BiScope: AI-generated Text Detection by Checking Memorization of Preceding Tokens

Hanxi Guo, Siyuan Cheng, Xiaolong Jin, Zhuo Zhang, Kaiyuan Zhang,
Guanhong Tao, Guangyu Shen, Xiangyu Zhang

PURDUE UNIVERSITY

THE UNIVERSITY OF UTAH

NEURAL INFORMATION PROCESSING SYSTEMS

Introduction

Why We Need to Detect AI-generated Text?

- **Emerging Applications with Large Language Models (LLMs)**

  LLMs are now widely used for tasks like document summarization and article enhancement, showing increasing importance in daily life.

- **Fast-growing Capabilities of LLMs**

  Models such as Claude 3.5 and GPT-4o offer unprecedented power and accessibility, making it harder to distinguish between AI-generated and human-generated content.

- **Increasing Misuse of LLMs**

  The widespread accessibility of large language models (LLMs) has led to a rise in misuse, compromising content authenticity and integrity, e.g., AI-generated phishing email and AI-assisted plagiarism

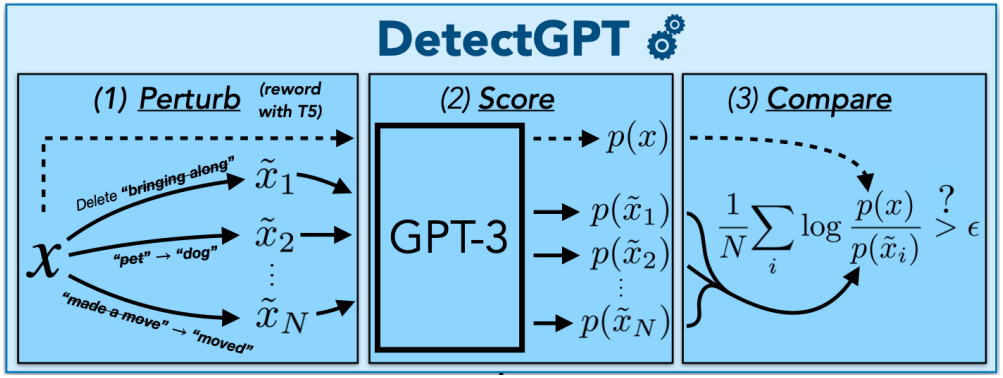
- **Inadequate Detection Methods**

  Existing detection tools lack the effectiveness and affordability to keep up with the development of advanced LLMs, creating a significant gap in detecting AI-generated artifacts.

Background

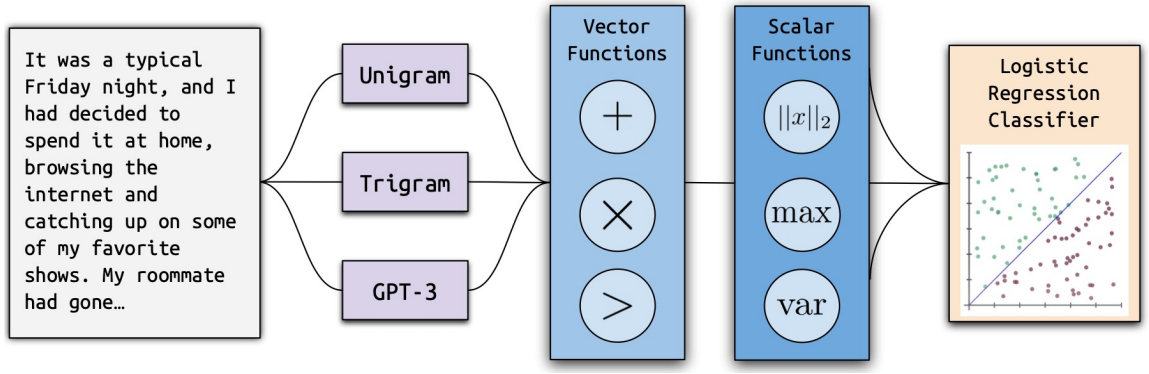
Taxonomy of Existing AI-generated Text Detection Techniques

Statistical Method



- Utilize one or several pre-trained surrogate models to simulate the generation process of LLMs
- Utilize various metrics on the surrogate model's output to calculate and assign a score to the given text

