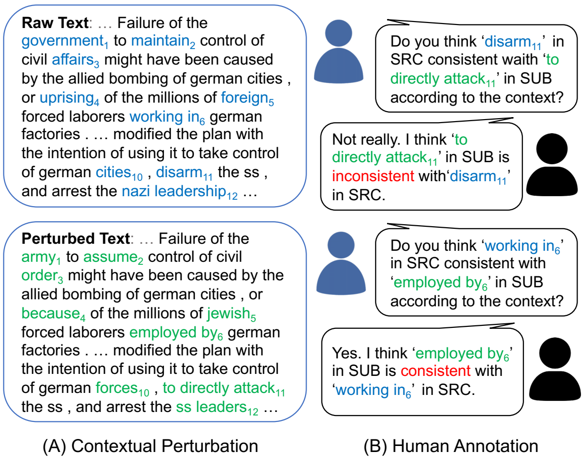
as a whole is a hallucinated artifact. Consequently, these high-level strategies may be insufficient to avoid hallucinations. As an alternative, at decoding time of an NLG system, we suggest that if the locus of hallucination can be identified at the token level, it may be possible to guide beam search or suppress the probability of certain tokens at real-time.

To this end, we propose a *reference-free, token-level hallucination detection* task and introduce an annotated training and benchmark testing dataset that we call HADES (**HA**llucination **DE**tection data**S**et). The *reference-free* property of this task yields greater flexibility in a broad range of generation applications. We expect the *token-level* property of this task to foster the development of models that can detect fine-grained signals of potential hallucination. In conjunction with consulting context to identify self-contradictory statements and access to commonsense and world knowledge, such fine-grained signals, when detected, should further mitigate real-time hallucination.

Our contributions include: **1)** We propose a *reference-free*, *token-level* hallucination detection task for free-form text generation. **2)** We support this task with a dataset that we call HADES, with ∼11k instances extracted from English Wikipedia using an iterative data collection strategy to address data imbalance issues. We also present comprehensive analyses on the statistical features to shed light on what is commonly recognized as hallucination in crowd-sourced judgments and its salient characteristics in free-form text generation. **3)** We create multiple baselines, including feature based models and pretrained models as a first step towards addressing the proposed task.

2 Task Overview

We formulate our hallucination detection task as a binary classification task. As shown in Fig 1, our goal is to assign either a "hallucination"[2](abbreviated as "$\mathcal{H}$") or a "not hallucination" (abbreviated as "$\mathcal{N}$") label to the highlighted spans.



Raw Text: … Failure of the government$_1$ to maintain$_2$ control of civil affairs$_3$ might have been caused by the allied bombing of german cities , or uprising$_4$ of the millions of foreign$_5$ forced laborers working in$_6$ german factories . … modified the plan with the intention of using it to take control of german cities$_{10}$, disarm$_{11}$ the ss , and arrest the nazi leadership$_{12}$ …

Do you think 'disarm$_{11}$' in SRC consistent waith 'to directly attack$_{11}$' in SUB according to the context?

Not really. I think 'to directly attack$_{11}$' in SUB is inconsistent with'disarm$_{11}$' in SRC.

Perturbed Text: … Failure of the army$_1$ to assume$_2$ control of civil order$_3$ might have been caused by the allied bombing of german cities , or because$_4$ of the millions of jewish$_5$ forced laborers employed by$_6$ german factories . … modified the plan with the intention of using it to take control of german forces$_{10}$, to directly attack$_{11}$ the ss , and arrest the ss leaders$_{12}$ …

Do you think 'working in$_6$' in SRC consistent with 'employed by$_6$' in SUB according to the context?

Yes. I think 'employed by$_6$' in SUB is consistent with 'working in$_6$' in SRC.

(A) Contextual Perturbation

(B) Human Annotation

Figure 2: The data collection process of HADES.

To simulate real-world NLG applications, we propose two sub-tasks with "*offline*" and "*online*" settings. In the *offline* setting, it is assumed that generation is complete, so the the model is able perceive the bidirectional context. This could be used in the post-generation examination of NLG systems. For *online* detection, the model can only access the unidirectional preceding context, which simulates on-the-fly generation. Online detection is important in practice as it enables NLG systems to proactively forestall potential hallucinations.

3 Dataset Creation

To collect the HADES dataset, we first perturb "raw text" web data into "perturbed text" (Fig 2A) (Sec 3.2). We then ask human annotators to assess whether the perturbed text spans are hallucinations given the original text (Fig 2B) (Sec 3.3).

3.1 Raw Data Collection

Our raw data are sampled from English WIKI-40B (Guo et al., 2020) dataset. WIKI-40B-EN is a cleaned collection of English Wikipedia articles. We randomly sample from the first paragraphs of these articles and filter out short text of fewer than 5 sentences. We use Wikipedia as our text source since it is stylistically formal and of high quality, and covers diverse topics and domains.

[2] "Hallucination" in our paper refers to certain types of mistakes (Fig 3) made by the NLG models. The notions of "consistency" and "not hallucination" are only for annotation purposes (Sec 3.3).

3.2 Contextual Perturbation

To acquire machine generated text in the free-form, we perturb the raw text [3] using BERT. In applying this contextual perturbation we maintained two principles: i) the fluency and syntactic correctness of the perturbed text should be preserved; ii) the perturbed text should be lexically diverse.

We leave the first two sentences in the raw text unchanged to serve as the preceding context, so as to avoid the "early token curse" (Press et al., 2020) where tokens are evaluated at the beginning with limited context. The text perturbation process is split into three pipelined operations, namely `MASK`, `REPLACE` and `RANK`.

- **i)** In the `MASK` operation, we mask the tokenized words to be replaced with the special token "[MASK]" in the BERT vocabulary. Starting from the third sentence, we randomly mask word spans by a pre-defined mask ratio ρ. By default we only mask one word in each perturbation, except for named entities identified by *Spacy*. We view the entity boundaries as minimal masking units to avoid collocation errors (*e.g.* "San Diego" should be masked as a whole). To reduce trivial instances, we do not mask stop words or punctuation identified by NLTK (Bird, 2006).

- **ii)** In the `REPLACE` operation, we leverage a pretrained BERT-base model to predict the masked span. The mask-then-predict training framework of the BERT model contextualizes the replacement with both preceding and subsequent text. For better fluency, we replace the masked tokens from left to right, *e.g.* a 3-token `REPLACE` operation will be "[MASK] [MASK] [MASK]" → "[A] [MASK] [MASK]" → "[A] [B] [MASK]" → "[A] [B] [C]"[4]. When performing the replacement, we remove the *original* token from the predicted distribution over the vocabulary at each position of the text span, to avoid duplicated text after perturbation. We compared several decoding strategies in token substitution, including greedy, top-k (k=5/10/50) and top-p (p=0.95/0.9/0.8) (Holtzman et al., 2020) sampling methods. For comparison we sample 30 perturbed text for each sampling method and count the number of incoherent perturbations. We choose top-k (k=10) sampling as its good trade-off between diversity (via number of distinct tokens) and coherence (via number of incoherent perturbations).

