

HALLU-NLI: Revisiting Natural Language Inference (NLI) Hallucination Detection Methods for LLM-Generated Biographies

Stanford CS224N Custom Project

Sarah Dong* **Sally Lee*** **Nathan Lim***
sarahjd@stanford.edu sunminl@stanford.edu ntlm77@stanford.edu

*Department of Computer Science, Stanford University

Abstract

Hallucinations, which are fluent but factually incorrect statements, remain a major obstacle to deploying large language models (LLMs) in real-world applications. Recent work such as SelfCheckGPT [1] proposes detecting hallucinations by measuring self-consistency across multiple stochastic generations of the same prompt, using heuristic aggregation of contradiction scores computed by a natural language inference (NLI) model. In this project, we investigate whether extracting contradiction-based features from a pretrained NLI model and training a logistic regression classifier to predict hallucination probabilities can improve performance on the WikiBio hallucination dataset. However, the model achieves performance comparable to the baseline (AUC-PR = 0.7471 vs. 0.7373). To understand this limitation and explore possible improvements, we attempt minor and three major extensions inspired by recent work on hybrid hallucination detection pipelines. These include bidirectional NLI features that capture asymmetric inconsistencies, semantic disagreement features that measure variation of NLI predictions, confidence-Weighted NLI aggregation, and NER-NLI hybrid approach that replaces stochastic self-comparisons with reference-based verification. All methods are implemented within the same feature-based training framework to allow direct comparison. Results show that the simple mean contradiction aggregation technique is a strong baseline that is difficult to surpass using feature extensions and logistic regression alone. However, incorporating NER-based reference verification with external knowledge leads to meaningful improvements.

1 Key Information to include

- TA mentor: Luke Bailey
- External collaborators: No; External mentor: No; Sharing project: No; Late days used: 2 ¹

2 Introduction

Large language models (LLMs) can generate fluent text but often produce hallucinations, which are statements that appear plausible yet are factually incorrect. This limits their reliability in applications such as question answering and summarization, where factual accuracy is critical. Detecting hallucinations is challenging because generated text may appear linguistically valid even when it contradicts source information or world knowledge.

One common approach uses natural language inference (NLI) models to measure logical consistency between generated statements and supporting context. Recent work such as SelfCheckGPT [1] leverages self-consistency across multiple stochastic generations, computing contradiction scores

¹Sarah (6 late days) + Sally (6 late days) + Nathan (3 late days) = 15 total late days for team size of 3
Late days applied for the report: 2

between a sentence and sampled generations and aggregating them with a heuristic rule. While effective as a strong baseline, this approach does not learn how to combine NLI signals from data.

In this project, we investigate whether hallucination detection can be improved by replacing the heuristic aggregation used in SelfCheckGPT-NLI with supervised models over NLI outputs. Our initial approach, HALLU-NLI, trains a classifier on statistics derived from contradiction scores and shows modest improvement over the baseline. We then explore several extensions within this framework, including bidirectional NLI with semantic disagreement features, confidence-weighted NLI aggregation, and a NER-NLI hybrid approach that replaces stochastic self-comparisons with reference-based verification. These experiments evaluate whether combining consistency, uncertainty, and reference-based signals improves hallucination detection beyond simple heuristic aggregation.

3 Related Work

SelfCheckGPT [1] proposes a self-consistency approach for hallucination detection by generating multiple stochastic outputs from the same prompt and measuring agreement between them. In its NLI-based variant, contradiction scores are computed between a generated sentence and sampled passages and aggregated using fixed heuristic rules to estimate hallucination probability. While effective and computationally practical, this approach relies on heuristic aggregation rather than learning how to combine NLI signals. Prior work on factual consistency evaluation similarly shows that entailment and contradiction scores from pretrained NLI models contain useful signals for detecting hallucinations, but performance depends strongly on how these scores are aggregated [2].