Training-based Method



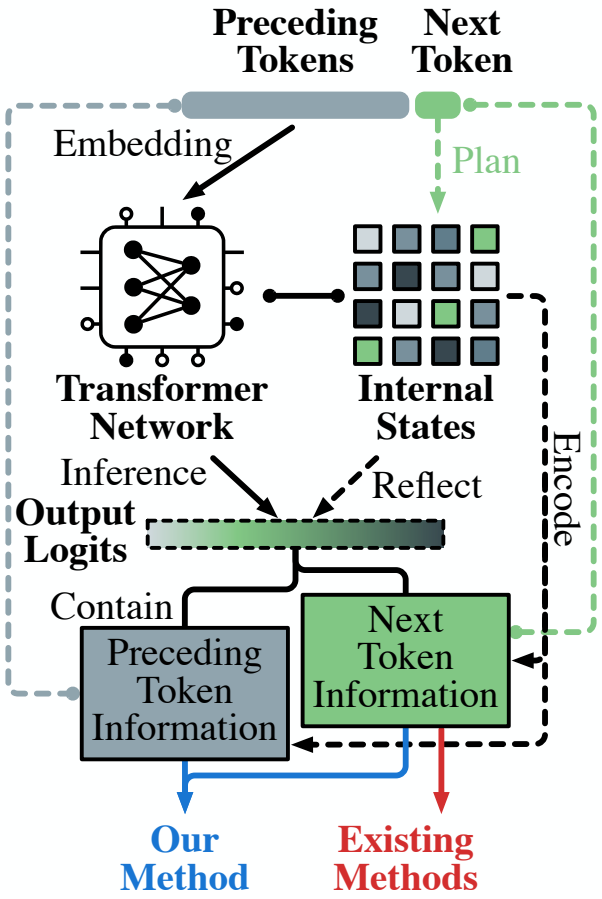
- Fine-tune a detection LLM to analyze the input text and predict its label.
- Exploit more complex features from surrogate model's output and train a classifier to do the prediction

PURDUE UNIVERSITY

- Left figure from Mitchell, Eric, et al. "Detectgpt: Zero-shot Machine-Generated Text Detection using Probability Curvature." ICML. 2023.
- Right figure from Verma, Vivek, et al. "Ghostbuster: Detecting Text Ghostwritten by Large Language Models." NAACL. 2024.

Motivation

Intuition Behind Our BiScope

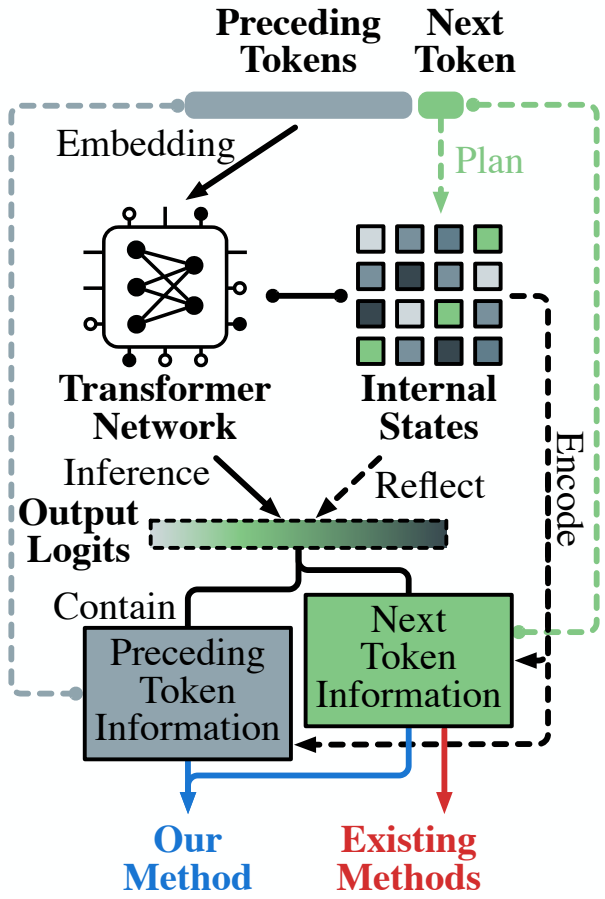


"For human-written text, the surrogate LLM has a poor prediction for the next token and a strong memory of the previous token, reflected in the output logits, whereas the behaviors for LLM-generated text are the opposite."