- **iii)** For each perturbed text, we substitute multiple word spans. Although being locally coherent, the perturbed text may still exhibit some global incoherence and syntactic issues, especially for longer text. We thus post-process the perturbed text with a `RANK` operation as an additional screening step. For each raw text, we generate 20 perturbed candidates and rank them according to language model perplexity using a GPT-2 (117M) model. We only keep the the candidate with lowest perplexity to ensure the fluency and syntactic correctness.

3.3 Data Annotation

We ended up with $\sim$1M perturbed text segments in the pool after contextual perturbation, not all of which contain hallucination, as the BERT model can generate factual information given that it is pretrained on a rich open web corpus. Thus, we sought to further annotate the automatically perturbed texts via crowd-sourcing. Human annotation is prohibitively expensive at this scale, so instead of annotating all 1M perturbed texts, we annotated a subset that is *less trivial* and would lead to a more *balanced* distribution, using an iterative model-in-the-loop annotation approach that is conceptually related to active learning (Cohn et al., 1996; Jia and Liang, 2017; Zellers et al., 2018; Nie et al., 2020).

Human annotation settings To perform the annotations, we hired judges on an internal (the name is redacted for double-blind review) crowd-sourcing platform comparable to AMT. The judges were limited to the North American English speakers with good records (recognized as experts in the platform, rejection rate $\leq 1\%$) and were screened via a simple 10-question qualification test (answering 8 out of 10 questions correctly). They were paid 0.15$ per HIT, which is more than prevailing

[3] In a pilot study, we tried to annotate a token-level dataset based on GPT-3 generated text. However, we found that annotators had trouble achieving consensus if we don't provide the "original text". The size of the resulting data would be small. We thus reduce the ambiguity and subjectivity in the annotation process by asking if the pinpointed position in perturbed text is consistent/hallucinated compared with the original reference text.

[4] It is possible to substitute the original tokens with more or fewer of tokens. However enumerating all possible token lengths is difficult, and empirically we see marginal gain in diversity in the resulting perturbed text. In our experiments we use same number of tokens for replacement.

local minimum wage. Protocols were implemented to block spammers in real time [5]. For each annotation, both original text and perturbed text were shown to the judges, with perturbed text span highlighted. The annotators were asked to determine whether the perturbed text spans are $\mathcal{H}$ (hallucination) or $\mathcal{N}$ (not hallucination) with the original text in terms of factualness and semantic coherence given the context. Each pair was judged by 4 annotators, and up to 6 if consensus was not reached. We retained only those annotations for which consensus was reached. Out of 12,719 annotated instances, 86.12% instances reach consensus and are included in HADES dataset; 78.47% instances reach $\geq$ 80% agreement among annotators, e.g. 4/5 or 5/6 vote for "hallucination" label; 71.24% instances reach 100% agreement in the annotation. For inter-annotator agreement (IAA), the Krippendorf's alpha between the annotators is 0.87.

Iterative Model-in-the-loop annotation Annotating all perturbed text segments is expensive and time-consuming. Thus, we resort to annotating a subset. We applied two principles for selecting the data to be annotated: $i)$ the data should be *balanced*. We found that with randomly sampled instances, the annotated label distribution is heavily skewed toward the "hallucination" class. Presumably most contextualized perturbations result in factual inconsistency to certain extent. However, we aim to have the number of instances in both classes on par with each other, so that the ROC (receiver operating characteristic) curve of tested models can be better characterized. $ii)$ the data for annotation should be *less trivial* [6]. The obvious instances contribute little to model training and method benchmarking, but cost as much annotation effort as other instances.

The challenge is that we cannot know *a priori* the annotation labels and ease of labeling, hence selecting *less trivial* instances and forming a *balanced* label distribution for annotation is not straightforward. To address this challenge, we adopt an iterative *Model-in-the-loop* annotation strategy. Specifically, we split the annotations into several rounds. For each round [7], we first retrain a hallucination detection model (initiated with BERT) based on the annotated instances in the previous rounds. This model is used for selecting the next batch of data to be annotated from the remaining unlabeled data.

To filter out trivial instances and focus on the more *useful* cases, we use a heuristic rule for the automatic screening by abandoning instances where the detection model assigns low or high probability to "hallucination" class (the threshold varies in different rounds to yield reasonable number of candidates). To eliminate cases where the perturbed text paraphrases the original text, we also measured the cosine similarity between the replaced text (through "[CLS]" representation) and corresponding original content using a RoBERTa model (without fine-tuning), and then filtered out cases with a similarity score greater than 0.9. We also remove a large portion of obvious hallucination instances where the target text span is recognized as a DATE or NAME, and replaced by a different DATE[8] or NAME.

In the initial rounds of annotation, we observed extreme label imbalance (around 90% are $\mathcal{H}$ class) between $\mathcal{H}$ (hallucination) and $\mathcal{N}$ (not hallucination) cases. To rebalance the label distribution so that each class received a decent amount of annotation, we performed additional subsamping based on the label predicted by the aforementioned detection model. We assume the human annotation for $\mathcal{H}$ and $\mathcal{N}$ cases is the oracle, indicating actual $\mathcal{H}/\mathcal{N}$. Since the actual "hallucinated" is dominant, we seek to subsample from instances that are predicted as $\mathcal{H}$ by the detection model to make the distribution of actual $\mathcal{H}/\mathcal{N}$ even. To do this, we estimate the true positive rate (TPR, α), true negative rate (TNR, β) and true precision (γ) of the detection model based on the annotation from last round. The hope is that after subsampling, the actual $\mathcal{H}$ (TP + FN) is roughly equal to actual $\mathcal{N}$ (FP + TN). The estimated subsampling ratio R for the predicted $\mathcal{H}$ (TP + FP) is given by[9]:

$$R = \frac{-2\alpha\beta\gamma + \alpha\beta + \beta\gamma + \alpha\gamma - \gamma}{(2\gamma - 1)\alpha(1 - \beta)} \qquad (1)$$

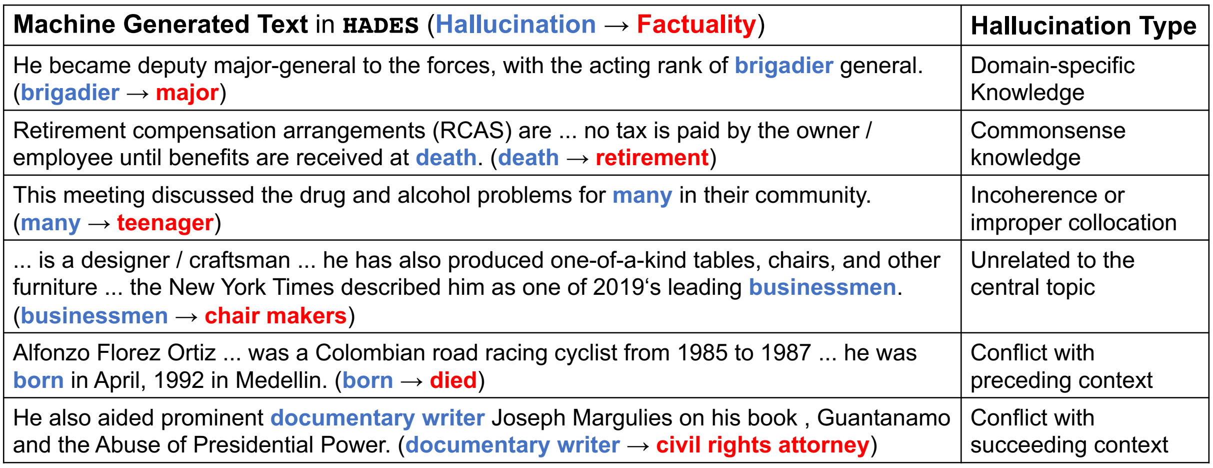
[5] If a worker keeps choosing the same label for all HITs, or the average time spent per HIT is less than 10 seconds, or more than 30% of their judgments conflict with others', we would manually check their annotations and block the spammers.

