Finance and Economics Discussion Series

Federal Reserve Board, Washington, D.C.
ISSN 1936-2854 (Print)
ISSN 2767-3898 (Online)

Total Recall? Evaluating the Macroeconomic Knowledge of Large Language Models

Leland D. Crane, Akhil Karra, Paul E. Soto

2025-044

Please cite this paper as:
Crane, D. Leland, Akhil Karra, Paul E. Soto (2025). "Total Recall? Evaluating the Macroeconomic Knowledge of Large Language Models," Finance and Economics Discussion Series 2025-044. Washington: Board of Governors of the Federal Reserve System, https://doi.org/10.17016/FEDS.2025.044.

NOTE: Staff working papers in the Finance and Economics Discussion Series (FEDS) are preliminary materials circulated to stimulate discussion and critical comment. The analysis and conclusions set forth are those of the authors and do not indicate concurrence by other members of the research staff or the Board of Governors. References in publications to the Finance and Economics Discussion Series (other than acknowledgement) should be cleared with the author(s) to protect the tentative character of these papers.

Total Recall? Evaluating the Macroeconomic Knowledge of Large Language Models[*]

Leland D. Crane[†] Akhil Karra[‡] Paul E. Soto[†]

June 24, 2025

Abstract

We evaluate the ability of large language models (LLMs) to estimate historical macroeconomic variables and data release dates. We find that LLMs have precise knowledge of some recent statistics, but performance degrades as we go farther back in history. We highlight two particularly important kinds of recall errors: mixing together first print data with subsequent revisions (i.e., smoothing across vintages) and mixing data for past and future reference periods (i.e., smoothing within vintages). We also find that LLMs can often recall individual data release dates accurately, but aggregating across series shows that on any given day the LLM is likely to believe it has data in hand which has not been released. Our results indicate that while LLMs have impressively accurate recall, their errors point to some limitations when used for historical analysis or to mimic real time forecasters.

[*] We thank Gary Cornwall, Anne Hansen, participants in Board brownbags, and participants at the 2025 SGE conference for useful comments. We thank Betsy Vrankovich for her technical expertise.
Opinions expressed herein are those of the authors alone and do not necessarily reflect the views of the Federal Reserve System or the Board of Governors.

[†] Board of Governors of the Federal Reserve System

[‡] Carnegie Mellon University

1 Introduction

The rise of large language models (LLMs) has generated interest in how they can be used for economic analysis and forecasting (e.g., Korinek 2023). The utility of LLMs depends on their understanding of economics-related facts and their ability to follow instructions precisely. We evaluate LLMs on several dimensions related to these capabilities. First, how well do LLMs estimate important macroeconomic variables from the past? Second, to what extent are LLMs' estimates contaminated with future information? And third, how well do LLMs recall data release *dates*? LLMs which have accurate knowledge of economic history (including data release dates) will likely be more useful when generating hypotheses and doing analysis. Separately, if LLMs can provide realistic quasi-real-time estimates—simulating forecasters from the past—then we can better understand how the LLM's forecasting process relates to human forecasts. On the other hand, LLM estimates which are inaccurate or contaminated with look-ahead bias may be of more limited use.

We find that for some variables LLMs have remarkable recall.[1] The LLM we focus on—Claude Sonnet 3.5—can recall the quarterly values of the unemployment rate and CPI with fairly high accuracy back to WWII. However, it fares much more poorly on more volatile real activity series like real GDP growth and industrial production (IP) growth. The LLM appears to miss many of the high-frequency swings in these series, though it does capture business cycle variation well.

Focusing on GDP, we develop evidence that the LLM estimate is a mixture of the first print value for the reference period and subsequent revised values for that reference period. This smoothing across data vintages appears regardless of whether we ask the LLM to provide the first print or the fully revised number. LLMs are trained on an enormous amount of data and—unless every part of the corpus is clearly date stamped and that information

[1] We use the term *recall* when the LLM is estimating a historical quantity which was (presumably) in its training data. This is distinct from "retrieval" in the context of retrieval augmented generation, where the LLM is backed by a search engine and reference documents. Our focus is on the LLM in isolation, and which historical facts it is able to estimate accurately.

is embedded in the model weights by the training process—it won't always be clear when the text was written or which vintage of GDP it is referring to. The mixing of first print and fully revised data is problematic, because it means (1) the model has a less than accurate retrospective understanding of the economic situation, and (2) the model will have difficulty simulating a real-time forecaster.

A related but distinct question is whether LLM estimates for a given reference period are influenced by future and past reference periods, keeping the vintage constant. In other words, are LLM estimates of data published for date t affected by published data values from $t+1$? We develop a test for whether the LLM's estimate for a particular date is influenced by future shocks to the series, controlling for expectations. We find suggestive evidence that LLM's do indeed use future reference period value when constructing an estimate, even when instructed to ignore future information. Any such smoothing is again a challenge for historical analysis and using LLMs to mimic real-time forecasters.

Finally, we document the LLM's knowledge of economic data release *dates*. We find that LLMs often have an accurate idea of when historical data releases occurred. However, they sometimes miss the true release date by a few days. The results are also sensitive to the details of the prompt; we find that varying the prompt to reduce the number of estimate release dates that are *late* leads to an increase in estimated release dates that are too *early*. Our prompt engineering doesn't lead to a strategy that increases accuracy to a very high level; rather we end up trading off different types of errors. The conclusion is that the LLM doesn't have a very strong conception of the individual data release dates. We find that—aggregating across major economic indicators—on a typical day there is a good chance the LLM falsely believes at least some major data releases have occurred. Interestingly, these errors are exactly the kind we would expect a human to make: sometimes too early, sometimes too late, and attempts to reduce one kind of error increase the other.

Our results paint a mixed picture of current LLM capabilities. LLM recall of historical data values and release dates is often very impressive. That said, there are also significant shortcomings in LLM recall, and the errors are often correlated with information from after the reference date. At a high level these errors are very human in that they can be interpreted as a good-faith effort to follow instructions while being hampered by a fuzzy recollection of the past. These patterns suggest that look-ahead bias may be an important challenge when using LLMs.

2 Literature Review

A number of recent papers have used LLMs for economic forecasting and analysis. Kim et al. (2024) find that an LLM can predict firm earnings when prompted with anonymized accounting data. Cook et al. (2023) use LLMs to analyze earnings calls. Pham and Cunningham (2024) present out-of-sample (i.e. post-knowledge cutoff) forecasts for inflation and Academy Awards. Schoenegger et al. (2024) show that GPT4 can help human forecasters on a variety of financial and political forecasting tasks, all of which occurred after the knowledge cutoff. Similarly, Phan et al. (2024) compare LLM forecasts with crowd-sourced forecasts. Jha et al. (2024) feed earnings call transcripts to GPT3.5 and show that it can help forecast capital investment and abnormal returns. As part of their robustness exercises they restrict the sample to the post-knowledge cutoff period, and separately try to anonymize the transcripts. Glasserman and Lin (2023) examine GPT3.5's ability to forecast stock returns from news headlines; they anonymize company names to avoid an in-sample "distraction" effect. Faria-e-Castro and Leibovici (2023) evaluate inflation forecasts from an LLM, both before and after the knowledge cutoff. Zarifhonarvar (2024) studies how different prompts and access to different information affect GPT4's inflation expectations. Separately, a strand of the literature has used LLMs as stand-ins for humans in surveys or strategic games (Manning et al. (2024), Kazinnik (2024), Tranchero et al. (2024).) Hansen et al. (2024) contribute to both literatures, simulating Survey of Professional Forecasters (SPF) respondents and evaluating the properties of the LLM-derived forecasts. Finally, a number of papers use LLMs as classifiers for things like news headlines, and then use the classifications to build indicators like sentiment indexes (Shapiro et al., 2022; Bybee, 2023; Cajner et al., 2024; van Binsbergen et al., 2024).

Many of these papers acknowledge look-ahead bias—the potential for an LLM that is supposed to mimic an agent acting at time t to use information from $t + 1$ or later—and attempt to address it with anonymization, post-knowledge-cutoff comparisons, and prompting techniques. Somewhat less has been done to directly measure the extent of look-ahead bias.[2] Sakar and Vafa (2024) is one exception, they show look-ahead bias arises in two contexts where GPT4 is asked to act as a real time forecaster: first, when assessing pre-pandemic earnings calls for risk factors, the LLM sometimes mentions pandemics and Covid. Second, the LLM is often able to "forecast" the winner of close elections. Lopez-Lira et al. (2025) evaluate recall and look-ahead bias for financial macroeconomic variables; interestingly, their estimates of recall of recall accuracy are higher than ours, suggesting some model- or prompt-specific effects. We complement these papers by developing more formal tests of data leakage in the macroeconomic setting and exploring the LLM's understanding for data release dates, a critical factor for real-time forecasting. Ludwig et al. (2025) also discuss look-ahead bias in the context of congressional legislation and financial news. To address these concerns Sarkar (2024) and He et al. (2025) develop sequences of LLMs trained only on data up to a known point in time, but of course these models are much smaller than the commercially available ones and do have the full set of capabilities available with frontier models.

Look-ahead bias is also a focus of our paper; we add to the literature by quantifying several practically important types of look-ahead bias, e.g. the contamination of an LLM's memories of first-print data with later revisions and uncertainty about the timing of data releases. We also develop a test for whether LLM's estimates are contaminated by future data values.

Assessing look-ahead bias is hard. LLMs have attracted attention from forecasters pre-

[2] See Croushore (2011) for a detailed discussion of the related topics of data revisions and forecast instability in traditional forecasting.

cisely because there is reason to think they might prove useful for prediction. This means that high accuracy at forecasting cannot be counted as strong evidence of look-ahead bias; LLMs are capable forecasters we should expect them to beat some other forecasts. In this paper we take an indirect approach, focusing on the LLM's recall of historical data values/release dates. It appears easier to show that errors in *recall* are influenced by future information than it is to prove that a *forecast* is "too accurate". Note that Hansen et al. (2024) prompt the LLM with recent values of macroeconomic indicators to ground it and help improve performance; this strategy may also help mitigate look-ahead bias. Our work complements theirs by documenting the capabilities and limitations of the raw LLM without additional information passed into the prompt.

Our assessment goes beyond the topic of look-ahead bias, as we test whether the LLM can accurately recall economic statistics in general. An analyst using an LLM to explore economic hypotheses would want the model to have a clear, precise understanding of economic history. Documenting the extent of recall and the limitations on LLM's knowledge will assist researchers considering how to use these tools.

3 Models and Data

For most of the paper we focus on four macroeconomic time series: GDP, inflation, industrial production, and unemployment. Similarly to Hansen et al. (2024), we restrict our attention to quarterly values so that we can compare to the SPF. The details of the series are as follows:

- Gross Domestic Product (GDP): The seasonally adjusted annualized one quarter growth rate of real GDP
- Inflation: The four quarter change in the seasonally adjusted Consumer Price Index (CPI)
- Industrial Production (IP): The seasonally adjusted annualized one quarter growth rate of IP

- Unemployment: The one quarter average of the seasonally adjusted level of the unemployment rate

We use both the fully-revised (current vintage) numbers, as well as the first-print values.