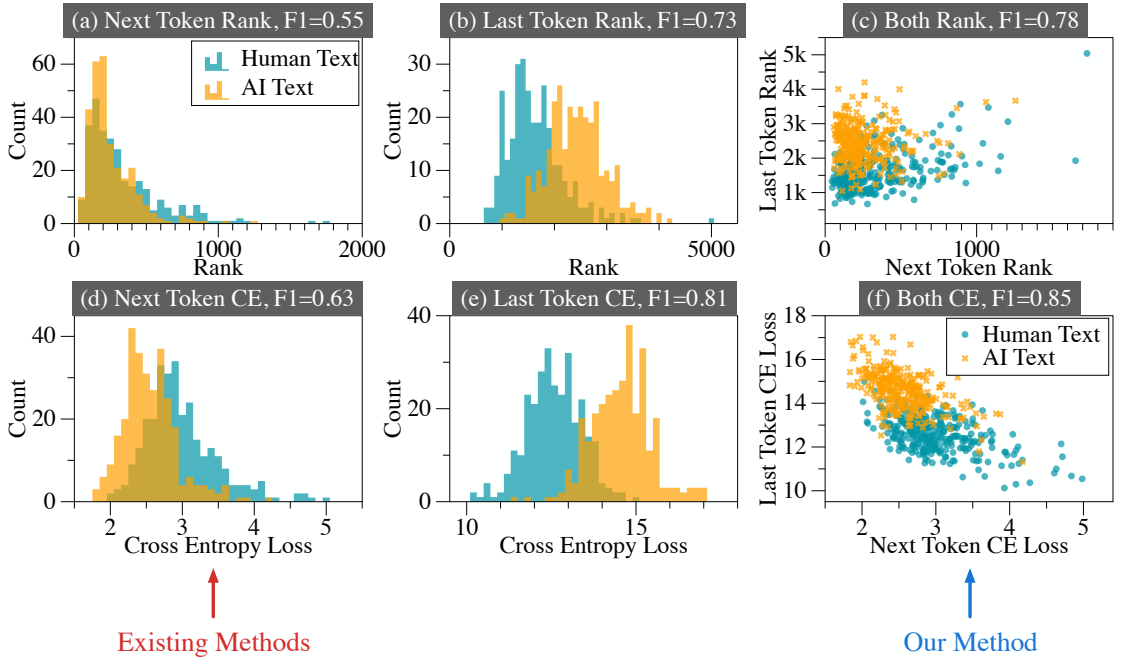
PURDUE UNIVERSITY

Motivation

Intuition Behind Our BiScope



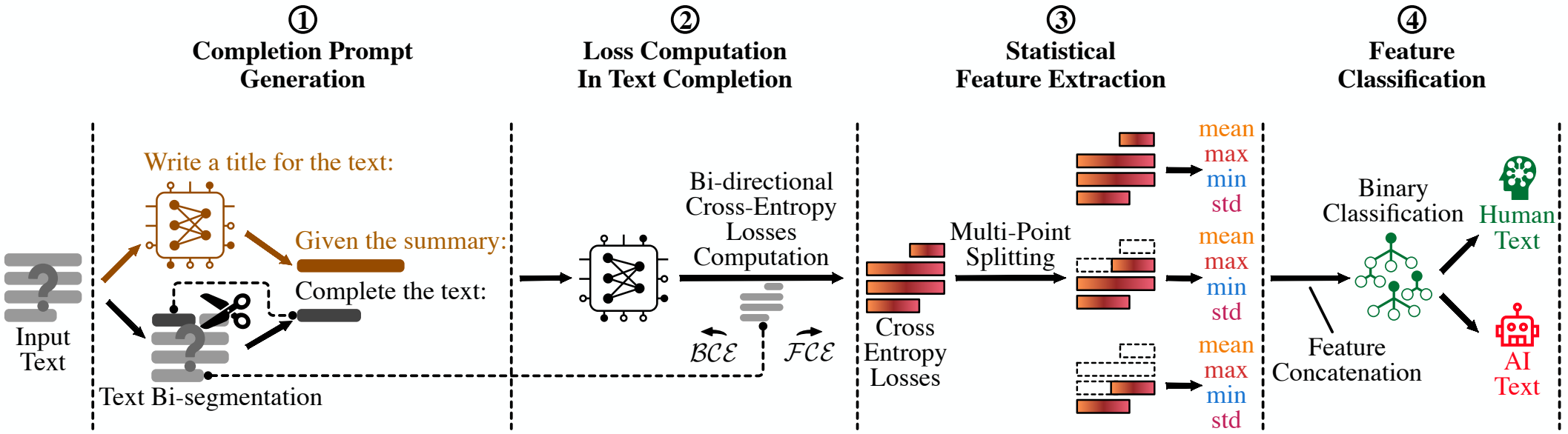
"For human-written text, the surrogate LLM has a poor prediction for the next token and a strong memory of the previous token, reflected in the output logits, whereas the behaviors for LLM-generated text are the opposite."