[6] Many perturbations are *trivial* to predict, *e.g.* replacements that change a specific date to a non-date-related phrase must be a hallucination.

[7] Except the first round, where we use random sampling.

[8] We only remove cases where the replaced date is *definitely* different (e.g., from "Monday" to "Tuesday"). We do not remove ambiguous cases such as from "today" to "Tuesday".

[9] Details are provided in the appendix.



Machine Generated Text in HADES (Hallucination → Factuality)	**Hallucination Type**
He became deputy major-general to the forces, with the acting rank of **brigadier** general. (**brigadier** → **major**)	Domain-specific Knowledge
Retirement compensation arrangements (RCAS) are ... no tax is paid by the owner / employee until benefits are received at **death**. (**death** → **retirement**)	Commonsense knowledge
This meeting discussed the drug and alcohol problems for **many** in their community. (**many** → **teenager**)	Incoherence or improper collocation
... is a designer / craftsman ... he has also produced one-of-a-kind tables, chairs, and other furniture ... the New York Times described him as one of 2019‘s leading **businessmen**. (**businessmen** → **chair makers**)	Unrelated to the central topic
Alfonzo Florez Ortiz ... was a Colombian road racing cyclist from 1985 to 1987 ... he was **born** in April, 1992 in Medellin. (**born** → **died**)	Conflict with preceding context
He also aided prominent **documentary writer** Joseph Margulies on his book , Guantanamo and the Abuse of Presidential Power. (**documentary writer** → **civil rights attorney**)	Conflict with succeeding context

Figure 3: Overview for different types of hallucination in the proposed HADES dataset.

3.4 Data Analysis

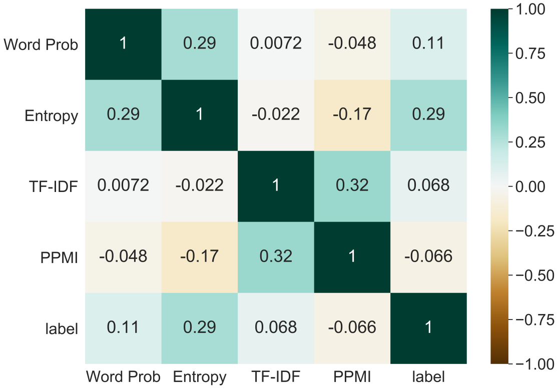
Below we provide data statistics and characterize the composition and properties of HADES.

Data statistics In total, after accumulating annotations for several rounds, we obtain 12,719 instances with 71,226 HITS from judges. We conduct 14 rounds of annotation, increasing the annotation scale with each round (ranging from ∼200 instances/round to ∼4000 instances/round). Out of 12,719 annotated instances, 10,954 instances reached consensus among judges and are included in the HADES dataset. We split the dataset into train, validation and test sets with sizes of 8754, 1000, 1200 respectively. In the final dataset, "hallucination" cases slightly outnumber "not hallucination" cases, with a ratio of 54.5%/45.5%. We summarize some typical hallucination types seen in the HADES dataset in Fig 3.

Parsing features In Fig 4 we show the ratio of "hallucination"($\mathcal{H}$)/ "not hallucination" ($\mathcal{N}$) cases for different Part-of-Speech (POS) and Name Entity Recognition (NER) tags, identified by *Spacy*. From a POS perspective, around two-thirds of verbs and verbal phrases in the dataset are identified as "not hallucination", while in other types of words/phrases, "hallucination" cases are in the majority, e.g., most adverbs (ADV), adjectives (ADJ) and acronyms of proper nouns (PROPN) are labeled as "hallucination". Presumably many verbs or verbal phrases are lower in word concreteness (Nelson and Schreiber, 1992) than other word types (*e.g.* "make" and "create" can be used interchangeably in many circumstances), and thus, as we observe in our dataset, are less prone to be perturbed into hallucinations. For NER tags, about 90% of word spans are not recognized as name entities. However, of the 10% of remaining instances, over 90% are "hallucination" cases.

Label	Word Prob*	Entropy	TF-IDF	PPMI
$\mathcal{H}$	$\mathbf{5.85}_{25.6}$	$\mathbf{2.58}_{1.49}$	$\mathbf{.021}_{.019}$	$.198_{.134}$
$\mathcal{N}$	$1.30_{7.67}$	$1.78_{1.07}$	$.019_{.014}$	$\mathbf{.216}_{.129}$

(A) $\mathbf{Mean}_{\mathbf{std}}$ statistics for **H**allucination ($\mathcal{H}$) and **not H**allucination ($\mathcal{N}$) labels (* indicates $\times 1e^{-8}$).



(B) Feature correlation heatmap between hallucination label and word probability, entropy, TF-TDF and PPMI.

Table 1: Analysis for statistical and model-based features of HADES.

Statistical and model-based features To analyze the characteristics of hallucinations in HADES, we compute the correlation between a selected group of statistical/model-based features and hallucination labels. As shown in Table 1[10] , we obtain the average word probability and average word entropy of a given text span with a BERT base model (without fine-tuning), as well as term frequency–inverse document frequency (TF-IDF),

[10] More statistical feature analysis is in the appendix.

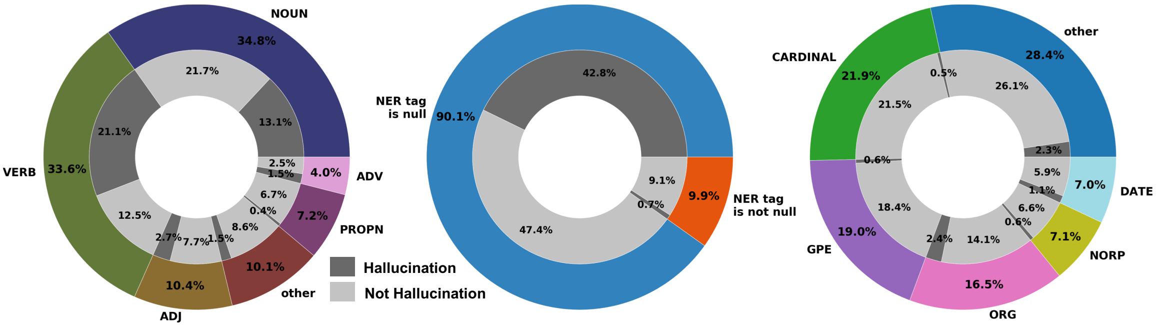


Figure 4: Distributions of POS (left), NER (middle) and a breakdown of non-null NER tags (right) in HADES.

positive pointwise mutual information (PPMI) features of the given word span. By comparing the features of the two labels ($\mathcal{H}/\mathcal{N}$) (Table 1A), we observe that in our dataset, hallucinations typically associate with higher entropy. A counter-intuitive observation is that the hallucinations tend to have higher average probability than factually consistent content. We presume the underlying reason might be that the word distribution generated by machine may diverge from the word distribution of real human-written text (Holtzman et al., 2020; See et al., 2019) owing to self-reinforcing the current generation based on previous generation. Consequently, many overconfident generation outputs are likely to fall into hallucination. We observe no strong correlation between hallucination labels and TF-IDF or PPMI as demonstrated in Table 1B.