More recent work explores stronger signals for hallucination detection. Raghuvanshi et al. [3] combine token level uncertainty, bidirectional NLI, and semantic entropy to capture disagreement across generations, showing that hallucinated statements often exhibit asymmetric predictions and higher uncertainty. Other work incorporates uncertainty more directly. Dey et al. [4] propose an uncertainty-aware ensemble that weights predictions by confidence, while Varshney et al. [5] detect hallucinations by identifying low-confidence generations during decoding and validating them.

Another line of work performs reference-based verification by decomposing generated text into atomic claims that can be independently checked against external evidence [6]. NLI models are frequently used in this setting because entailment and contradiction probabilities correlate with factual correctness [7]. Unlike self-consistency approaches like SelfCheckGPT, which compare multiple generations, this directly evaluates whether specific claims are supported by reference evidence. Despite these advances, it remains unclear how much these signals can improve hallucination detection within simple NLI-based pipelines. Motivated by these ideas, we explore several extensions to the SelfCheckGPT NLI approach, investigating whether richer NLI-based signals and uncertainty cues can improve feature-based hallucination detection.

4 Approach

4.1 Problem Setup

Given a generated sentence s from a language model and a set of stochastic generations $\{g_1, g_2, \dots, g_k\}$ produced from the same prompt, the goal is to estimate the probability that s is hallucinated. Following SelfCheckGPT [1], we use a pretrained natural language inference (NLI) model to measure logical consistency between s and each sampled passage g_i .

For each pair (g_i, s) , the NLI model outputs probabilities for entailment, contradiction, and neutral: P_e, P_c, P_n . To focus on factual consistency, we apply entailment-contradiction renormalization by discarding the neutral class:

$$C_i = \frac{P_c}{P_c + P_e} \quad (1)$$

For each sentence, we obtain a set of contradiction scores $\{C_1, C_2, \dots, C_k\}$ which are used to construct features for hallucination detection.

4.2 Baseline: SelfCheckGPT-NLI

Our baseline follows the NLI-based method in SelfCheckGPT [1]. The original method computes contradiction probabilities between the generated sentence and multiple sampled passages and

aggregates them using a fixed heuristic:

$$\hat{g} = \frac{1}{k} \sum_{i=1}^k C_i$$

where $\hat{g}$ is the hallucination score.

We re-run the SelfCheckGPT-NLI implementation using the authors’ released code on the same WikiBio dataset to ensure a fair comparison.

4.3 HALLU-NLI with Supervised Aggregation of NLI Features

Instead of using a fixed aggregation rule, we train a supervised classifier to combine statistics derived from contradiction scores.

For each sentence, we compute the following summary statistics: mean contradiction, maximum contradiction, standard deviation, fraction of scores > 0.5 , entropy, interquartile range (IQR), and mean of top-3 contradiction scores

This produces a feature vector $x \in \mathbb{R}^7$. Then, we train a logistic regression model

$$p(y = 1 \mid x) = \sigma(w^T x + b)$$

where $y = 1$ denotes hallucination and σ is the sigmoid function.

All feature extraction and supervised modeling code were implemented by us, while the dataset and NLI scoring framework were taken from the SelfCheckGPT repository [1].

We first explored several minor modifications to the supervised aggregation model.

Regularization sweep We varied the logistic regression regularization parameter C to test whether performance was limited by under- or over-regularization:

$$C \in \{0.01, 0.1, 0.5, 1, 2, 5, 10\}$$

Split-by-document evaluation The WikiBio dataset contains multiple sentences from the same generated biography. To avoid information leakage between training and test sets, we evaluate a split-by-document setting, where all sentences from the same biography are assigned to the same split. We also explored more substantial modifications inspired by recent work on hallucination detection.

4.4 HALLU-NLI with Bidirectional NLI and Semantic Disagreement

Inspired by Raghuvanshi et al. [3], we build a new feature vector that includes some similar statistics from the initial approach and some additional signals based on asymmetry and disagreement in NLI predictions.