3.1 Models

We use Anthropic's Claude Sonnet 3.5 large language model as provisioned through AWS Bedrock.[3] Sonnet 3.5 is widely considered to be comparable to OpenAI's contemporaneous offerings (though it does not have the reasoning capabilities of o1 and later models), and it performs very well on benchmarks. Note that this model does not have internet search or tool use enabled; it cannot access any updated information aside from what is included in the prompt. We do not use OpenAI's models because we do not have an easy way to access them.

3.2 Methodology

Our main queries instruct the LLM to think step-by-step, write out their reasoning, and only write the final answer at the end. This is intended to improve performance, as LLMs can benefit from reasoning step-by-step before committing to an answer (Wei et al., 2022). The system prompt can be found in Figure 18, and an example user prompt is shown in Figure 19.

The responses to the queries are verbose. We use a secondary "summarizer" LLM and prompt to extract the estimate from the responses. The summarizer is instructed to read the original response and return an answer approximately of the form "`Answer:{estimate}`", where `{estimate}` is the desired estimate. We then parse the summarizer's answers with a regular expression (regex) to extract the numeric point estimate.

It is worth noting that the development of the prompts is an iterative process. Our initial

[3] The model ID is `anthropic.claude-3-5-sonnet-20240620-v1:0`. This is the original Sonnet 3.5, not the newer version of Sonnet 3.5 released in October 2024.

attempts yielded many ranges (not point estimates) and many failures to answer. To address this we added instructions to always produce an answer and to avoid giving ranges. As another example, our parser would sometimes fail to locate the answer. We found this was because the summarizer was not consistent about capitalizing "`Answer`", which we fixed by changing the regex.

3.3 Nondeterminism in Answers

In typical use LLM responses are stochastic. The LLM generates a response one token at a time and the token generated is a function of the text—either in the prompt or the incomplete response—up to that point in time.[4] The LLM generates tokens by sampling from the model's probability distribution of next tokens, so more probable completions are chosen more often.

Several parameters govern the sampling process. In older, smaller LLMs (like GPT-2) the most important is the *temperature*. In simple LLMs a temperature of zero corresponds to an essentially deterministic response. However, frontier models include other factors (like mixture of experts) that introduce other sources of randomness.

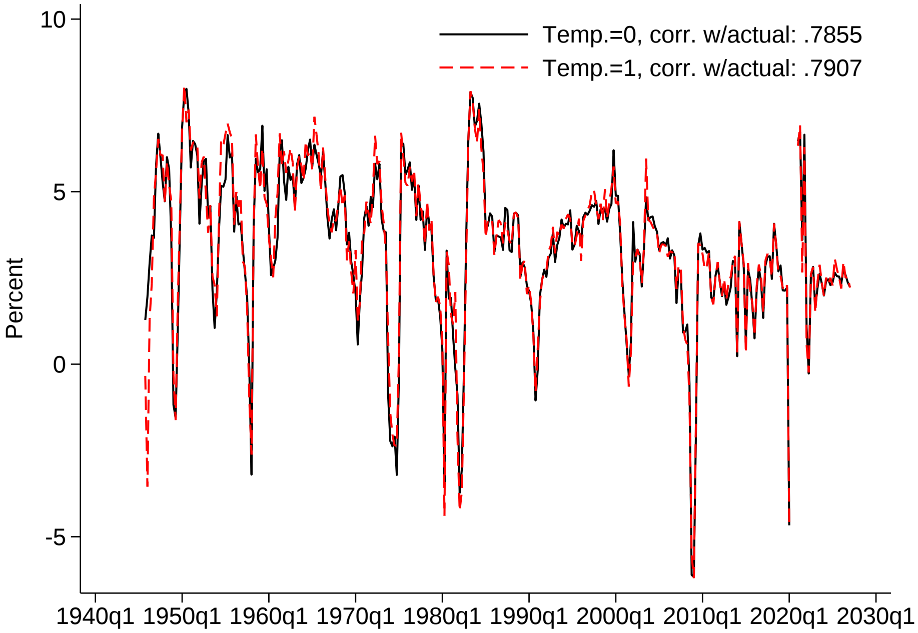
We run each query several times and average estimates in order to attenuate the randomness in LLM responses. We also calculate the standard error of this mean estimate and use it to plot confidence intervals. The averaged responses are close to deterministic, and the confidence intervals show us where there is still significant randomness.

3.4 Choosing the Temperature

We need to evaluate how much the temperature parameter matters in our context and what value to set it to. Figure 1 shows two GDP estimates: one with the temperature set to one (the default), and one with the temperature set to zero.[5] The two series are extremely similar.

[4] Tokens are words or word parts, for example `‘‘the’’` may be a single token but "generates" might be tokenized as `generat,es`

[5] For the temp.=0 version we also set the "top k" parameter equal to one; in a simple LLM this would ensure that the LLM chooses only the most probable next token conditional on the set of available tokens and their



Note: LLM estimates of GDP under different temperature parameters. Correlations are with actual, final print GDP. Covid period not plotted to keep scale readable.
Source: Authors' calculations, BEA

Figure 1: Temperature and Recall of GDP

Their correlations with actual first-print GDP are also similar, though the temp.=1 series has a marginally higher correlation. Based on this—and the fact that the temperature is set to one by default—we use temp.=1 as the main specification in most of what follows.

3.4.1 Digression: Nondeterminism at Temperature=0

Interestingly, the (within-quarter) standard deviations of the different temperature series are also very similar. In particular, for the temp.=1 series the average within-quarter standard deviation of the estimates is 0.786, while the average standard deviation for the temp.=0 series is 0.7616. While the temp.=0 series appears to have marginally less variability, the size of the effect is very small.

A lack of complete determinism with temp.=0 is understood to be a feature of the larger

probabilities. Like setting temp.=0, this would make the response deterministic in a simpler LLM.

LLMs.[6] But the near-identical results we see above raise questions as to whether the temperature parameter has any material impact at all, or whether our code base is setting it correctly. Table 1 shows that we can in fact document some effect of temperature. For this exercise we look at the raw, text response of the LLM, before parsing and summarization. We fix a character length N (say, 50 characters) and compare the first N characters of two random responses. The comparison is done within quarters, so the prompts for the two responses are identical. We check whether the first N characters of the response are identical, and record an indicator variable that equals 1 for a match and 0 for a difference. Thus each pair of responses generates a single indicator variable, and we repeat the process many times. Table 1 shows the results. When looking at the first 50 characters, with temperature set to zero 42 percent of response pairs are identical; this drops to 22 percent with temperature set to one. This amounts to a significant change in the variability of the responses, though there is obviously a great deal of variation in the zero temperature responses.

It appears that setting the temperature to zero for Sonnet 3.5 on Bedrock does indeed make the response string more deterministic as measured by increasing response similarity across identical queries. However, setting temperature to zero does not remove randomness by any means and makes very little difference for the substance of the response: the GDP estimate. Our results generally mirror those of Ouyang et al. (2025), who show significant non-determinism in OpenAI's GPT-3.5 and GPT-4 models even with temperature set to 0. We would caution users against assuming that temp.=0 ensures deterministic or even mostly deterministic results. Even with temp.=0 averaging across several queries still seems necessary to ensure that results are reproducible.

[6] The documentation for Claude mentions that "Note that even with temperature of 0.0, the results will not be fully deterministic." See also Ouyang et al. (2025).

Sequence Length	Temperature	Obs.	Mean	St. Dev.
50 chars.	0	3150	0.42	0.49
	1	3150	0.22	0.42
100 chars.	0	3150	0.37	0.48
	1	3150	0.14	0.34
200 chars.	0	3150	0.27	0.44
	1	3150	0.03	0.18

Table 1: Fraction of responses identical at various sequence lengths

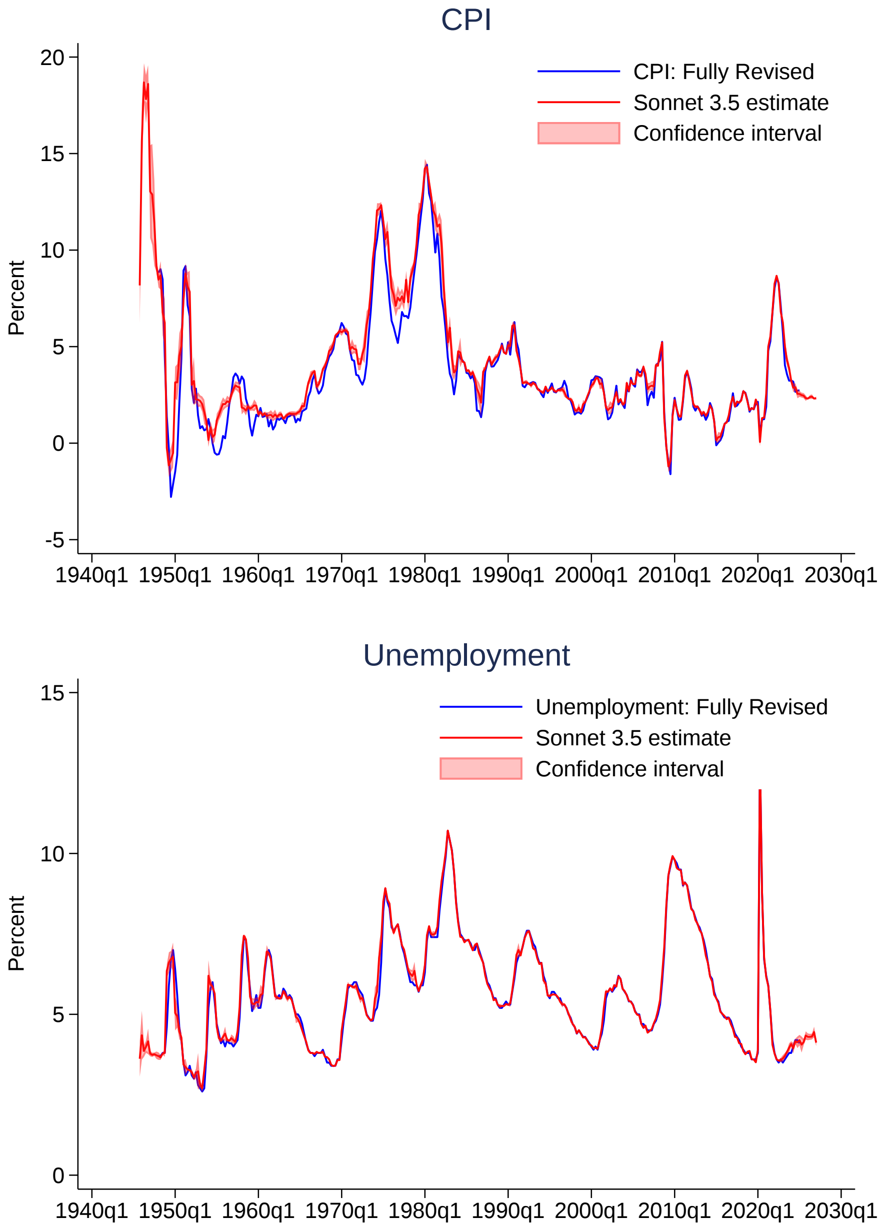
4 Testing LLM Recall

In this section we test how well LLMs recall important macroeconomic statistics. The prompt—shown in Figure 19—asks the LLM to use all information available to them (i.e., the LLM is not instructed to behave as a real time forecaster.) We ask the LLM for estimates through 2027 which it provides even though its knowledge cutoff is in 2024. Examining the LLM responses in these cases show it decides to provide a forecast in these cases. The most recent actual data available as of this writing is for 2025Q1.