PURDUE UNIVERSITY

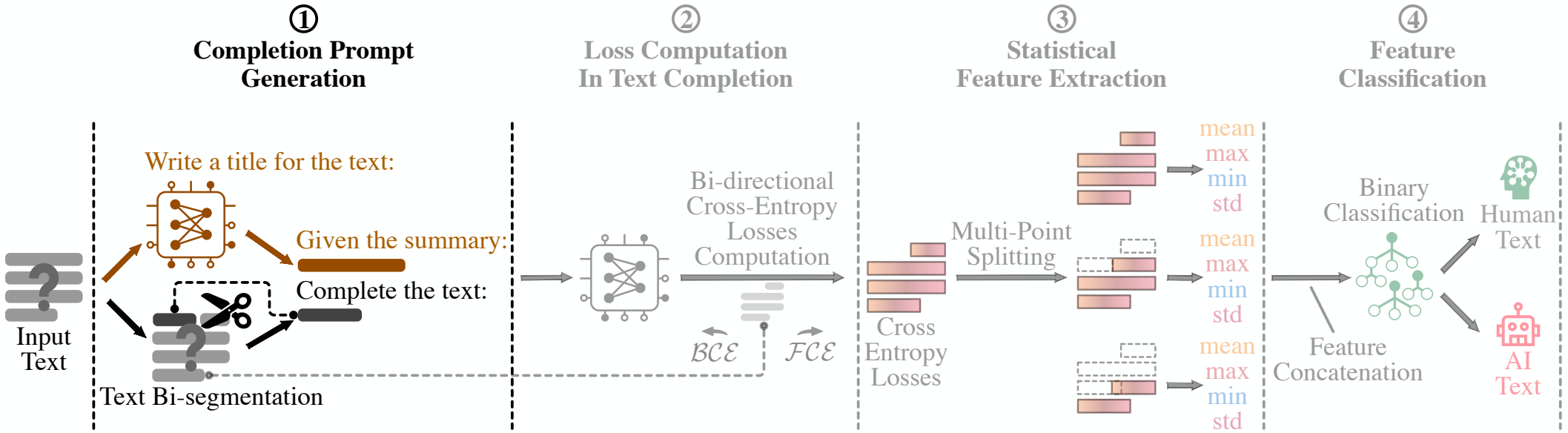
Design

Overview of BiScope



Design

Step 1: Completion Prompt Generation

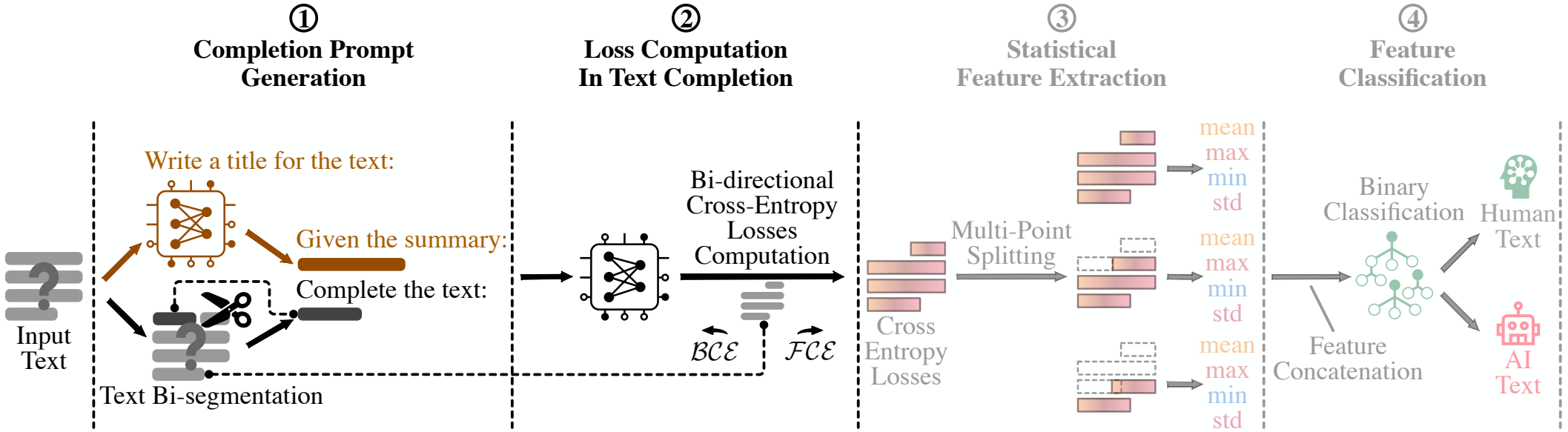


- We formulate the detection within a guided text completion scenario
- We first utilize a surrogate LLM to summarize the entire input text and obtain a summary as guidance
- Then we divide the input text as two segments and let the surrogate LLM complete the segment 2 based on the summary and segment 1

PURDUE UNIVERSITY

Design