Bidirectional NLI In addition to the forward comparison $g_i \rightarrow s$, we also compute the reverse comparison $s \rightarrow g_i$, obtain contradiction probabilities C_i^{fwd}, C_i^{rev} , and define the per-sample asymmetry $D_i = |C_i^{fwd} - C_i^{rev}|$.

Then, we have $\text{mean}(C^{rev})$, $\text{std}(C^{rev})$, and asymmetry measures

$$\frac{1}{k} \sum_i D_i, \quad |\text{mean}(C^{fwd}) - \text{mean}(C^{rev})|$$

These statistics are appended to the feature vector.

Semantic disagreement The initial feature set already measures variation in contradiction scores, but only along a single scalar dimension. To capture richer disagreement patterns, we consider the joint NLI predictions across sampled passages.

For each sample we form a point (E_i, C_i) where E_i and C_i are the entailment and contradiction probabilities.

We cluster these points using k-means and compute the entropy of the cluster distribution to measure disagreement, where p_j is the proportion of samples in cluster j (Eq. 2). We also compute entropy directly from contradiction probabilities (Eq. 3).

$$H = - \sum_j p_j \log p_j \quad (2) \quad H_C = - \frac{1}{k} \sum_i (C_i \log C_i + (1 - C_i) \log(1 - C_i)) \quad (3)$$

These disagreement features are concatenated with the original feature vector before training the classifier.

4.5 HALLU-NLI with Confidence-Weighted NLI Aggregation

Motivated by recent work on uncertainty-aware aggregation [4], we extend the supervised NLI feature framework by incorporating confidence-weighted aggregation of NLI predictions.

The SelfCheckGPT-NLI baseline assigns hallucination scores by averaging contradiction probabilities across sampled passages, treating all NLI comparisons equally. To account for varying prediction confidence, we compute confidence-weighted averages of contradiction and entailment probabilities.

For each comparison between a sentence and a sampled passage, we define the confidence weight as

$$w_i = \max(E_i, C_i),$$

where E_i and C_i denote the entailment and contradiction probabilities produced by the NLI model.

Using these weights, we then compute confidence-weighted averages

$$\tilde{C} = \frac{\sum_{i=1}^k w_i C_i}{\sum_{i=1}^k w_i}, \quad \tilde{E} = \frac{\sum_{i=1}^k w_i E_i}{\sum_{i=1}^k w_i}.$$

The final feature vector extends the 7-dimensional feature set described in Section 4.3 by adding two confidence-weighted aggregates: the weighted mean contradiction score $\tilde{C}$ and the weighted mean entailment score $\tilde{E}$. The resulting 9-dimensional feature vector is used to train the same logistic regression classifier for hallucination detection.

4.6 HALLU-NLI with NER-NLI Hybrid

In order to expand upon the original SelfCheckGPT-NLI baseline which uses a reference-free approach. We propose an additional reference-based verification method. This hybrid approach breaks down GPT-3 generated sentences into atomic verifiable claims, which are then scored independently against the ground-truth Wikipedia reference in the WikiBio hallucination dataset [1]. Our HALLU-NLI with NER-NLI Hybrid approach can be broken down into the four main steps below:

Atomic Claim Extraction Using a spaCy English NLP pipeline, we perform joint named entity recognition (NER) and dependency parsing for each generated sentence, allowing us to isolate checkable facts and reduce irrelevant noise for the NLI model. We identify three key units: root verb, subject subtree, and auxiliary verb phrase. To extract atomic claims, we combine the subject and verb phrase with each root modifier subtree containing a named entity. For example, “John Doe was born in 1982 in London” yields “John Doe was born in 1982” (DATE) and “John Doe was born in London” (GPE). Full sentences are treated as fallback claims if entities only appear in subject or sentence has no entities.