Figure 2 shows the results for CPI inflation and the unemployment rate. In each panel the blue line is the true, fully-revised series. The red line is the average estimate returned by the LLM, and the pink band is the 95 percent confidence interval based on the variability of the 10 iterations of each query. It is evident that the LLM generally recalls something very close to truth for both series. The only major visible gaps appear for pre-1990 CPI inflation, where the LLM seems to be biased up when inflation is low. In addition, the confidence bands are tight, indicating little variability in the LLM responses.

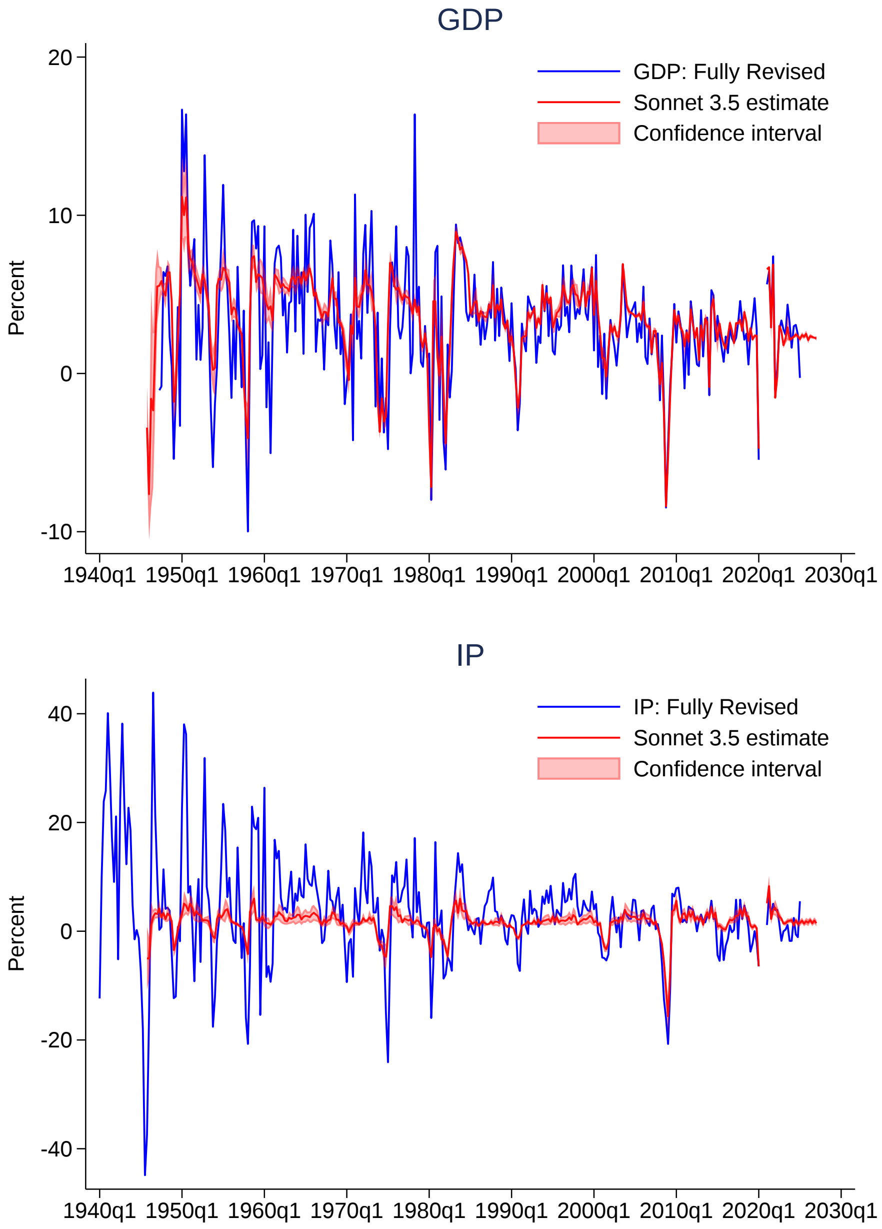
Figure 3 shows the same exercise for real GDP growth and industrial production growth. Here, the story is quite different. The LLM consistently misses the high-frequency swings in these series, though it does track many business cycle movements. Note that the year 2020 is not plotted since the pandemic real activity swings would dwarf the rest of the variation.

It is easier to see the dynamics in Figures 4 and 5, which focus on the 1990-2019 period. During this period, CPI inflation and the unemployment rate are recalled precisely. On the



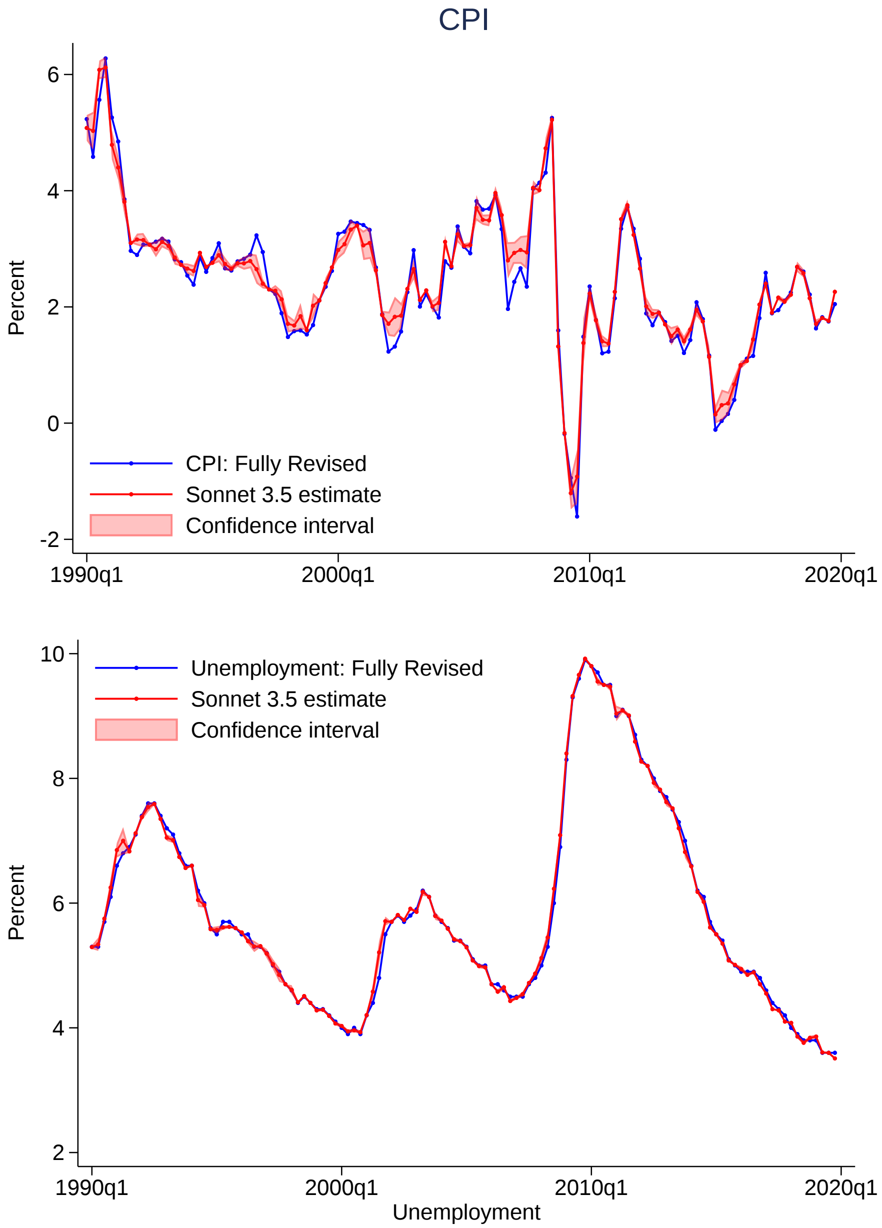
Note: LLM estimates of quarterly variables. 95% Confidence intervals based on 10 repetitions of the same query. Data go through 2025Q1, LLM estimates through 2027Q1.
Source: BLS, authors' calculations

Figure 2: LLM Recall of CPI and Unemployment



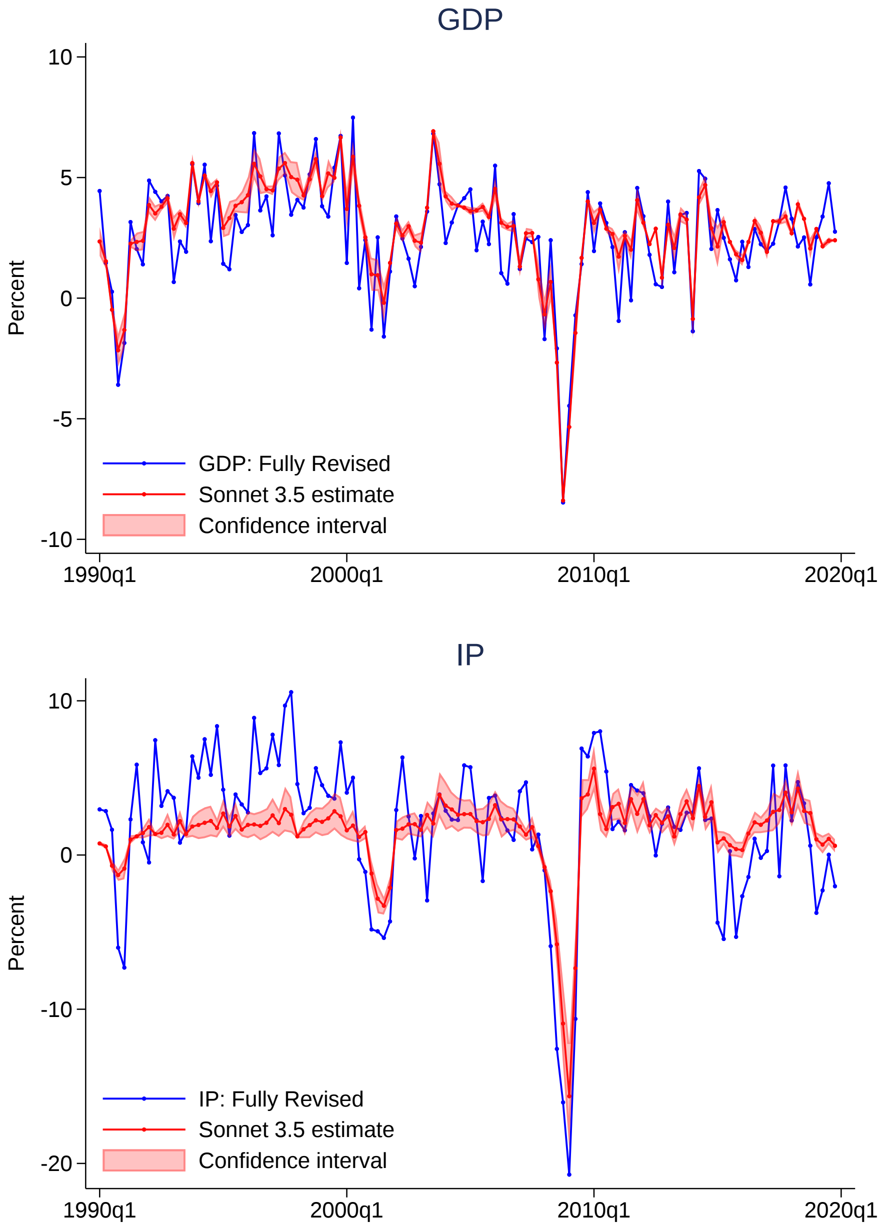
Note: LLM estimates of quarterly variables. 95% Confidence intervals based on 10 repetitions of the same query. Covid period not plotted to keep scale readable. Data go through 2025Q1, LLM estimates through 2027Q1.
Source: BEA, Federal Reserve Board, authors' calculations