4 Baseline Models

As an initial step towards tackling the proposed hallucination detection task and benchmarking methods, we create several baseline detection models[11].

Feature-based models As elaborated in Sec 3.4, the statistical/model-based features like average word probability, average entropy, TF-IDF, PPMI, as well as parsing features like POS and NER tags can be vague indicators of hallucinations. The former two are context-aware and the latter four are not. We incorporate them as features to build classifiers including logistic regression (**LR**) and support vector machine (**SVM**) using *scikit-learn* (Pedregosa et al., 2011). The maximum number of iteration is set as 100, with an early-stop strategy which stops training if the loss does not drop within 5 iterations.

Transformer-based models We also build baseline detection models based on pretrained transformer models including BERT, GPT-2, XLNet (Yang et al., 2019) and RoBERTa (Liu et al., 2020). These transformer-based models represent the state-of-the-art, and can potentially better leverage context or embedded world knowledge to detect self-contradictory or anti-commonsense content.

Specifically, for an input text segment, we fine-tune a pretrained model $\mathcal{M}$ to predict binary hallucination labels $\mathbf{y}$ for each given text span. During inference time, from the last layer hidden states $\mathbf{H} \in \mathbb{R}^{l\times h}$ (h, l are hidden size and sequence length, respectively) of $\mathcal{M}$, suppose the target text span starts at position s and ends at position t, we first obtain the representation $\mathbf{w} \in \mathbb{R}^h$ for the target span with max pooling (*i.e.*, $\mathbf{w} = \mathrm{max\_pool}(\mathbf{H}_{s:t})$). We then map $\mathbf{w}$ to a binary hallucination label $y \in \{0, 1\}$ with a MLP network using $\tanh$ as activation. During training time, we fine-tune the model using cross entropy objective between the predicted labels and the actual labels.

5 Experimental Setup

Baseline configurations For the transformer-based baselines, we experiment with a variety of pretrained models via Hugging Face Transformers (Wolf et al., 2020), including **BERT**-large (335M), **GPT2**-medium (345M), **XLNet**-large (340M), **RoBERTa**-large (355M). We use Adam optimizer (Kingma and Ba, 2015) with different learning rates, i.e. 5e-3 for GPT2 and BERT and 1e-3 for other models.

We explored multiple model architectures and setups to determine the optimal configuration using BERT-large model. These include $i)$ span representation with mean/max pooling ; $ii)$ number of layers of the MLP network; $iii)$ hidden dimension

[11] The proposed token-level, reference-free hallucination detection hasn't been covered in the existing literature. Thus this thread is first-of-its-kind. We are unable to find a feasible baseline that perfectly fits in our setting, therefore we propose multiple feature-based/pretrained baselines.

Model	Acc	G-Mean (↑)	BSS (↓)	AUC	Not Hallucination			Hallucination		
					P	R	F1	P	R	F1
LR	62.25	60.77	-	-	62.35	72.08	66.86	62.10	51.24	60.33
SVM	63.67	61.50	-	-	62.89	76.18	68.90	65.05	49.65	56.31
BERT	71.92	**71.95**	19.06	78.63	**74.46**	71.29	72.84	69.31	**72.61**	70.92
RoBERTa	**72.83**	70.94	**18.78**	78.72	74.06	74.76	74.41	71.43	70.67	**71.05**
XLNet	72.33	71.39	18.79	**78.93**	71.15	**80.13**	**75.37**	**74.07**	63.60	68.44

Table 2: Benchmark (numbers in percentages (%)) for the offline setting on HADES, where detecting models have access to the bidirectional context. ↓/↑ indicates lower/higher is better. Significant tests are in the appendix.

Model	Acc	G-Mean (↑)	BSS (↓)	AUC	Not Hallucination			Hallucination		
					P	R	F1	P	R	F1
GPT-2	**71.58**	**70.98**	19.13	77.71	**71.32**	77.29	**74.19**	**71.93**	**65.19**	**68.40**
BERT	71.00	70.43	**18.66**	**78.83**	70.91	76.50	73.60	71.12	64.84	67.84
RoBERTa	70.67	70.14	19.77	77.07	70.74	75.87	73.22	70.58	64.84	67.59
XLNet	70.08	69.17	19.76	76.59	69.39	**77.60**	73.27	71.08	61.66	66.04

Table 3: Benchmark (numbers in percentages (%)) for the online setting on HADES, where detection models only have the access to left context. ↓/↑ indicates lower/higher is better. Significant tests are in the appendix.

of the MLP ; iv) whether or not to freeze the parameters of $\mathcal{M}$ up to the last layer, and choose the best configuration according to model performance on the validation set. The best configuration uses max-pooling, employs 2 layers of MLP with hidden dimension of $h/2$, and freezes the model parameters up to the last layer of $\mathcal{M}$ and just fine-tunes the binary MLP classifier. We apply the same network configuration to all other pretrained models as empirically we see marginal performance gain after enumerating different configurations for individual pretrained models other than BERT.

As discussed in Sec.2, HADES can serve as benchmark for hallucination detection in both offline (model can see bidirectional context) and online (only preceding context can be leveraged) settings. Note that we apply the feature-based baselines only in the offline setting (Table 2), because a good estimation of those features requires bidirectional context. The transformer with causal attention (GPT-2) can only fit in the online setting.

Evaluation metrics We evaluate the baselines on HaDes with standard classification metrics including accuracy, precision, recall, F1 and AUC (Area Under Curve) with respect to ROC. We also utilize the G-Mean metric which measures geographic mean of sensitivity and specificity (Espíndola and Ebecken, 2005) and they were reported useful especially for the imbalanced label distribution scenarios. We also employ the Brier Skill Score (BSS) metric (Center, 2005), which calculates the mean squared error between the reference distribution and the hypothesis probabilities.

6 Results

Baseline performance Table 3 and Table 2 show the performance of the baseline models [12] in both online and offline settings respectively. In both settings, the predictions for "not hallucination" cases have higher F1 scores than "hallucination" cases. All models perform better in the offline setting compared with the online setting, indicating that the succeeding context of the target words helps identify hallucinations. The transformer-based baselines are generally on par with each other. Under the offline setting, the pretrained models outperform feature-based models by a large margin; this indicates that the powerful contextualized feature extractor is important for successfully identifying hallucinations at fine granularity. Under the online setting, we observe that, for most of the metrics, GPT-2 yields the best performance of all baselines.