NLI Verification Claim-Level Each claim c_i is evaluated against the Wikipedia reference text R (wiki_bio_text) using DeBERTa-v3-base-mnli. We retain the full softmax distribution over Entailment (P_E), Neutral (P_N), and Contradiction (P_C), instead of applying entailment–contradiction renormalization like in original SelfCheckGPT. Claims are scored in a single batched forward pass, with results cached by (R, c_i) .

Entity-Weighted Aggregation Each claim gets weight $W_i \in \{1.0, 1.2\}$, where $W_i = 1.2$ if the claim contains a named entity of type DATE, GPE, LOC, PERSON, ORG, NORP, or FAC, and $W_i = 1.0$ otherwise. The rule-based sentence score is

$$S_{\text{rule}} = \max_i (P_{C,i} \cdot W_i)$$

Neutral-Aware Calibration Because *minor_inaccurate* labels often yield neutral NLI predictions, we incorporate neutral probability into the final score

$$S_{\text{cal}} = \max_i [(P_{C,i} + \gamma P_{N,i}) \cdot W_i]$$

where $\gamma = 0.5$, thereby preventing us from treating uncertain signals as definitive hallucinations.

5 Experiments

5.1 Data

We evaluate our methods on the WikiBio hallucination dataset released with SelfCheckGPT [1]. The dataset contains GPT-generated biographies with sentence-level factuality annotations. Each example includes a generated biography, multiple stochastic generations of the same prompt, and human labels indicating whether each sentence is *accurate*, *minor inaccurate*, or *major inaccurate*.

Following prior work, we convert this into a binary hallucination detection task by mapping *accurate* to label 0 (factual) and both *minor inaccurate* and *major inaccurate* to label 1 (hallucinated). Each sentence is treated as an individual example.

For each sentence s , the input consists of the sentence and a set of sampled passages $\{g_1, \dots, g_k\}$ generated from the same prompt. These passages are used to compute NLI scores between s and each sampled generation, and the model outputs the probability that s is hallucinated.

We use the official dataset split and also evaluate a 70/30 train–test split at both the sentence level and document level. In the document-level split, all sentences from the same biography are assigned to the same split to prevent information leakage.

5.2 Evaluation method

We evaluate hallucination detection using the Area Under the Precision–Recall Curve (AUC-PR), following SelfCheckGPT [1]. The task is a binary classification problem with class imbalance, where hallucinated sentences are less frequent than factual ones. In this setting, AUC-PR is preferred over ROC-AUC because it focuses on performance for the positive class (hallucinated sentences).

For each sentence, the model outputs a hallucination probability. We compute the precision–recall curve by varying the decision threshold and report the area under this curve.

All methods are compared against the appropriate SelfCheckGPT-NLI baseline method, which uses the mean contradiction score across sampled passages as a heuristic hallucination score. To ensure fair comparison, the baseline is evaluated on the same dataset splits as our models. When creating train–test splits, we preserve class distributions using stratified sampling, and for document-level splits, stratification is performed at the document level.

5.3 Experimental details

All experiments use the pretrained NLI model microsoft/deberta-v3-base (used through the SelfCheckGPT SelfCheckNLI pipeline), rather than the default deberta-v3-large-mnli, to reduce runtime and make repeated sweeps/evaluations feasible on CPU. NLI scores are computed with the same SelfCheckNLI procedure (1). In all experiments, we fix the test split configuration (`test_size=0.3`, `seed=42`) and report AUC-PR as the primary metric. Also, we use batch size 32 for NLI inference and run on GPU when available. Feature extraction results are cached to disk to ensure reproducibility across runs. Random seed is fixed to 42 for all experiments.

Supervised Aggregation of NLI Features: As aforementioned, we extract seven summary statistics to form a 7-dimensional vector for each sentence. We train a logistic regression classifier using the scikit-learn implementation with L2 regularization, solver `lbfgs`, maximum iterations set to 1000, and regularization parameter $C = 1.0$.