Step 2: Loss Computation In Text Completion



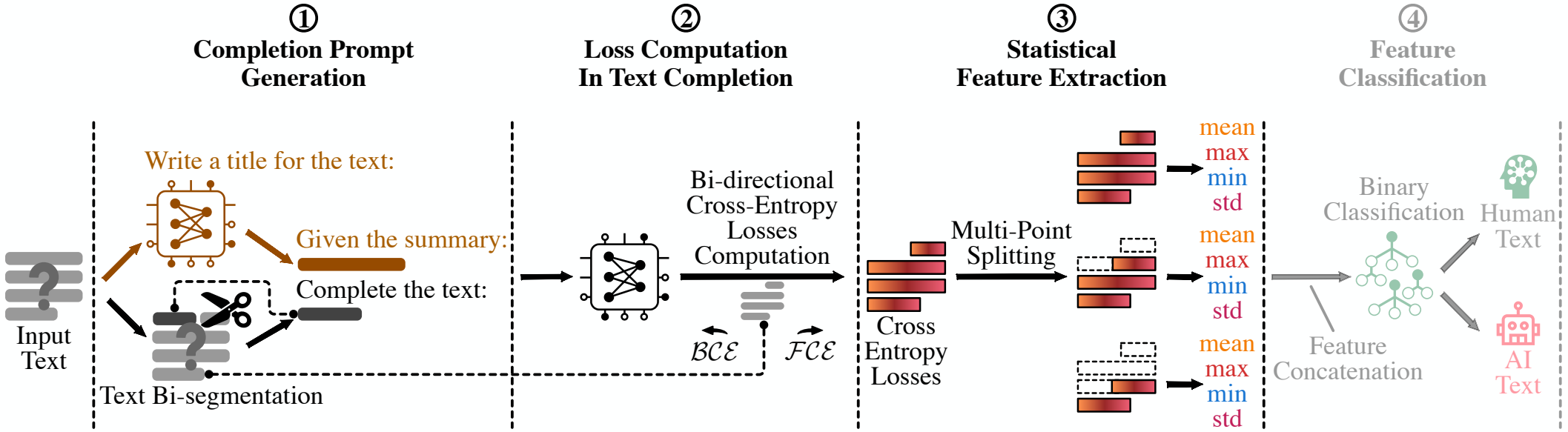
- We propose a novel bi-directional cross entropy loss calculation method to capture both the prediction and memorization information in the output logits, consisting of forward cross-entropy (*FCE*) loss and backward cross-entropy (*BCE*) loss

$$\mathcal{FCE}_i = -\sum_{z=1}^{||\mathcal{V}||} \tilde{\mathcal{P}}_{i+1}^z \cdot \log(\mathcal{P}_i^z) \qquad \mathcal{BCE}_i = -\sum_{z=1}^{||\mathcal{V}||} \tilde{\mathcal{P}}_i^z \cdot \log(\mathcal{P}_i^z)$$