[12] To identify the clear winner among baseline models, we report the significant tests for the baseline models in Table 3 and Table 2 as follows: For the offline setting (Table 2), there is no obvious winner among pretrained models, e.g. RoBERTa wins in ACC; XLNet wins in F1 for not hallucination cases; BERT wins in G-mean. For the online setting (Table 3), we ran significant tests for the mean performance (over 5 runs) between GPT-2 and BERT; GPT-2 and XLNet; GPT-2 and RoBERTa, the differences in terms of ACC; G-mean; F1 scores for both hallucination and not hallucination labels are significant (alpha=0.01) after Bonferroni correction.

Context: *as mandated by the statistics act, statistics canada carries out a census of agriculture every five years.*

GPT3 w/ greedy: the census of agriculture is a complete enumeration of all farms in canada , and is used to determine the number of farms in canada , the size of farms , and the value of agricultural production .

GPT3 w/ sampling: these censuses cover a wide range of topics , including agricultural practices , land use , greenhouse gas emissions , farm financial data , structures and buildings , farm machinery , agricultural input use and farm product sales .

Figure 5: The visualization of predicted hallucination scores for a sample of GPT-3 generated text, provided by BERT (large, offline) detector. Darker green signifies higher risk to be hallucinations.

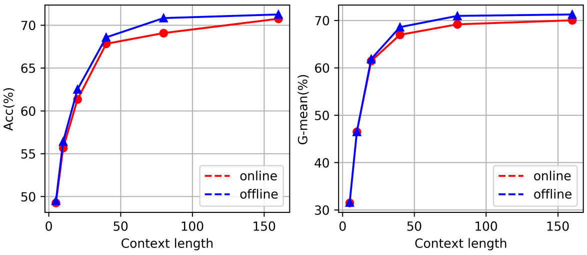


Figure 6: The performance of BERT-large based detecting model with different context lengths.

Presumably, the causal language model pretraining method makes GPT-2 perform better in the auto-aggressive (online) detection setting.

Context matters in HADES To investigate extent to which contextual information helps the hallucination detection in HADES, we run BERT-large detection model with different context lengths and characterize its performance in both online and offline settings in Fig 6. Starting from the target words, we set a fixed size (5/10/20/40/80/160) context window and truncate all text beyond this window. As we enlarge the context window, model performance grows rapidly when context length is smaller than 80, and then gradually converges. This observation highlights the importance of context in hallucination detection. Interestingly, we observe that the model obtains higher performance in the offline mode than in the online setting. The performance gap between the two settings maximizes when context length is around 75, and vanishes with long (> 150) or short (< 20) context windows. We surmise that for long (> 150) context window, the preceding context information might already be adequate for detection, while for short (< 20) context windows, the context, regardless whether it is unidirectional or bidirectional, might not contain enough information for detection.

Model predictions on GPT-3 generated text We visualize the predictions of BERT-large (offline) model on GPT-3 generated text in Fig 5. According to the 2021 census instruments [13], some identified spans like "greenhouse gas emission" and "complete enumeration" are indeed not included in the census, we assume they are recognized due to the topic or knowledge irrelevance with the "census of agriculture" in the pretrained corpus. Interestingly, the detection model predicts the high hallucination risk on "structures and buildings", which has subtle differences with "total greenhouse area including enclosed structures" (included in the instruments). The case study demonstrates the potentials of our model in identifying hallucinated content in the actual outputs of large-scale pretrained models.

7 Related Work

Reference-based Hallucination Detection Apart from human verification (Chen and Bansal, 2018), researchers have developed effective reference-based methods which automatically detect hallucination in the generated text using statistical n-gram matching (Dhingra et al., 2019; Liu et al., 2019), edit distance heuristics (Zhou et al., 2021), natural language inference (Kryscinski et al., 2020; Falke et al., 2019), information extraction (Zhang et al., 2020; Goodrich et al., 2019) or question answering (Scialom et al., 2019; Eyal et al., 2019; Wang et al., 2020a). Our approach differs from them in that we investigate the reference-free hallucination detection scenario.

To reduce hallucinations in the reference-based setting, researchers have applied iterative training (Nie et al., 2019), post editing (Dong et al., 2020),

[13] https://www.statcan.gc.ca/en/statistical-programs/instrument/3438_Q1_V6

soft constraints, e.g. attention manipulation (Kiddon et al., 2016; Hua and Wang, 2019; Tian et al., 2019) or optimal transport (Wang et al., 2020b), and template/scaffold guided schema with explicit plans (Ma et al., 2019; Moryossef et al., 2019; Balakrishnan et al., 2019; Du et al., 2020; Liu et al., 2021), e.g. text sequences which specify the narrative ordering, and implicit plans (Wiseman et al., 2018; Ye et al., 2020; Shen et al., 2020; Li and Rush, 2020), e.g. (structured) hidden variables that corresponds to certain surface realization.

Reference-free Detection Approaches Reference-free hallucination detection is closely related to fake news detection (Zellers et al., 2019; Zhou and Zafarani, 2020; Zhong et al., 2020), which aims to identify deliberate disinformation in a reference-free manner on social media and usually involves common-sense and world knowledge reasoning (Monti et al., 2019), or fact checking (Thorne et al., 2018), where practitioners are asked to verify given claims without references by retrieving related evidence from Wikipedia. Another line of research is to classify sentence-level language specificity (Li and Nenkova, 2015; Gao et al., 2019), which scales from 1 (very general) - 5 (very specific) for short text, e.g. tweets, according to human annotation.

The proposed hallucination detection aims to examine the text in a finer granularity than fake news detection and fact checking. In the proposed task, most parts of the text remain faithful; our goal is to identify subtle hallucinations at the token-level. Fake news detection or specificity assessment, on the other hand, usually focus on sentence- or document-level detection.

8 Conclusions

We have proposed a *token-level reference-free* hallucination detection task and introduced a benchmark dataset HADES for identifying fine granularity hallucination in free-form text generation. To create this dataset, we perturbed texts to simulate hallucination in NLG system, and performed an interative model-in-the-loop annotation approach to annotate the perturbed text in an imbalanced label scenario. We have further provided comprehensive analyses of HADES and evaluated several baseline models to establish initial benchmarks. We hope that the proposed task and dataset will shed light on high-resolution hallucination detection in free-form text generation and will eventually lead to real-time hallucination prevention.

Broader Impact and Ethnic Consideration

This study aims to facilitate the recognition of potential hallucinated content produced by large-scale pretrained models in the free-form generation. We support this goal with a novel *reference-free, token-level* hallucination task and the corresponding annotated dataset HADES. The detection model trained with HADES could be potentially useful in both online and offline settings. For online settings it is possible to guide beam search or suppress the probability of hallucinated tokens through the detection models. For offline settings our system may expedite the human-in-the-loop post-examination in product deployment.

We design our model to detect hallucination to factual statement. The learned knowledge should be able to be transferred to other domain like social chatbot once the chat is regarding certain facts (e.g. a celebrity, a historical event). Wikipedia dataset covers a lot of facts, domains and topics, making it ideal for our study. We thus collect the HADES dataset from Wikipedia. All text on Wikipedia is licensed under the Creative Commons Attribution/Share-Alike 3.0 Unported License. During the annotation, all involved annotators voluntarily participated with decent payment.