Figure 3: LLM Recall of GDP and IP



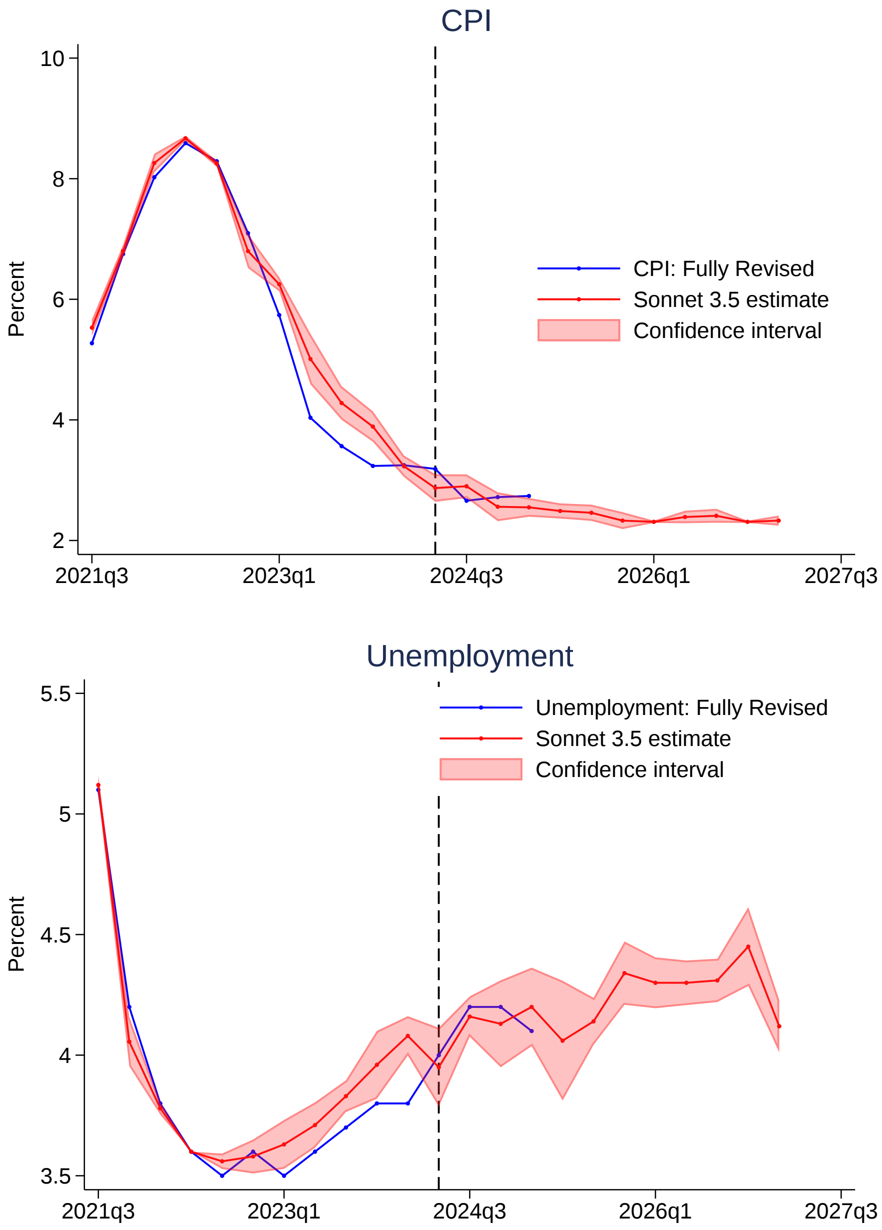
Note: LLM estimates of quarterly variables. 95% Confidence intervals based on 10 repetitions of the same query.
Source: BLS, authors' calculations

Figure 4: Pre-Pandemic Recent History: CPI and Unemployment



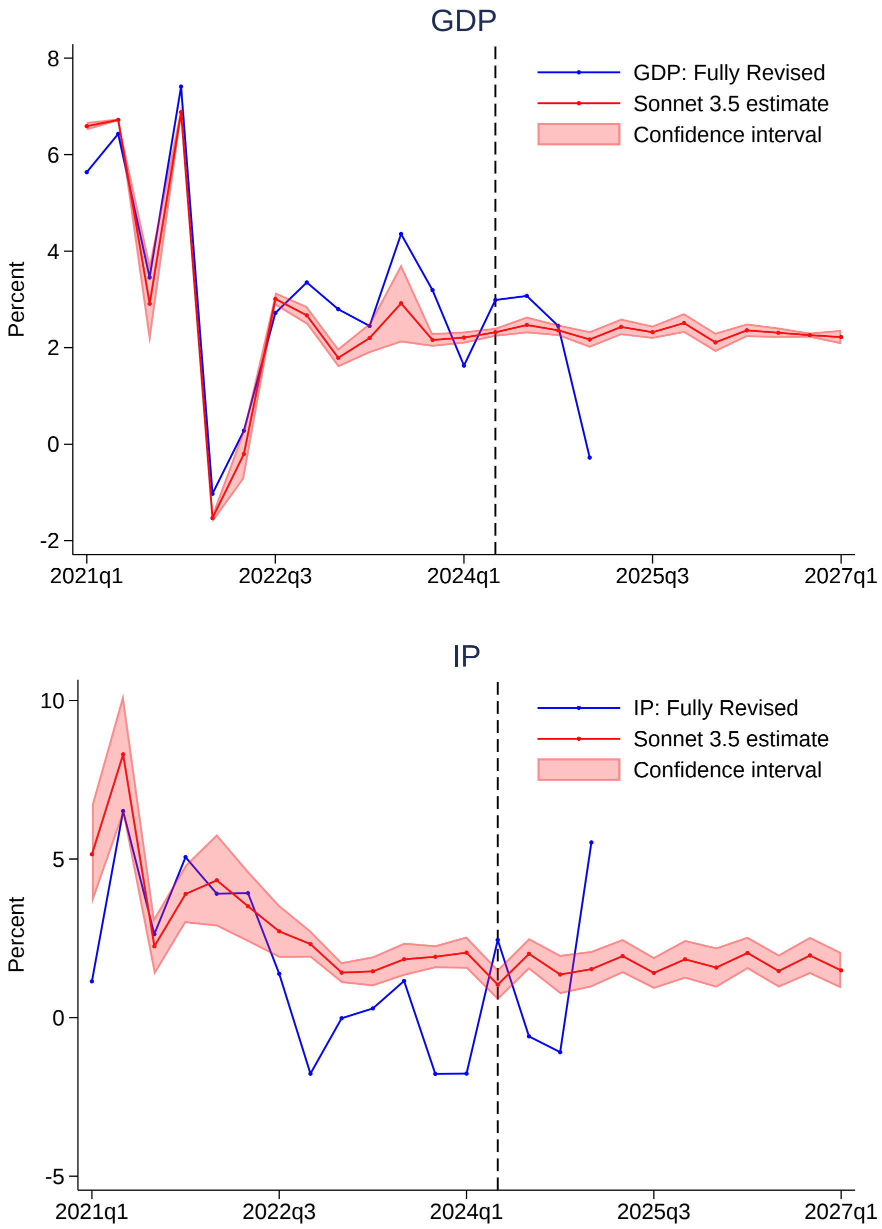
Note: LLM estimates of quarterly variables. 95% Confidence intervals based on 10 repetitions of the same query.
Source: BEA, Federal Reserve Board, authors' calculations

Figure 5: Pre-Pandemic Recent History: GDP and IP



Note: LLM estimates of quarterly variables. 95% Confidence intervals based on 10 repetitions of the same query. Vertical line shows Sonnet 3.5's knowledge cutoff (April 2024). Data go through 2025Q1, LLM estimates through 2027Q1.
Source: BLS, authors' calculations

Figure 6: Post-2021 CPI and Unemployment



Note: LLM estimates of quarterly variables. 95% Confidence intervals based on 10 repetitions of the same query. Vertical line shows Sonnet 3.5's knowledge cutoff (April 2024). Data go through 2025Q1, LLM estimates through 2027Q1.
Source: BEA, Federal Reserve Board, authors' calculations

Figure 7: Post-2021 GDP and IP

other hand, for GDP and IP the LLM misses many of the quarterly swings. The LLM tracks GDP growth throughout business cycles well and appears to become more accurate towards the end of the sample. LLM performance on IP growth is not as good; it picks up almost none of the quarterly variation and is consistently biased downward pre-2000. In addition, the confidence intervals show considerable variation in the LLM estimates.

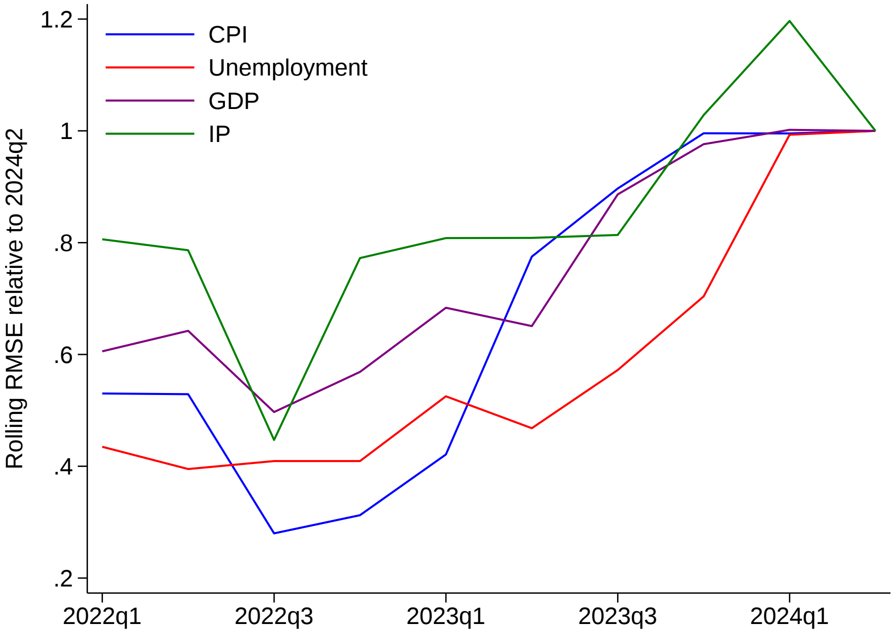
Figures 6 and 7 focus on the post-2021 period. The dashed vertical line is the knowledge cutoff for Sonnet 3.5; the date of the last training data for the model.[7] Note that Sonnet continues to provide economic estimates well after its knowledge cutoff. These estimates follow a fairly smooth trend jumping off of the knowledge cutoff and of course do not anticipate the low 2025Q1 GDP reading or the strong 2025Q1 IP reading. It appears that accuracy falls off somewhat after as the knowledge cutoff approaches; in particular, for each variable post-2023 accuracy seems noticeably worse than accuracy before that year. This is shown more clearly in Figure 8, which plots rolling six-quarter trailing root mean squared errors for each variable, normalized to unity in the 2024q2 knowledge cutoff date. The error in each LLM estimate climbs more or less steadily for the year leading up to the knowledge cutoff, suggesting that the LLM has less precise information about the period just before the cutoff. Though the sample sizes are small, the magnitude of the change in RMSEs is notable: for most variables the errors roughly double in size leading up to the cutoff. It is possible that the training data become more sparse in the months just before the knowledge cutoff, as there has been less time to collect data. In addition, while statistical press releases and news articles will always mention indicators as soon as they are available, books and academic papers discussing the economic situation will only appear months or years after the fact, constricting the amount of relevant training data.