PURDUE UNIVERSITY

Design

Step 3: Statistical Feature Extraction

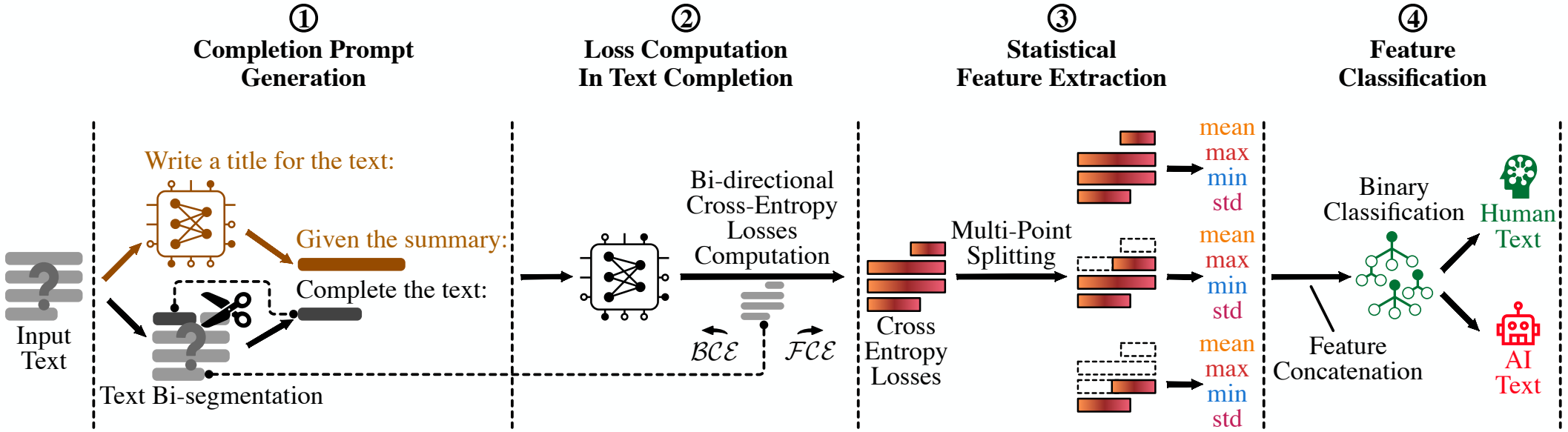


- We partition the entire loss sequence into *n* segments
- For each segment *i*, we compute the statistical values of the sub-sequence *[i, i+1, …, n]*
- The entire step only uses one-time LLM inference to simulate the features when different lengths of input text are given to the surrogate model

PURDUE UNIVERSITY

Design

Step 4: Feature Classification



- In the last step, we concatenate all the statistical features of both the *FCE* and *BCE* vectors into a one-dimensional feature vector, which is then used to train a binary classifier to perform the classification.
- This trained classifier can be directly deployed to detect unseen data, whether from unknown LLMs or unfamiliar text domains.