Bidirectional NLI and Semantic Disagreement: For the hybrid approach based on bidirectional NLI and semantic disagreement, we build a new feature vector with statistics computed from both forward and reverse NLI predictions as aforementioned. We then train the same downstream classifier family (logistic regression) to isolate feature-design effects from classifier effects.

Confidence-Weighted NLI Aggregation For this extension, we use the same pretrained NLI model and sampling setup as the baseline. For each sentence and its sampled passages, we compute NLI entailment and contradiction scores using the SelfCheckNLI pipeline and construct an expanded feature vector including the baseline mean contradiction score, contradiction distribution statistics, entropy-based uncertainty features, and confidence-weighted aggregates. These features are used with the same logistic regression classifier described earlier. Models are trained using the same train-validation split as the baseline, and performance is evaluated using AUC-PR.

NER-NLI Hybrid: We use a reference-augmented setup where WikiBio `wiki_bio_text` (the subject’s Wikipedia lead) serves as the NLI premise. Generated sentences are decomposed and scored using approach 4.6, giving us final sentence score of

$$S_{\text{cal}} = \max_i [(P_{C,i} + \gamma P_{N,i}) \cdot W_i]$$

with $\gamma = 0.5$ mapping neutral predictions toward hallucination. We also include a non-calibrated ($\max_i (P_{C,i} \cdot W_i)$) version for comparison.

5.4 Results

Method	AUC-PR
Baseline (SelfCheckGPT-NLI)	0.7373
Self-Reference Methods	
Logistic Regression Trained (no extensions)	0.7471
Logistic Regression Bidirectional	0.7267
Logistic Regression Confidence-Weighted NLI Aggregation	0.7024
Reference-Based Methods	
Rule Non-calibrated (NER-NLI)	0.8714
Calibrated (NER-NLI)	0.8752

Table 1: Comparison of self-reference and reference-based hallucination detection methods using AUC-PR

Supervised Aggregation of NLI Features: The trained model slightly improves over the baseline, but the improvement is only by 0.0098 (Table 1). This result is consistent with our expectation that replacing the heuristic aggregation with a learned classifier may provide some benefit, but the contradiction statistics alone may not contain enough additional information to produce a large performance gain.

Increasing the regularization parameter C (which corresponds to weaker L2 regularization) leads to a gradual improvement in AUC-PR (Table 2). We have the highest AUC-PR of 0.7682, which is 0.0309 improvement over the baseline. This suggests that strong regularization causes slight underfitting, and allowing the model to use larger weights helps it better fit the training data.

C	AUC-PR
0.01	0.7346
0.1	0.7462
0.5	0.7461
1	0.7471
2	0.7510
5	0.7628
10	0.7682

Table 2: Effect of regularization parameter C

Method	AUC-PR
Baseline (split-doc)	0.7205
Trained (split-doc)	0.7716

Table 3: Split-by-document results

With the split-by-document setting, the baseline performance decreases while the trained model shows higher AUC-PR, which is 0.0511 greater than baseline’s (Table 3). The drop in the baseline is expected, since this split removes similarities between sentences from the same biography that may make the heuristic contradiction score appear more reliable.

Bidirectional NLI and Semantic Disagreement:: The hybrid feature model achieves an AUC-PR of 0.7267, which is lower than both the trained model from the initial approach and the original

SelfCheckGPT baseline (Table 1). This result is unexpected, since the hybrid method incorporates bidirectional NLI and semantic disagreement features that were intended to provide richer consistency and uncertainty signals than the simple contradiction-based statistics used in the initial feature set.

Confidence-Weighted NLI Aggregation: The confidence-weighted NLI aggregation model achieves an AUC-PR of 0.7024, which is lower than the SelfCheckGPT-NLI baseline score of 0.7373. Although incorporating model confidence and uncertainty was intended to improve the aggregation of NLI predictions, the result suggests that confidence-weighted features alone may not provide a stronger signal than the original contradiction-based heuristic. This suggests that the baseline mean contradiction score may already capture much of the useful signal in the NLI outputs.