Table 2 collects statistics for estimation error by decade. We include both the average estimation error (the bias) and the root mean squared error for each variable. The bias in

[7] This is April 2024. It is possible for the model to obtain some post-cutoff information, either through inadvertent mixing of more recent data into the training set or the implicit biases coming from the second stage "post-training" with humans who know of events after the cutoff.

	Average Error				RMSE			
Decade	GDP	CPI	Unemp.	IP	GDP	CPI	Unemp.	IP
1940s	−2.66	−0.05	−0.28	0.65	4.37	1.41	0.67	14.66
1950s	−0.53	−0.66	−0.08	3.95	3.68	1.46	0.36	13.45
1960s	−0.45	−0.28	0.00	3.47	3.12	0.38	0.11	7.41
1970s	0.01	−0.98	−0.12	2.02	3.54	1.19	0.27	7.59
1980s	−0.32	−0.59	−0.03	0.75	2.36	0.91	0.12	5.25
1990s	−0.38	−0.01	0.01	2.33	1.12	0.23	0.08	3.74
2000s	−0.37	−0.07	−0.04	−0.51	1.28	0.30	0.10	2.77
2010s	−0.24	−0.06	0.03	−0.82	1.07	0.15	0.07	2.60

Table 2: Estimation Errors by Decade



Note: Rolling 6-quarter RMSEs of the LLM estimates, normalized to unity in 2024Q2.

Source: BEA, BLS, Federal Reserve Board, authors' calculations

Figure 8: Root Mean Squared Errors

CPI and unemployment is generally small, though the LLM estimate for CPI is often 0.5-1 percentage points too high prior to the 1990s. The LLM estimate for real GDP growth has been about 0.3 percentage points too high since the 1980s. The estimates for IP show large biases, shifting from being consistently too low before 2000 to somewhat too high thereafter.

Turning to the RMSEs, we see that estimation errors are markedly higher in the early periods than in the late periods. It is tempting to attribute this to a relative lack of training data in the pre-internet era, but we need to be cautious. An alternative (but not entirely distinct) interpretation is that the LLM's estimation process is stable but underlying economic volatility was also higher in the pre-Great Moderation period, so the errors could simply reflect the fact that the series have more "noise". For example, if the LLM's estimate is approximately an N quarter moving average we would expect larger errors in more volatile periods.

4.1 Real Time Data

Economic time series often revise several times after their initial release, reflecting additional data, seasonal adjustment, and methodology changes. Fully revised data are the best retrospective estimates of what happened historically. However, data revisions rarely make much imprint in the popular press and are usually only of interest to analysts. The initial data releases garner much more interest, so it is possible LLMs will have more accurate beliefs about the initial release. In this section we focus on real GDP growth and evaluate the relationship between the initial release, fully revised data, and LLM estimates of both. We use the Philadelphia Fed's Real-Time Data Set for historical initial release values.

We modify the prompt slightly (shown in Figure 20) to explicitly ask for the first print value while continuing to instruct the LLM that it can use all of its information set. As before, we run the prompt 10 times for each quarter and average the results. While the prompt refers to the first print of GDP, reading the LLM's reasoning makes clear that it is at least partially aware that what was published prior to 1991 was Gross National Product

	Average Error		RMSE	
Decade	Fully Revised	First Print	Fully Revised	First Print
1940	-2.66	-	4.37	-
1950	-0.53	-	3.68	-
1960	-0.45	-0.86	3.12	1.62
1970	0.01	-0.65	3.54	3.11
1980	-0.32	-0.99	2.36	2.55
1990	-0.38	-0.73	1.12	1.36
2000	-0.37	-0.05	1.28	1.09
2010	-0.24	-0.32	1.07	0.55

Table 3: Summary of Estimates: GDP

(GNP), and there have been other revisions since.

For reference, Figure 9 shows both published fully revised GDP (i.e. the same series in the earlier figures) and the first print value. While the series are extremely highly correlated, the first print does diverge noticeably at times. Figure 10 shows the same comparison for the LLM estimates—first print vs. full revised. Turning to the estimation errors, Table 3 shows the average errors and RMSEs for first print and fully revised GDP. To be clear, columns 1 and 3 compare published fully revised GDP to the LLM estimate of fully revised GDP, and columns 2 and 4 compare published first print GDP to the LLM estimate of first print GDP. The average errors do not show a clear pattern. For the RMSEs, however, first print GDP seems to be estimated more accurately for most deceades, marked so in the 2010s. It is possible that the availability of online news and analysis since 2000—which might focus on first prints—has tipped the balance of training data towards the first print.

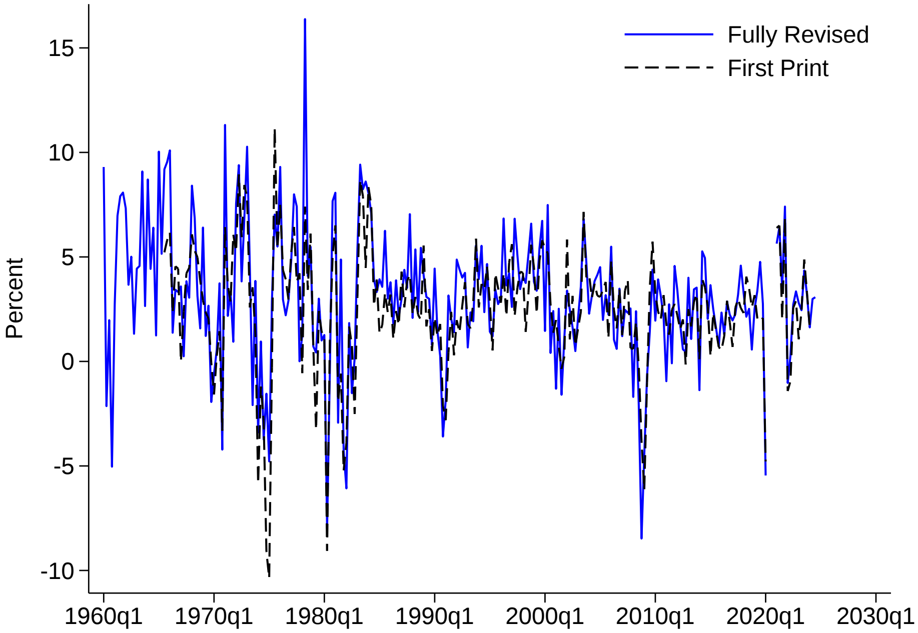
One question of interest is whether LLM estimates for fully revised data and first print data are blending together information from actual first prints with later revisions. In other words, is the LLM estimate mixing the first print and fully revised values even though we specify that the estimate should be fully revised? Table 4 shows regressions of the LLM estimate of fully revised GDP on the published first print and fully revised values. The sample period is 1980-2019. Starting from a specification with only fully revised GDP (column 2), adding first print GDP (column 3) raises the R^2 of the regression about 3.5 percentage

	(1)	(2)	(3)	(4)
Published first print GDP	0.830^{***}		0.336^{***}	0.408^{***}
	(0.072)		(0.075)	(0.066)
Published fully revised GDP		0.753^{***}	0.520^{***}	0.592^{***}
		(0.054)	(0.071)	(0.066)
Constant	1.060^{***}	0.995^{***}	0.824^{***}	
	(0.239)	(0.192)	(0.194)	
RMSE	1.538	1.358	1.280	1.399
Adjusted R^2	0.628	0.708	0.742	.

Table 4: Dependent variable: LLM estimate of fully revised GDP

	(1)	(2)	(3)	(4)
Published first print GDP	0.696^{***}		0.446^{***}	0.656^{***}
	(0.065)		(0.084)	(0.068)
Published fully revised GDP		0.573^{***}	0.263^{***}	0.344^{***}
		(0.052)	(0.072)	(0.068)
Constant	1.242^{***}	1.348^{***}	1.122^{***}	
	(0.212)	(0.184)	(0.196)	
RMSE	1.298	1.373	1.227	1.465
Adjusted R^2	0.625	0.578	0.665	.

Table 5: Dependent variable: LLM estimate of first print GDP

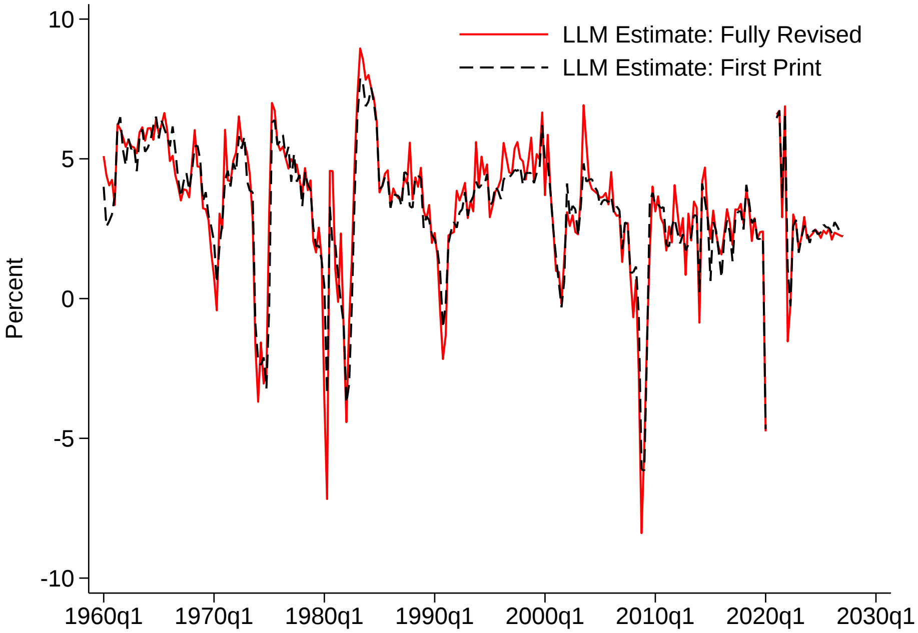


Note: First print and fully revised GDP growth
Source: BEA, Philadelphia Fed, authors' calculations

Figure 9: Published Data: Comparison of fully revised and first print real GDP growth

points. Column 3 shows that both versions of published GDP are highly statistically significant predictors of the LLM estimate and the coefficients are similar in magnitude. Put differently, the gap between fully revised GDP and the LLM estimate is correlated with first print published GDP. Column 4 forces the regression to predict the LLM estimate using a convex combination of the published numbers: we remove the constant and constrain the coefficients to add up to unity. This weighted average predictor puts similar equal weight on the two series (though somewhat more on the fully revised series), once again making the point that the LLM estimate seems to be mixture of the two.