Evaluation

Dataset Settings

		Normal Dataset					Paraphrased Dataset				
	Data Type	Dataset Size	Average Len.	Min Len.	Max Len.	Median Len.	Dataset Size	Average Len.	Min Len.	Max Len.	Median Len.
Arxiv	Human	350	786.7	132	1736	715.0	-	-	-	-	-
	Machine	1750	787.1	101	1701	810.5	1400	875.8	174	1874	931.5
	All	2100	787.0	101	1736	799.5	1400	875.8	174	1874	931.5
Code	Human	164	631.5	132	1993	572.0	-	-	-	-	-
	Machine	819	413.3	41	1908	352.0	656	493.6	13	2333	382.5
	All	983	449.7	41	1993	387.0	656	493.6	13	2333	382.5
Yelp	Human	2000	554.9	37	4959	407.0	-	-	-	-	-
	Machine	9740	461.1	10	2548	414.0	8000	586.5	54	2593	537.0
	All	11740	477.1	10	4959	413.0	8000	586.5	54	2593	537.0
Essay	Human	1000	4249.9	1276	41470	3301.5	-	-	-	-	-
	Machine	4897	3827.7	515	21094	3486.0	3999	3666.8	129	19878	3284.0
	All	5897	3899.3	515	41470	3449.0	3999	3666.8	129	19878	3284.0
Creative	Human	1000	2899.0	499	9933	2462.5	-	-	-	-	-
	Machine	4840	2851.9	176	13716	2620.0	4000	2924.4	85	16812	2674.5
	All	5840	2860.0	176	13716	2588.5	4000	2924.4	85	16812	2674.5

- We extend existing datasets and craft a large-scale public dataset for more challenging AI-generated texts, consisting of **25** distinct groups and more than **22,000** samples generated from **5** latest commercial LLMs from OpenAI, Anthropic, and Google.
- Our extended datasets contains two long natural language datasets (Essay, Creative), two short natural language datasets (Arxiv, Yelp), and a code dataset (Code)

PURDUE UNIVERSITY

Evaluation

Detection Performance Compared to Existing Open-source Techniques

		Normal Dataset					Paraphrased Dataset			
	Method	GPT-3.5 Turbo	GPT-4 Turbo	Claude-3 Sonnet	Claude-3 Opus	Gemini 1.0-pro	GPT-3.5 Turbo	GPT-4 Turbo	Claude-3 Sonnet	Claude-3 Opus
Arxiv	Zero-shot Query	0.5768	0.5835	0.6764	0.6667	0.6666	0.5587	0.6116	0.6916	0.6935
	Log Rank	0.6572	0.7006	0.8015	0.8809	0.8560	0.6628	0.6660	0.6634	0.6747
	LRR	0.6602	0.7031	0.8116	0.8596	0.8544	0.6654	0.6654	0.6654	0.6654
	DetectGPT	0.6654	0.6634	0.6673	0.6673	0.6673	0.6641	0.6628	0.6654	0.6654
	RADAR	0.9566	0.7858	0.7034	0.7754	0.7868	0.9203	0.6970	0.6884	0.7202
	Raidar	0.8316	0.8157	0.8029	0.8289	0.7366	0.9004	0.8851	0.8052	0.8303
	OpenAI Detector	0.7889	0.6660	0.6673	0.6673	0.6976	0.7062	0.6654	0.6673	0.6673
	Binoculars	0.9097	0.9135	0.9256	0.9699	0.9560	0.6617	0.6971	0.8112	0.8672
	GhostBuster	0.9716	0.9886	0.9815	0.9813	0.9571	0.9700	**0.9943**	**0.9814**	0.9856
	BiScope	0.9870	0.9928	0.9796	0.9885	0.9708	0.9769	0.9800	0.9625	0.9870
	BiScope*	**0.9928**	**0.9943**	**0.9869**	**0.9913**	**0.9797**	**0.9870**	0.9859	0.9593	**0.9884**