NER-NLI Hybrid Our calibrated method achieves an AUC-PR of 0.8752, representing a +17.4% improvement over the SelfCheckGPT-NLI baseline (0.7373). This is expected, since our NER-NLI pipeline additionally uses the Wikipedia reference text, ensuring our method operates under different information access assumptions than the SelfCheckGPT-NLI baseline. Thus, the +17.4% improvement highlights the quantitative improvement of incorporating reference-based verification over self-consistency alone, indicating that relying on ground truth references gives stronger signals than simply comparing to sample comparisons. Moreover, the calibrated version outperforms the non-calibrated version in AUC-PR by around 0.0038, indicating modeling neutral predictions as partial evidence rather than excluding it improves detection performance.

6 Analysis

From the results of the initial approach, we observe that training a logistic regression classifier over NLI-based features produces only a small improvement over the original SelfCheckGPT heuristic. Sweeping the regularization parameter C changes the performance slightly, but does not lead to a substantial increase in AUC-PR, which suggests that the main limitation is not the classifier itself, but the features used as input to the classifier. The largest improvement in performance in self-reference methods appears when using the split-by-document setting. This indicates that sentence-level splitting may introduce similarities between training and test examples that make the heuristic baseline appear stronger than it actually is. When these similarities are removed, the trained classifier can better use the feature statistics to separate factual and hallucinated sentences. This suggests that data splitting has a significant impact on the measured performance and that the dataset structure plays an important role in this task.

Adding bidirectional NLI and semantic disagreement features unexpectedly performs worse than both the baseline and the initial trained model. Since these features were designed to provide richer consistency and uncertainty signals, we expected them to improve performance. One possible explanation is that the additional features increase the dimensionality and variability of the feature space, making it harder for the logistic regression classifier to learn a stable decision boundary given the limited size of the dataset. This result suggests that simply adding more features does not necessarily improve hallucination detection, and that the effectiveness of feature-based methods may depend on both feature quality and the amount of training data available.

A similar pattern is observed for the confidence-weighted NLI aggregation method. If the NLI model produces similar confidence levels for both factual and hallucinated sentences, confidence-weighting provides little additional information beyond the original contradiction scores. Additionally, the expanded feature set increases model dimensionality without introducing strongly discriminative signals. Given the relatively small dataset, the logistic regression classifier may struggle to reliably estimate useful feature weights. These results indicate that simple feature-based extensions may be insufficient to outperform the strong heuristic aggregation used by SelfCheckGPT.

Our improved NER-NLI hybrid performance in comparison to the baseline highlights the strengths of an atomic claim reference-based verification approach. By decomposing generated sentences into smaller verifiable claims using NER, NLI models can better evaluate factual consistency at a finer granularity, reducing the effects of noise from irrelevant details and allowing the system to focus on entity inconsistencies. This appears particularly beneficial for detecting entity-level inconsistencies, which are common sources of hallucination, thereby showcasing that NLI alone is not as powerful as NER with NLI combined. In addition, based on Figure 1, NER-NLI methods maintain higher precision-recall curves with slower declines than the baseline, showcasing better distinction between hallucinated vs non-hallucinated sentences across recall levels. This suggests that breaking down

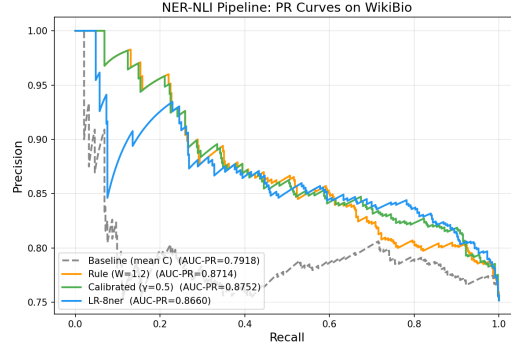


Figure 1: Precision-Recall curve comparing the NER-NLI hybrid method with the SelfCheckGPT-NLI baseline.

sentences into entity-centered claims and verifying them against a reference provides a more reliable hallucination signal than self-consistency alone.