Table 5 repeats the exercise, but uses the LLM estimate of first print GDP as the dependent variable. The pattern is largely the same: both versions of published GDP help explain the LLM estimate, and the "wrong" published series—fully revised GDP—reduces the R^2 of a regression with the "right" published series as a predictor. Both sets of results suggest



Note: LLM estimates of first print and fully revised GDP growth

Source: Authors' calculations

Figure 10: LLM Estimates: Comparison of fully revised and first print real GDP growth

the LLM is estimating historical values by—in part—smoothing across data vintages; mixing together various versions of the data that are in its training data. This is not especially surprising. An LLM with imperfect recall would naturally look to both fully revised and first print information when forming an estimate (just as a human might.) Further, LLMs are trained on enormous quantities of sometimes messy data. Even if the LLM was able to interpret and "understand" each segment of text, not all text would include clear date stamps that would signal whether the discussion of GDP was from the days after the first print or sometime later.

The mixture of first-print data with later revisions suggests that an LLM instructed to act as a real time forecaster may know "too much". Whereas an actual real time forecaster will only have access to the first-print values of the most recent GDP estimates, the LLM will (inadvertently) be working with a GDP estimate that incorporates future revisions, perhaps leading to forecasts that depend on this data leakage. Symmetrically, these results show that an LLM asked to act as retrospective analyst will not only have errors in historical recall, but those errors are partially attributable to recalling first print rather than fully revised values.

4.2 Test for Smoothing Within Vintages

The smoothing across vintages highlighted in the previous section raises issues for exercises using LLMs as real-time forecasters. A distinct form of contamination can come from smoothing data within a fixed vintage. A striking feature of Figure 5 is how much less volatile the LLM estimate is as compared to the published real activity series. This pattern is potentially consistent with the LLM returning estimates that are smoothed across time within a vintage. Abstracting from data revisions, the LLM may estimate each variable by an approximate moving average. If the moving average is two-sided this behavior would be problematic for real-time forecasting exercises, since the LLM's beliefs about "current" conditions would incorporate information from the future.

To evaluate this possibility, we use a slightly different prompt, shown in Figure 21. This prompt again asks for the first print value, but specifies that the LLM should not use information after the reference date. In particular, we explicitly instruct the LLM not to use future values of the variable (or any other variable) in making an estimate. If the LLM is able to follow these instructions the estimates should be independent of future shocks to the series.

Let y_t be the first print value of a variable for reference period t, and let $\hat{y}_t$ be an estimate based on (possibly incomplete) information available for reference periods $r \leq t$. The estimation error is

$$\varepsilon_t = y_t - \hat{y}_t. \tag{1}$$

Even if $\hat{y}_t$ is based on incomplete information, ε_t ought to be orthogonal to true shocks which occur in periods $t+1$ and later. We can approximate such shocks by using the SPF expectations for y_{t+1} as of period t, SPF^t_{t+1}. The quantity

$$\omega_{t+1} = y_{t+1} - SPF^t_{t+1} \tag{2}$$

will be the unforecastable period $t+1$ shock to y, to the extent that the SPF forecast is efficient.[8] Then our test of whether the LLM is smoothing using future data is simply a test of whether ε_t is independent of ω_{t+1}.

Table 6 shows the results for a simple test of this condition, regressing ε_t on ω_{t+1}. There is no statistically significant relationship, suggesting that there is no evidence of the LLM smoothing its estimates. Table 7 shows another specification, which controls for period t GDP and its lags (all first prints). These variables may help explain ε_t, particularly if the LLM is smoothing using lagged values. But if the LLM is not smoothing the same orthogonality condition between ε_t and ω_{t+1} should hold. We additionally control for SPF^t_{t+1}, the SPF median expectation for $t+1$ GDP growth as of quarter t. This is expected component

[8] Coibion and Gorodnichenko (2012) and others show that SPF median expectations are not necessarily rational & efficient forecasts. Nonetheless, we believe the SPF is a good approximation for these purposes.

of $t+1$ GDP growth, while ω_{t+1} is the unexpected. We see in columns 2 and 3 that there is a statistically significant relationship between the shock to $t+1$ GDP growth and the LLM's estimate of period t GDP growth. In addition, the coefficients on ω_{t+1} and SPF_{t+1}^{t} are—as expected—negative: holding GDP_t constant, stronger future GDP growth (whether expected or unexpected) leads to a stronger LLM estimate and makes ε_t more negative. The relation between ω_{t+1} and ε_t appears to be economically significant too. The bottom line of the table shows the RMSE of the regressions when ω_{t+1} is dropped; this leads to a 21 percent and 10 percent increase in the RSMEs in columns 2 and 3 respectively.

We take this as preliminary evidence of smoothing, though it is not decisive. If LLMs were predominantly smoothing the true data to form estimates we would presumably see a strong association between ε_t and ω_{t+1} even in the absence of controls. In addition, it is important to emphasize that we rely on the SPF estimates to capture all relevant period t information relevant for forecasting GDP_{t+1}. While it is known that this is not literally true—Coibion and Gorodnichenko (2012) and others document deviations from efficiency and rationality—we are comfortable with it as a baseline. To understand why, it is helpful to contrast our approach with one that focuses only on forecasts. Imagine that one evaluated LLM one-quarter-ahead real-time forecasts and found they had smaller errors than SPF forecasts. This would not be strong evidence of look-ahead bias, since it is possible that the LLM is able to synthesize relevant information (while following the information constraints) better than the SPF. Put differently, it is understood that SPF forecasts are not fully efficient so better performance by an alternative—which in some ways has far more data than any SPF participant—is not clear evidence of data leakage. In contrast, our approach is to show that the error in the LLM's recall of GDP_t is correlated with the SPF forecast error for GDP_{t+1}. If the SPF is reasonably close to efficient then we've shown that the LLM is using the unanticipated shock to GDP_{t+1} to estimate GDP_t, a clear case of look-ahead bias. On the other hand, if the SPF is not efficient and the LLM has a better forecasting methodology, then the LLM may have observe ω_{t+1} while respecting the information constraint not

	1960-2024	1960-1989	1990-2024, ex. 2020
	(1)	(2)	(3)
ω_{t+1}	-0.035	-0.109	-0.030
	(0.033)	(0.107)	(0.054)
Constant	-0.580^{***}	-0.956^{***}	-0.365^{***}
	(0.133)	(0.312)	(0.094)
Adjusted R^2	0.000	0.003	-0.005
RMSE	1.951	2.812	1.069

Table 6: Tests for Smoothing. Dependent Variable: ε_t

use information from beyond t. But the regression shows ω_t is predictably related to the LLM's recall errors, and a good forecast would eliminate errors that are correlated with the information set. What is implausible—but admittedly not impossible—is that the LLM has insight into forecasting beyond what the SPF is capable of yet still make predictable recall errors which could be solved by making use of that information. This mismatch is mostly easily explained by look-ahead bias.

5 Forecasting with LLMs

In this section we examine the forecasting performance of LLMs. We follow a methodology similar to Faria-e-Castro and Leibovici (2023), Lopez-Lira et al. (2025), and Hansen et al. (2024): ask the LLM to pretend to be a forecaster at date t, and make a forecast using only information available as of that date. In particular, we ask for 1-quarter-ahead forecasts and ask the LLM to use information available as of the 15th day of the second month of the quarter. Thus, the forecasts for 2024Q2 are made with the information in hand as of February 15, 2024. This is meant to make the results comparable to the SPF which is fielded in the second month of each quarter.

We compare the forecasts to the realizations and calculate root mean square errors.[9] Fig-

[9] We use the published values provided by the SPF. For CPI, unemployment, and industrial production these values are those available at the middle of the following quarter, and thus may be second print values. For GDP

	1960-2024	1960-1989	1990-2024, ex. 2020
	(1)	(2)	(3)
ω_{t+1}	-0.015	-0.268^{***}	-0.164^{***}
	(0.053)	(0.043)	(0.039)
SPF^{t}_{t+1}	0.092	-0.297^{***}	-0.581^{***}
	(0.167)	(0.073)	(0.086)
GDP_t	0.213^{**}	0.768^{***}	0.515^{***}
	(0.094)	(0.052)	(0.044)
GDP_{t-1}	0.043	-0.041	-0.146^{***}
	(0.061)	(0.054)	(0.038)
GDP_{t-2}	-0.018	-0.072^{*}	-0.009
	(0.038)	(0.037)	(0.017)
Constant	-1.395^{***}	-1.761^{***}	0.204
	(0.317)	(0.209)	(0.214)
Adjusted R^2	0.254	0.816	0.648
RMSE	1.692	1.209	0.635
alt RMSE w/o ω_{t+1}	1.682	1.471	0.699

Table 7: Tests for Smoothing. Dependent Variable: ε_t

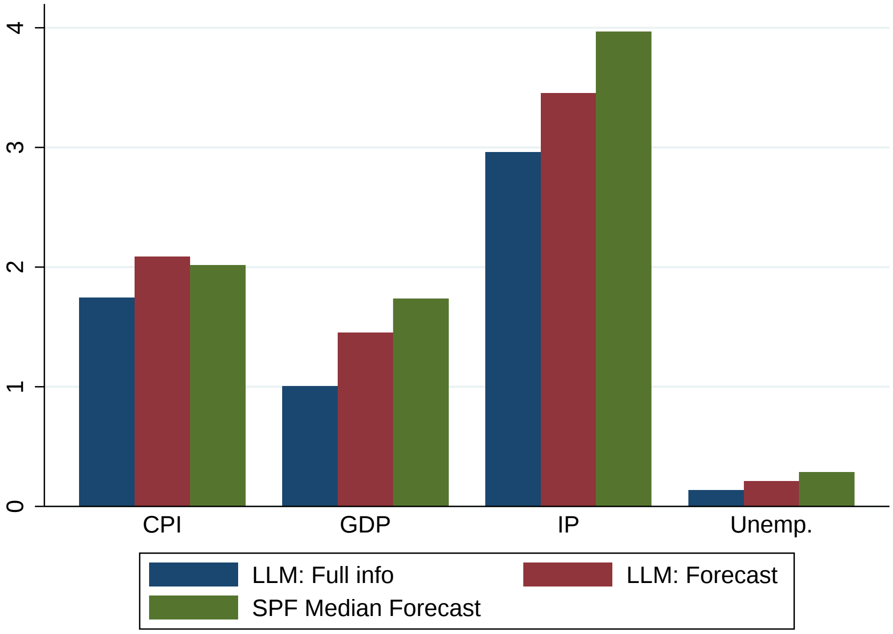
ure 11 shows the results. For comparison, we also show the RSMEs for the full information LLM estimate (i.e. the prompt which instructs the LLM to use all available information) and the SPF median. The LLM forecast RMSEs are always higher than the full information values, suggesting that the LLM is attempting to follow the prompt and not use future information in its estimate. Interestingly, the RSMEs are generally similar: asking the LLM to ignore all knowledge of the reference quarter and subsequent history only produces a modest reduction in accuracy. This is perhaps puzzling, we might expect having the LLM ignore all information from date *t* onward, including the realization of the variable, would significantly reduce accuracy. Note also that the LLM forecasts are comparable to the accuracy of the SPF and often somewhat better; if the RMSEs are valid then LLMs could be an invaluable tool for forecasting. However, the evidence of look-ahead bias in the previous sections suggests that we should not go that far—the RMSEs may be a function of the LLM drawing on data that post dates *t* but are still in the training set.