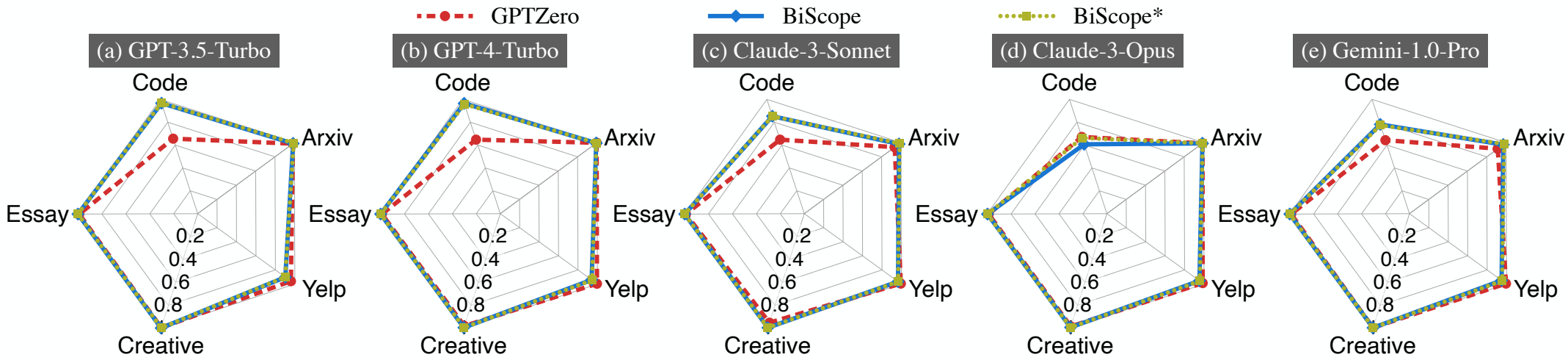
		Normal Dataset					Paraphrased Dataset			
	Method	GPT-3.5 Turbo	GPT-4 Turbo	Claude-3 Sonnet	Claude-3 Opus	Gemini 1.0-pro	GPT-3.5 Turbo	GPT-4 Turbo	Claude-3 Sonnet	Claude-3 Opus
Code	Zero-shot Query	0.6300	0.5833	0.4351	0.3524	0.1854	0.6690	0.6784	0.6400	0.4545
	Log Rank	0.6581	0.6610	0.6611	0.6569	0.6583	0.6612	0.6611	0.6556	0.6581
	LRR	0.6639	0.6639	0.6639	0.6639	0.6542	0.6639	0.6639	0.6639	0.6639
	DetectGPT	0.6361	0.6474	0.6583	0.6612	0.6682	0.6612	0.6639	0.6639	0.6612
	RADAR	0.6680	0.6653	0.6652	0.6597	0.6626	0.6598	0.6653	0.7322	0.6653
	Raidar	0.9368	0.8220	0.6121	0.6156	0.4858	0.9325	0.8744	0.8250	0.6197
	OpenAI Detector	0.7213	0.6977	0.6916	0.6542	0.6666	0.7514	0.6639	0.6639	0.6695
	Binoculars	0.7073	0.6512	0.6612	0.6653	0.6624	0.7101	0.6338	0.8041	0.7179
	GhostBuster	0.8524	0.7942	0.6556	**0.6749**	0.3860	0.8662	0.7729	0.7757	0.5390
	BiScope	0.9665	**0.9655**	**0.8528**	0.6069	**0.7809**	**0.9659**	**0.9464**	**0.9691**	**0.9250**
	BiScope*	**0.9692**	0.9586	0.8526	0.6620	0.7741	0.9597	0.9435	0.9600	0.9222

- On natural language datasets, BiScope outperforms **nine** state-of-the-art baseline detection methods with **0.26** additional detection F1 score on average
- On code dataset, BiScope outperforms all the **nine** baselines with **0.21** detection F1 score increase on average

PURDUE UNIVERSITY

Evaluation

Detection Performance Compared to Commercial Detection Tool



- We also compared our BiScope with the latest version of the renowned commercial AI-generated text detection tool – GPTZero[1].
- Our BiScope outperforms GPTZero in **72%** of cases, particularly in the Code dataset, where it achieves a **0.19** average F1 score improvement.

[1] Edward Tian and Alexander Cui. "Gptzero: Towards detection of ai-generated text using zero-shot and supervised methods", 2023.

Conclusion

Our Achievements with BiScope

- We propose a novel AI-generated text detection algorithm that exploits both the preceding token information (i.e., memorization) and the next token information (i.e., prediction) via an innovative bi-directional cross-entropy loss calculation method.

- We are the first to utilize text summaries to guide the detection, further enhancing its effectiveness and robustness toward heterogeneous data.

- We extend existing datasets and craft a large-scale public dataset for more challenging AI-generated texts, consisting of 25 distinct groups and more than 22, 000 samples sourced from five latest commercial LLMs.

- We develop a prototype named BiScope, a detection pipeline without any fine-tuning needed for the detection LLM. We evaluate it on our dataset and show that BiScope outperforms nine state-of-the-art baseline techniques.

Thank You!

PURDUE UNIVERSITY®