7 Conclusion

In this project, we investigated whether hallucination detection in LLM-generated biographies can be improved by replacing the heuristic aggregation used in SelfCheckGPT-NLI with supervised models trained on features derived from NLI outputs. Using the WikiBio hallucination dataset, we evaluated several approaches that progressively expanded the feature representation derived from contradiction, entailment, and uncertainty signals. Our results show that supervised aggregation of contradiction-based statistics can provide small improvements over the baseline in some settings, particularly with appropriate regularization and evaluation splits. However, more complex feature extensions did not consistently improve performance. The bidirectional NLI and semantic disagreement method performed worse than the baseline, and the confidence-weighted NLI aggregation approach also failed to outperform the strong mean-contradiction heuristic. In contrast, the NER-NLI hybrid achieved a better performance (+17.4%) than the baseline and other self-reference methods, suggesting that atomic claim reference-based verification approaches can outperform self-reference methods.

Overall, our findings highlight two key insights. First, the simple mean contradiction aggregation technique used by SelfCheckGPT-NLI is a strong baseline that is difficult to exceed using incremental self-reference feature extensions and logistic regression alone. Second, extending NLI with structured claim decomposition with NER and external references can lead to meaningful improvements through richer factual signals. A key limitation of this work is the relatively small dataset and reliance on a fixed pretrained NLI model and simple classifiers. Future work could explore stronger NLI backbones, larger hallucination datasets, and enhanced retrieval-based grounding mechanisms that allow models to verify claims against external knowledge sources.

Considered together, our results suggest that while self-consistency remains a relatively efficient approach to hallucination detection, methods that utilize structured claim extraction like NER with reference-grounded verification represent a promising way forward to improve the detection and consistency of factual reliability in LLMs.

Team contributions

Sally worked on Bidirectional NLI and Semantic Disagreement. Sarah worked on Confidence-Weighted NLI Aggregation. Nathania worked on NER-NLI Hybrid approach. All worked on Supervised Aggregation of NLI Features and write up.

References

- [1] Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023.
- [2] Or Honovich, Leshem Choshen, and Omer Levy. True: Re-evaluating factual consistency evaluation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2022.
- [3] Shalini Raghuvanshi, Yashs Tiwari, and Abhishek Yadav. Integrating token-level uncertainty, bidirectional nli, and semantic entropy for robust hallucination detection in large language models. In *Proceedings of the 2025 International Conference on Artificial Intelligence and Emerging Technologies (ICAJET)*, 2025.
- [4] Prasenjit Dey, Srujana Merugu, and Sivaramakrishnan Kaveri. Uncertainty-aware fusion: An ensemble framework for mitigating hallucinations in large language models. In *Companion Proceedings of the ACM Web Conference (WWW '25)*, pages 947–951, 2025.
- [5] Neeraj Varshney, Wenlin Yao, Hongming Zhang, Jianshu Chen, and Dong Yu. A stitch in time saves nine: Detecting and mitigating hallucinations of llms by validating low-confidence generation. *arXiv preprint arXiv:2307.03987*, 2023.
- [6] Alexander Fabbri, Chien-Sheng Wu, and Caiming Liu, Wenhao Xiong. Qafacteval: Improved qa-based factual consistency evaluation for summarization. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2587 – 2601. Association for Computational Linguistics, July 2022.
- [7] Song Wang, Xun Wang, Jie Mei, Yujia Xie, Si-Qing Chen, and Wayne Xiong. Developing a reliable, fast, general-purpose hallucination detection and mitigation service. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 3: Industry Track)*, pages 971–978. Association for Computational Linguistics, April 2025.