6 Recall of Release Dates

In this section we focus on the date that data were released, rather than the value of the data release. Data release dates are another useful way to assess the LLM's historical knowledge. In real time, a forecaster's information set is governed by the release dates of the relevant series. If an LLM can accurately recall the release dates of important releases it may be able to simulate a real time forecaster. If, on the other hand, the LLM has incorrect beliefs about data release dates, any attempt to simulate a real time forecaster will be problematic.

Macroeconomic data release dates are a good way to evaluate look-ahead bias for several reasons. First, the data release are important for forecasters and widely watched. Second, they are regular and can be pinpointed to a particular day, unlike other news stories which might circulate informally before breaking in major publications. Third, the release dates are not completely regular: The exact day of release depends on holidays and other factors.

the value used is the first print.



Note: Root mean square errors for 1995Q1-2019Q4

Source: Survey of Professional Forecasters, authors' calculations

Figure 11: Root Mean Square Errors

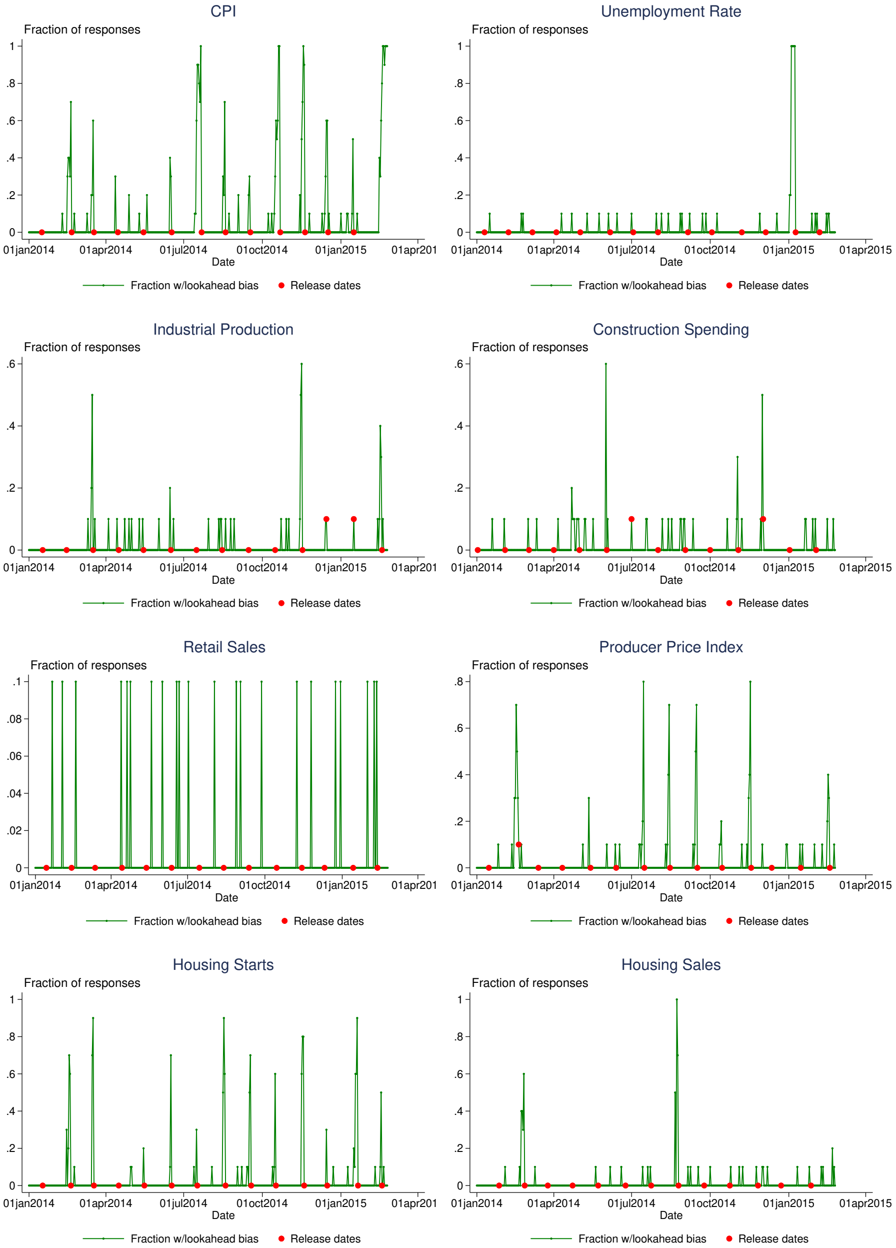
This means that to answer correctly the LLM has to know more than a simple rule, it has to recall the actual date of the release.

Our approach is as follows. For each day t, we ask the LLM to pretend to be an analyst living at 5pm on day t. Taking CPI as an example, we ask the LLM to give us the reference period of the most recent CPI release available at that time. We repeat this ten times for each data release and each day. For this section, we focus only on monthly indicators (i.e., not GDP). To build a more complete picture expand the set of series beyond CPI, IP and unemployment. We draw additional indicators mostly from the set of Principal Federal Economic Indicators (PEI); the PEI series are designated by the Office of Management and Budget and subject to rules about the release of data. In general these series are widely watched by forecasters, widely reported on, and many move markets. We do the exercise above for each indicator and for a 60 week period beginning on January 1, 2014. Example prompts are found in Section A.1. We use ALFRED[10] to get release dates for each series.

Figures 12 and 13 present daily fraction of queries that suffer from look-ahead bias: The LLM states that data has been released which in fact will only be released in the future. The green line plots this fraction, and the red dots mark actual release dates. Several things stand out. First, significant look-ahead bias is fairly rare; most series only have a handful of days where more than half of the responses indicate look-ahead bias. Second, as we might expect, look-ahead bias tends to occur in the days just before the actual data release. In other words, the LLM clearly knows approximately when the data release is supposed to happen, but sometimes misses by a couple days. This is consistent with the LLM having only fuzzy recall of the exact dates. Third, there is significant variation across series. For example, the unemployment rate suffers very little look-ahead bias, while CPI has more.

The mistakes the LLM makes appear generally sensible. For example, in early January 2015 the LLM confidently says that the December 2014 unemployment rate (i.e. the employment situation report) had been released. It turns out that the December 2014 employment

[10] https://alfred.stlouisfed.org/



CPI

Unemployment Rate

Industrial Production

Construction Spending

Retail Sales

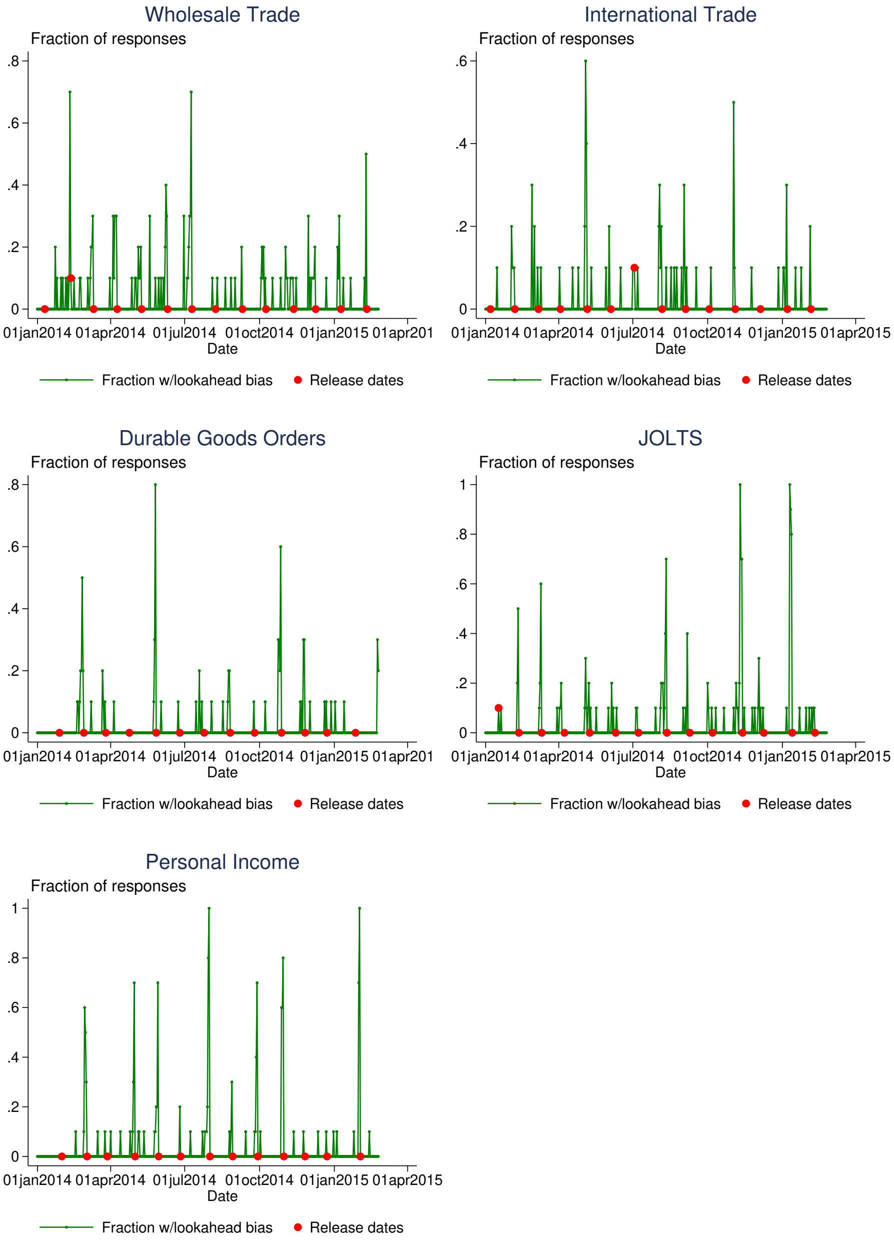
Producer Price Index

Housing Starts

Housing Sales

Note: Prompt asks LLM to state the reference date of the most recent data release as of a given day. Green line shows the fraction of responses that cited a reference date that has not been released yet. Red dots are the true data release dates.
Source: ALFRED, authors' calculations

Figure 12: LLM Recall of release dates



Note: Prompt asks LLM to state the reference date of the most recent data release as of a given day. Green line shows the fraction of responses that cited a reference date that has not been released yet. Red dots are the true data release dates.
Source: ALFRED, authors' calculations

Figure 13: LLM Recall of release dates

situation was released unusually late, on January 9th 2015. The BLS typically releases the employment report on the third Friday after the conclusion of the reference week (the week containing the 12th), which is generally in the first seven days of the release month. So in January 2015 the LLM appears to have expected the data release on the 2nd, which would have been more standard.

To summarize the exercises more compactly, we develop a metric to measure look-ahead bias across series. For each series and each day, we flag the day as problematic if more than half the queries suffer from look-ahead bias (i.e., the LLM cites a future data release as current). This is a fairly conservative criteria, as we might ask an LLM to *never* cite future data instead of lowering the bar to only half the time. Then, we count the number of days in the sample that had any problematic series among the 13 we consider. Again, this is fairly conservative as we have restricted ourselves to prominent, well-reported series. It turns out that 20.2 percent of days have at least one problematic series. This high number is the product of each series having a reasonably low proportion of problematic days (less than 7 percent for CPI, and lower for all others), but those days are *different* for each series.