Acknowledgments

The authors would like to thank the anonymous reviewers for their thoughtful and constructive comments. Tianyu and Zhifang gratefully acknowledge the support of the National Key Research and Development Program of China 2020AAA0106701 and National Science Foundation of China project U19A2065.

References

Anusha Balakrishnan, Jinfeng Rao, Kartikeya Upasani, Michael White, and Rajen Subba. 2019. Constrained decoding for neural NLG from compositional representations in task-oriented dialogue. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 831–844, Florence, Italy. Association for Computational Linguistics.

Steven Bird. 2006. Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, pages 69–72.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind

Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

NOAA-CIRES Climate Diagnostics Center. 2005. Brier skill scores, rocs, and economic value diagrams can report false skill.

Yen-Chun Chen and Mohit Bansal. 2018. Fast abstractive summarization with reinforce-selected sentence rewriting. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 675–686, Melbourne, Australia. Association for Computational Linguistics.

David A Cohn, Zoubin Ghahramani, and Michael I Jordan. 1996. Active learning with statistical models. *Journal of artificial intelligence research*, 4:129–145.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. 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 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Bhuwan Dhingra, Manaal Faruqui, Ankur Parikh, Ming-Wei Chang, Dipanjan Das, and William Cohen. 2019. Handling divergent reference texts when evaluating table-to-text generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4884–4895, Florence, Italy. Association for Computational Linguistics.

Yue Dong, Shuohang Wang, Zhe Gan, Yu Cheng, Jackie Chi Kit Cheung, and Jingjing Liu. 2020. Multi-fact correction in abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9320–9331, Online. Association for Computational Linguistics.

Yuheng Du, Shereen Oraby, Vittorio Perera, Minmin Shen, Anjali Narayan-Chen, Tagyoung Chung, Anushree Venkatesh, and Dilek Hakkani-Tur. 2020. Schema-guided natural language generation. In *Proceedings of the 13th International Conference on Natural Language Generation*, pages 283–295, Dublin, Ireland. Association for Computational Linguistics.

Rogério P Espíndola and Nelson FF Ebecken. 2005. On extending f-measure and g-mean metrics to multi-class problems. *WIT Transactions on Information and Communication Technologies*, 35.

Matan Eyal, Tal Baumel, and Michael Elhadad. 2019. Question answering as an automatic evaluation metric for news article summarization. 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 3938–3948, Minneapolis, Minnesota. Association for Computational Linguistics.

Tobias Falke, Leonardo F. R. Ribeiro, Prasetya Ajie Utama, Ido Dagan, and Iryna Gurevych. 2019. Ranking generated summaries by correctness: An interesting but challenging application for natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2214–2220, Florence, Italy. Association for Computational Linguistics.

Yifan Gao, Yang Zhong, Daniel Preoţiuc-Pietro, and Junyi Jessy Li. 2019. Predicting and analyzing language specificity in social media posts. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6415–6422.

Ben Goodrich, Vinay Rao, Peter J Liu, and Mohammad Saleh. 2019. Assessing the factual accuracy of generated text. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 166–175.

Mandy Guo, Zihang Dai, Denny Vrandečić, and Rami Al-Rfou. 2020. Wiki-40B: Multilingual language model dataset. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2440–2452, Marseille, France. European Language Resources Association.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Xinyu Hua and Lu Wang. 2019. Sentence-level content planning and style specification for neural text generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 591–602, Hong Kong, China. Association for Computational Linguistics.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2021–2031, Copenhagen, Denmark. Association for Computational Linguistics.

Chloé Kiddon, Luke Zettlemoyer, and Yejin Choi. 2016. Globally coherent text generation with neural checklist models. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 329–339, Austin, Texas. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR (Poster)*.

Wojciech Kryscinski, Bryan McCann, Caiming Xiong, and Richard Socher. 2020. Evaluating the factual consistency of abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9332–9346, Online. Association for Computational Linguistics.

Junyi Li and Ani Nenkova. 2015. Fast and accurate prediction of sentence specificity. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29.

Xiang Lisa Li and Alexander Rush. 2020. Posterior control of blackbox generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2731–2743, Online. Association for Computational Linguistics.

Tianyu Liu, Fuli Luo, Pengcheng Yang, Wei Wu, Baobao Chang, and Zhifang Sui. 2019. Towards comprehensive description generation from factual attribute-value tables. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5985–5996, Florence, Italy. Association for Computational Linguistics.

Tianyu Liu, Xin Zheng, Baobao Chang, and Zhifang Sui. 2021. Towards faithfulness in open domain table-to-text generation from an entity-centric view. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13415–13423.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Roberta: A robustly optimized bert pretraining approach.

Shuming Ma, Pengcheng Yang, Tianyu Liu, Peng Li, Jie Zhou, and Xu Sun. 2019. Key fact as pivot: A two-stage model for low resource table-to-text generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2047–2057, Florence, Italy. Association for Computational Linguistics.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, Online. Association for Computational Linguistics.

Federico Monti, Fabrizio Frasca, Davide Eynard, Damon Mannion, and Michael M Bronstein. 2019. Fake news detection on social media using geometric deep learning. *arXiv preprint arXiv:1902.06673*.

Amit Moryossef, Yoav Goldberg, and Ido Dagan. 2019. Step-by-step: Separating planning from realization in neural data-to-text generation. 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 2267–2277, Minneapolis, Minnesota. Association for Computational Linguistics.

Douglas L Nelson and Thomas A Schreiber. 1992. Word concreteness and word structure as independent determinants of recall. *Journal of memory and language*, 31(2):237–260.

Feng Nie, Jin-Ge Yao, Jinpeng Wang, Rong Pan, and Chin-Yew Lin. 2019. A simple recipe towards reducing hallucination in neural surface realisation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2673–2679, Florence, Italy. Association for Computational Linguistics.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901, Online. Association for Computational Linguistics.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830.

Ofir Press, Noah A Smith, and Mike Lewis. 2020. Shortformer: Better language modeling using shorter inputs. *arXiv preprint arXiv:2012.15832*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Clément Rebuffel, Marco Roberti, Laure Soulier, Geoffrey Scoutheeten, Rossella Cancelliere, and Patrick Gallinari. 2021. Controlling hallucinations at word level in data-to-text generation. *arXiv preprint arXiv:2102.02810*.

Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. 2018. Object hallucination in image captioning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4035–4045, Brussels, Belgium. Association for Computational Linguistics.

Thomas Scialom, Sylvain Lamprier, Benjamin Piwowarski, and Jacopo Staiano. 2019. Answers unite! unsupervised metrics for reinforced summarization models. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3246–3256, Hong Kong, China. Association for Computational Linguistics.

Abigail See, Aneesh Pappu, Rohun Saxena, Akhila Yerukola, and Christopher D. Manning. 2019. Do massively pretrained language models make better storytellers? In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 843–861, Hong Kong, China. Association for Computational Linguistics.

Xiaoyu Shen, Ernie Chang, Hui Su, Cheng Niu, and Dietrich Klakow. 2020. Neural data-to-text generation via jointly learning the segmentation and correspondence. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7155–7165, Online. Association for Computational Linguistics.

James Thorne and Andreas Vlachos. 2018. Automated fact checking: Task formulations, methods and future directions. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3346–3359, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. 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 809–819, New Orleans, Louisiana. Association for Computational Linguistics.

