DatedGPT: Preventing Lookahead Bias in Large Language Models with Time-Aware Pretraining

Yutong Yan[α] Raphael Tang[β]

[α]The Chinese University of Hong Kong [β]University College London

Motivation

- **Definition:** Lookahead bias occurs when models are trained on future data and evaluated on past events
  - Model accesses information unavailable at prediction time
  - Creates artificially inflated performance metrics
  - Violates temporal causality in predictive modeling
- **Example:** LLM predicting 2020 corporate risks using 2019 earnings calls [1]
  - Generated "COVID-19" in 6.8% of outputs despite term not existing in 2019
  - Showed indirect leakage: "pandemic" and "supply chain" mentioned significantly more for 2020 vs 2019 predictions
  - Appears prescient but relies on impossible future knowledge
  - Real 2019 deployment would lack awareness of impending pandemic

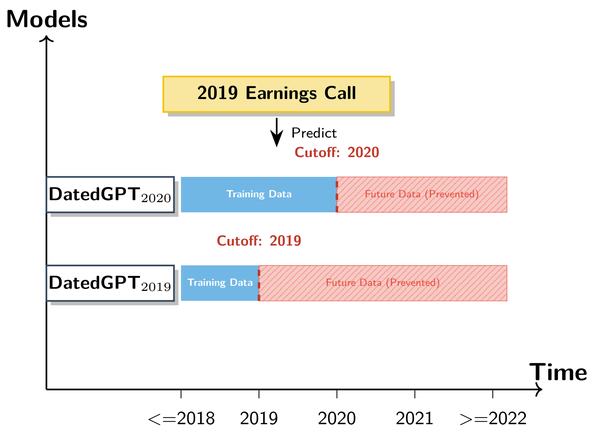
Research Question

How can we train large language models that reflect knowledge available at specific time points?

- Prevent future information use for historical predictions

DatedGPT Solution

- **Time-Aware Framework:** Trained strictly on pre-cutoff data to ensure temporal integrity.
- **Unprecedented Scale:** Largest model family in financial research (GPT-3-XL scale, 1.3B parameters).
- **Core Innovation:** Eliminates future data leakage → Reliable models.



Methodology

Time-Aware Training Pipeline

1. Temporal Dataset Construction
   - FineWeb dataset from annual CommonCrawl snapshots (2013–2024)
   - Most recent crawl per calendar year, filtered and deduplicated
   - Generate multiple cutoff-specific datasets with strict temporal boundaries
2. Sequential Annual Model Training
   - Train separate models (1.3B parameters) from scratch for each year
   - 100B tokens per model, preventing future information leakage
   - 12 model variants spanning complete temporal range

⇒ Model Family:
$\{\text{DatedGPT}_{2013}, \text{DatedGPT}_{2014}, \ldots, \text{DatedGPT}_{2024}\}$

DatedGPT Pretraining

Temporal Training Process:

$$\mathcal{D}_t = \{(x_i, y_i) : \text{timestamp}(x_i, y_i) \leq t\} \quad (1)$$

$$\theta_t^* = \arg\min_{\theta} \mathcal{L}(\theta; \mathcal{D}_t) \quad (2)$$

$$\text{DatedGPT}_t = f_{\theta_t^*} \quad (3)$$

Key Design Principles:

- Strict temporal data filtering with no lookahead
- Progressive training across chronological periods
- Consistent architecture (comparable to GPT-3 XL) across all variants

Case Study

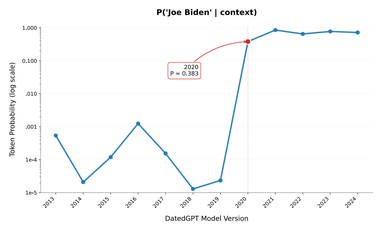


Figure 1. Log-scale probability of generating "Joe Biden" in response to the prompt "The winner of the 2020 U.S. presidential election is President-elect", across DatedGPT models.

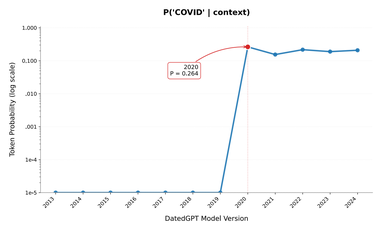


Figure 2. Log-scale probability of generating the token "Covid" in response to the prompt "One of the top concerns for the U.S. economy in year 2020 is the impact of", across DatedGPT models.

Chat Demo: DatedGPT.com

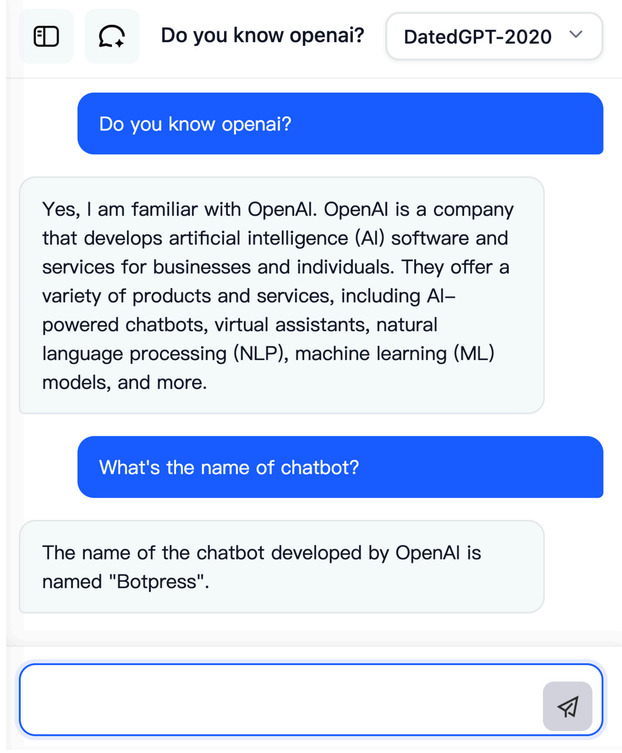


Figure 3. Example output from the DatedGPT-2020 model.

LLM Benchmark Results

Model	HellaSwag	ARC-Challenge	ARC-Easy	Avg.
TinyLLaMA-1B	43.5	26.5	53.0	41.0
DatedGPT-2013	48.8	36.0	66.9	50.6
DatedGPT-2024	54.4	39.4	70.6	54.8

Contact

Contact me and try demo.
`https://yutongyan.xyz/`
`yutong.yan@link.cuhk.edu.hk`