Ran Tian, Shashi Narayan, Thibault Sellam, and Ankur P Parikh. 2019. Sticking to the facts: Confident decoding for faithful data-to-text generation. *arXiv preprint arXiv:1910.08684*.

Alex Wang, Kyunghyun Cho, and Mike Lewis. 2020a. Asking and answering questions to evaluate the factual consistency of summaries. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5008–5020, Online. Association for Computational Linguistics.

Chaojun Wang and Rico Sennrich. 2020. On exposure bias, hallucination and domain shift in neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3544–3552, Online. Association for Computational Linguistics.

Zhenyi Wang, Xiaoyang Wang, Bang An, Dong Yu, and Changyou Chen. 2020b. Towards faithful neural table-to-text generation with content-matching constraints. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1072–1086, Online. Association for Computational Linguistics.

Sam Wiseman, Stuart Shieber, and Alexander Rush. 2018. Learning neural templates for text generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3174–3187, Brussels, Belgium. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi 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 Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Rong Ye, Wenxian Shi, Hao Zhou, Zhongyu Wei, and Lei Li. 2020. Variational template machine for data-to-text generation. In *International Conference on Learning Representations*.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. SWAG: A large-scale adversarial dataset for grounded commonsense inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 93–104, Brussels, Belgium. Association for Computational Linguistics.

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. Defending against neural fake news. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 9054–9065. Curran Associates, Inc.

Yuhao Zhang, Derek Merck, Emily Tsai, Christopher D. Manning, and Curtis Langlotz. 2020. Optimizing the factual correctness of a summary: A study of summarizing radiology reports. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5108–5120, Online. Association for Computational Linguistics.

Wanjun Zhong, Duyu Tang, Zenan Xu, Ruize Wang, Nan Duan, Ming Zhou, Jiahai Wang, and Jian Yin. 2020. Neural deepfake detection with factual structure of text. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2461–2470, Online. Association for Computational Linguistics.

Chunting Zhou, Graham Neubig, Jiatao Gu, Mona Diab, Francisco Guzmán, Luke Zettlemoyer, and Marjan Ghazvininejad. 2021. Detecting hallucinated content in conditional neural sequence generation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1393–1404, Online. Association for Computational Linguistics.

Xinyi Zhou and Reza Zafarani. 2020. A survey of fake news: Fundamental theories, detection methods, and opportunities. *ACM Computing Surveys (CSUR)*, 53(5):1–40.

A Detailed Statistical Analysis

In Table 4, we provide detailed statistical analyses for different POS and NER tags in the HADES dataset. Although the average word probability and average word entropy features differ among POS/NER tags, hallucinated content typically associates with higher word probability and word entropy irrespective of POS/NER tag. Strong correlation between hallucination labels and TF-IDF or PPMI features is not observed.

B Annotation Interface

The annotation interface is provided in Fig 7.

Note that throughout the annotation process we choose to involve an even number of, e.g. 4 or 6, annotators (Sec 3.3) for an instance. The reason is that, we manage to involve extra annotators for controversial cases. If we pick an odd number of, e.g. 5 rather than 4, annotators, for each datapoint (binary classification) all possible results would be 0:5/1:4/2:3/3:2/4:1/5:0 in terms of the ratio of Hallucination/Consistent labels, which means no more annotators would be involved as they always reach consensus (majority wins).

C Subsampling Ratio For Label Rebalance

We adopt an iterative model-in-the-loop method in data annotation. Since observe a label imbalance between "hallucination" ($\mathcal{H}$) and "not hallucination" ($\mathcal{N}$) in the initial rounds of annotation, we employ subsampling to rebalance the label distribution in Sec 3.3. We accumulate the data annotated in the all previous rounds, and train a detection model using the accumulated data. Then we apply the detection model to the unannotated data in the candidate data pool in order to select next batch of data as elaborated in Sec 3.3.

We assume that the human annotation for $\mathcal{H}$ and $\mathcal{N}$ cases is the oracle, indicating actual $\mathcal{H}/\mathcal{N}$. Since the actual "hallucinated" is dominant, we try to subsample from the instances that are predicted by the detection model to be $\mathcal{H}$, in order to even out the distribution of actual $\mathcal{H}/\mathcal{N}$. To do this, we estimate the true positive rate[14] (TPR, α), true negative rate (TNR, β) and true precision (γ) of the detection model based on the annotations from the previous rounds.

$$\text{TPR} = \frac{\text{TP}}{(\text{TP}+\text{FN})} \triangleq \alpha \tag{2}$$

$$\text{TNR} = \frac{\text{TN}}{(\text{TN}+\text{FP})} \triangleq \beta \tag{3}$$

$$\text{precision} = \frac{\text{TP}}{(\text{TP}+\text{FP})} \triangleq \gamma \tag{4}$$

Where TP, FP, TN, FN are the abbreviations of "true positive", "false positive", "true negative" and "false negative" cases. We aim to subsample from the instances that are predicted as $\mathcal{H}$ from the detection model (TP + FP) with a subsampling ratio s, so that the actual $\mathcal{H}$ (TP + FN) is roughly equal to actual $\mathcal{N}$ (FP + TN) after the resampling. We denote TP and TN as x and y and represent FN and FP with $x, y, \alpha, \gamma, \beta$:

$$\text{FN} = \frac{1-\alpha}{\alpha}x \tag{5}$$

$$\text{FP} = \frac{1-\beta}{\beta}y \tag{6}$$

By substituting FN, FP into Eq. (4), we have:

$$\gamma = \frac{x}{x + \frac{1-\beta}{\beta}y} \tag{7}$$

To make the distribution of actual $\mathcal{H}/\mathcal{N}$ even (sTP+FN=sFP+TN), we have:

$$sx + \frac{1-\alpha}{\alpha}x = s\frac{1-\beta}{\beta}y + y \tag{8}$$

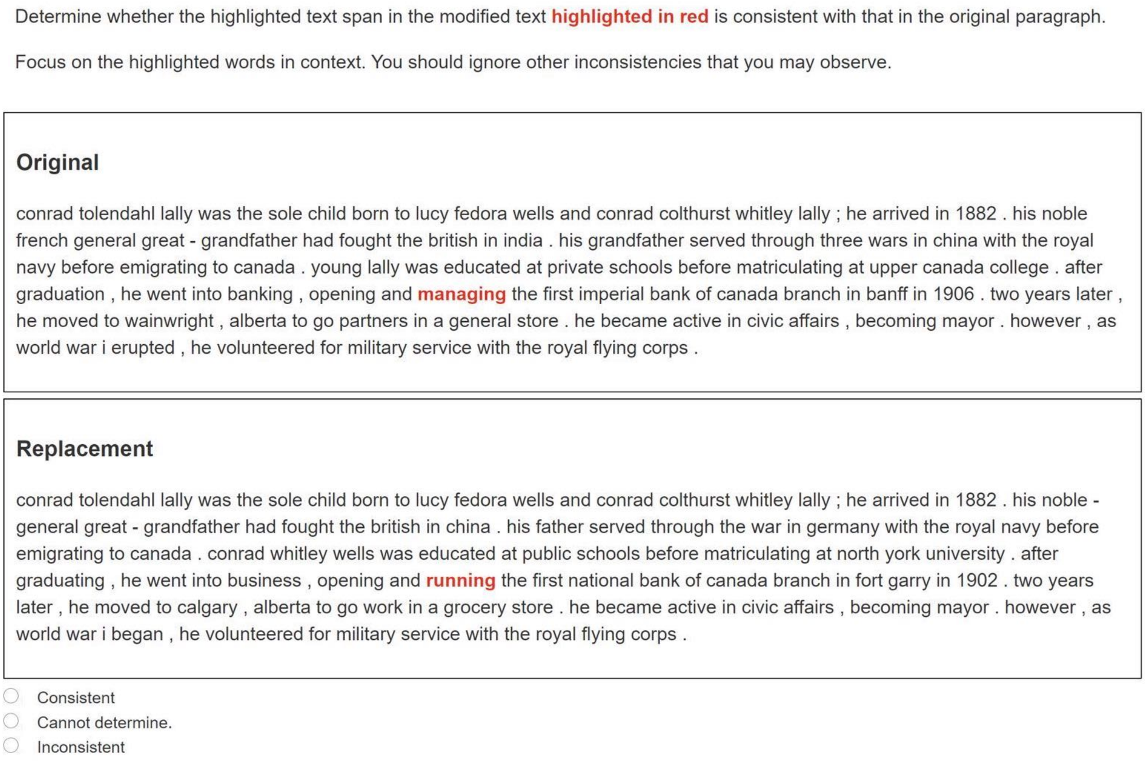
By combining Eq. (7) and Eq. (8), we figure out the optimal subsampling ratio s^*.

$$s^* = \frac{-2\alpha\beta\gamma + \alpha\beta + \beta\gamma + \alpha\gamma - \gamma}{(2\gamma - 1)\alpha(1-\beta)} \tag{9}$$

[14] Defining $\mathcal{H}$ as the positive class.

Tag	Word Prob($\times 1e^{-8}$)		Entropy		TF-IDF		PPMI	
	$\mathcal{H}$	$\mathcal{N}$	$\mathcal{H}$	$\mathcal{N}$	$\mathcal{H}$	$\mathcal{N}$	$\mathcal{H}$	$\mathcal{N}$
POS:NOUN	$\mathbf{6.98}_{32.0}$	$\mathbf{1.68}_{6.34}$	$\mathbf{2.75}_{1.52}$	$\mathbf{1.86}_{1.13}$	$\mathbf{.025}_{.021}$	$\mathbf{.023}_{.018}$	$\mathbf{.213}_{.145}$	$\mathbf{.228}_{.140}$
POS:VERB	$\mathbf{2.51}_{9.33}$	$\mathbf{0.69}_{2.89}$	$\mathbf{2.25}_{1.25}$	$\mathbf{1.76}_{1.00}$	$\mathbf{.019}_{.012}$	$\mathbf{.018}_{.011}$	$\mathbf{.206}_{.112}$	$\mathbf{.216}_{.119}$
POS:ADJ	$\mathbf{8.16}_{44.8}$	$\mathbf{2.86}_{18.9}$	$\mathbf{2.95}_{1.46}$	$\mathbf{2.38}_{1.23}$	$\mathbf{.021}_{.017}$	$\mathbf{.017}_{.009}$	$\mathbf{.180}_{.128}$	$\mathbf{.164}_{.117}$
POS:ADV	$\mathbf{5.13}_{14.2}$	$\mathbf{2.65}_{12.2}$	$\mathbf{2.56}_{1.18}$	$\mathbf{1.97}_{1.09}$	$\mathbf{.016}_{.011}$	$\mathbf{.014}_{.008}$	$\mathbf{.181}_{.114}$	$\mathbf{.182}_{.105}$
POS:PROPN	$\mathbf{14.3}_{33.6}$	$\mathbf{4.35}_{17.8}$	$\mathbf{3.12}_{1.73}$	$\mathbf{1.56}_{1.39}$	$\mathbf{.029}_{.026}$	$\mathbf{.033}_{.029}$	$\mathbf{.198}_{.150}$	$\mathbf{.312}_{.275}$
POS:other	$\mathbf{9.56}_{31.1}$	$\mathbf{3.28}_{15.7}$	$\mathbf{2.64}_{1.61}$	$\mathbf{1.26}_{0.97}$	$\mathbf{.013}_{.013}$	$\mathbf{.011}_{.010}$	$\mathbf{.158}_{.107}$	$\mathbf{.205}_{.092}$
NER:null	$\mathbf{5.37}_{25.6}$	$\mathbf{1.24}_{7.19}$	$\mathbf{2.52}_{1.47}$	$\mathbf{1.79}_{1.06}$	$\mathbf{.021}_{.019}$	$\mathbf{.019}_{.014}$	$\mathbf{.200}_{.132}$	$\mathbf{.215}_{.126}$
NER:other	$\mathbf{8.43}_{25.4}$	$\mathbf{5.06}_{21.5}$	$\mathbf{2.93}_{1.56}$	$\mathbf{1.65}_{1.44}$	$\mathbf{.023}_{.023}$	$\mathbf{.026}_{0.024}$	$\mathbf{.189}_{.146}$	$\mathbf{.263}_{.237}$
All	$\mathbf{5.85}_{25.6}$	$\mathbf{1.30}_{7.67}$	$\mathbf{2.58}_{1.49}$	$\mathbf{1.78}_{1.07}$	$\mathbf{.021}_{.019}$	$\mathbf{.019}_{0.014}$	$\mathbf{.198}_{.144}$	$\mathbf{.216}_{.129}$

Table 4: Detailed statistical features ($\mathbf{Mean_{std}}$) for "hallucinated" ($\mathcal{H}$) and "not hallucinated" ($\mathcal{N}$) cases.



Determine whether the highlighted text span in the modified text **highlighted in red** is consistent with that in the original paragraph.

Focus on the highlighted words in context. You should ignore other inconsistencies that you may observe.

Original

conrad tolendahl lally was the sole child born to lucy fedora wells and conrad colthurst whitley lally ; he arrived in 1882 . his noble french general great - grandfather had fought the british in india . his grandfather served through three wars in china with the royal navy before emigrating to canada . young lally was educated at private schools before matriculating at upper canada college . after graduation , he went into banking , opening and **managing** the first imperial bank of canada branch in banff in 1906 . two years later , he moved to wainwright , alberta to go partners in a general store . he became active in civic affairs , becoming mayor . however , as world war i erupted , he volunteered for military service with the royal flying corps .

Replacement

conrad tolendahl lally was the sole child born to lucy fedora wells and conrad colthurst whitley lally ; he arrived in 1882 . his noble - general great - grandfather had fought the british in china . his father served through the war in germany with the royal navy before emigrating to canada . conrad whitley wells was educated at public schools before matriculating at north york university . after graduating , he went into business , opening and **running** the first national bank of canada branch in fort garry in 1902 . two years later , he moved to calgary , alberta to go work in a grocery store . he became active in civic affairs , becoming mayor . however , as world war i began , he volunteered for military service with the royal flying corps .

○ Consistent
○ Cannot determine.
○ Inconsistent

Figure 7: The annotation interface for the proposed hallucination detection task.