A 20.2 percent error rate should give us pause. While the results for any single series are impressive, an LLM pretending to be a real time forecaster would make frequent mistakes. From the perspective of historical analysis, an LLM may not reliably recall the details of real time data flow during historical episodes, limiting the reliability of historical analysis.

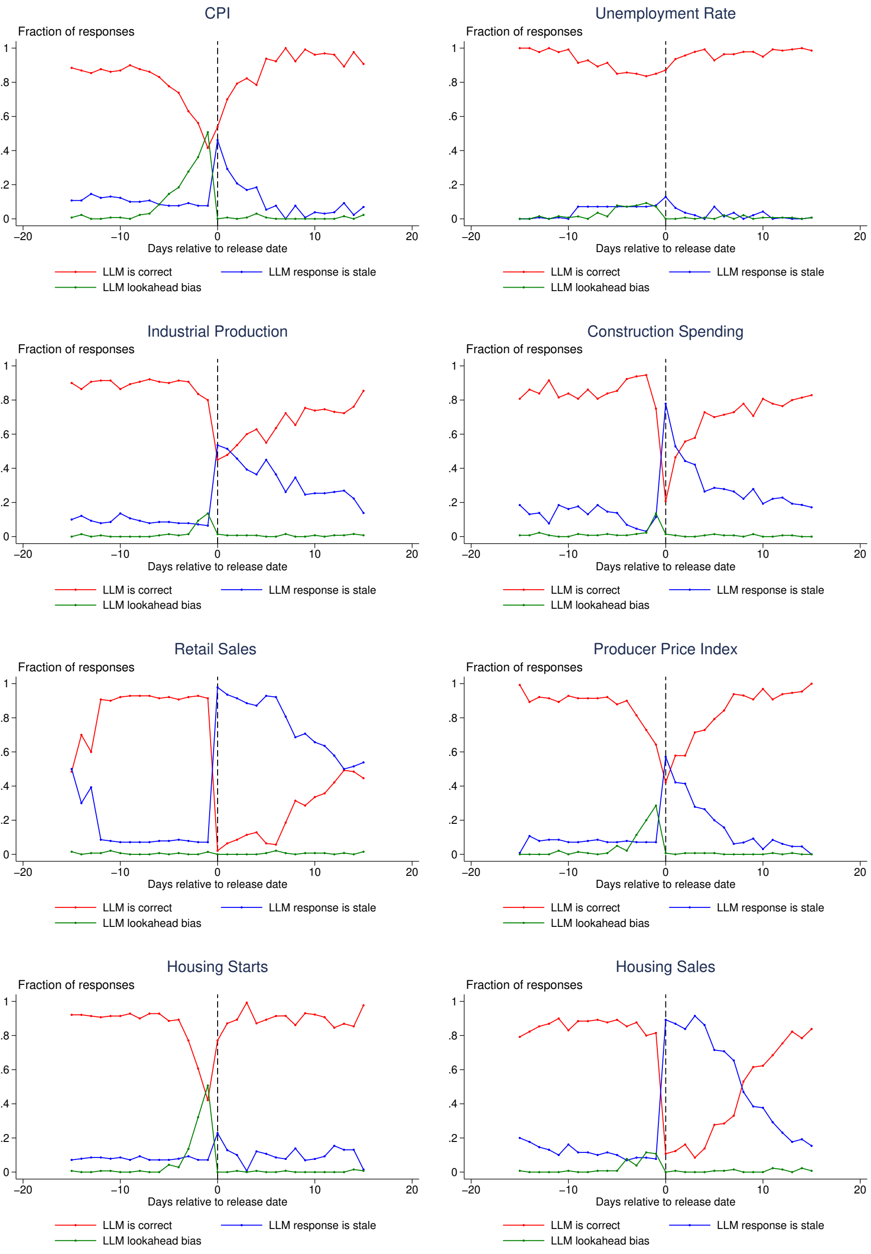
To further examine LLM performance we renormalize the data, averaging performance in a window around data releases and plot performance measures for 15 days on either side of data releases. Figures 14 and 15 plot the results. The x-axis counts days before and after a data release. The red line shows the fraction of LLM responses for a particular (relative) date that is correct. The green line shows the fraction suffering from look-ahead bias: the response cites data that have not been released yet. The blue line shows the fraction making the opposite error: the LLM cites an old data release when a newer one is available. Note that the lines all add up to one.

Many of the patterns from the other charts are apparent here: look-ahead bias peaks in the days just before a data release, and there is often very different behavior across series. Unemployment, in particular, is recalled very accurately. This may be because unemployment (and the Employment Situation report) is very widely reported. In addition, the Employment Situation is almost always released on the third Friday after the end of the week of the 12th, which in turn is generally the first Friday after the reference month. This regularity likely assists with accurate recall.

Interestingly, "look-behind bias"—citing stale data—is fairly common. Series such as construction spending, the PPI, and housing sales all have big spikes in citing stale data in the days after a data release. This highlights the multiple risks from using LLMs to understand real-time phenomena: While they may engage in data peeking they also may fail to properly update their information sets. Along some dimensions, these error might roughly offset, leading to decent forecasting performance that is a mix of look-ahead bias contamination and stale data.

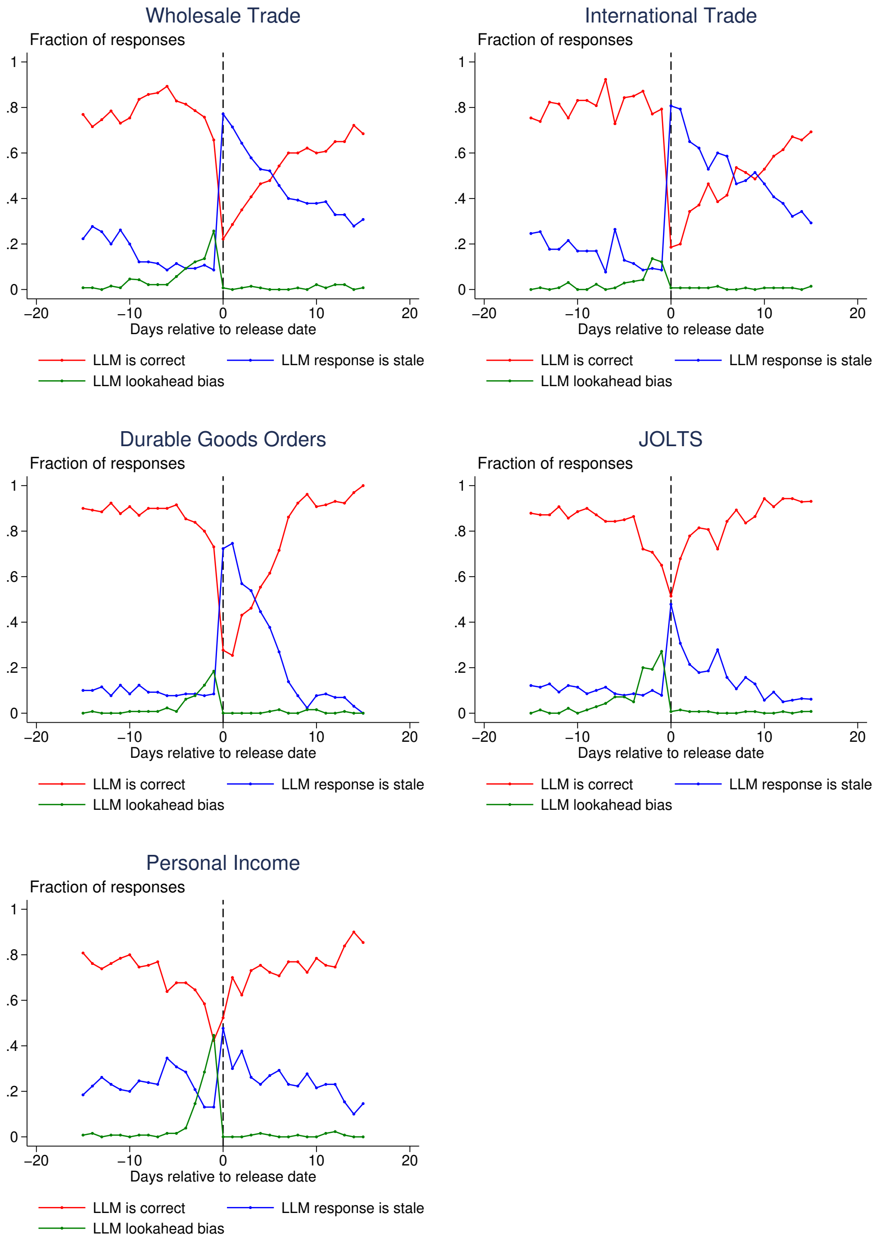
From Figures 14 and 15 it is apparent that "look-behind bias" is more common than look-ahead bias. An examination of the LLM responses shows that the LLM sometimes states it is being "conservative", in the sense of only saying a data release has occurred when it is very sure that is the case. Note that while the prompt did not ask the LLM to be conservative in this sense, it is apparently imposing an asymmetric penalty on itself.

To explore this further we change the prompt to explicitly tell the LMM to not be conservative: if it is 51% sure that a new data release is available, that should be the answer, if it is only 49% certain than it should not be. The full prompt is in Figure 25. We run the new prompt for CPI and construction spending, see Figures 16 and 17. For construction spending, the new prompt is a big improvement in accuracy: While the fraction of responses suffering look-ahead bias increases, accuracy is higher and look-behind bias is lower. The fractions of look-ahead and look-behind bias are roughly equal as we would expect with a symmetric penalty.



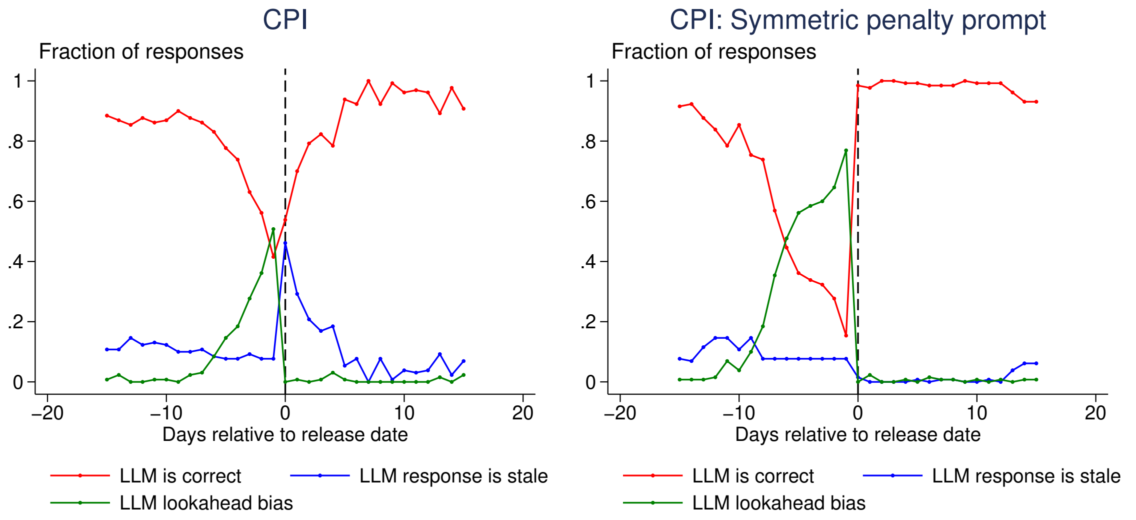
Note: Prompt asks LLM to state the reference date of the most recent data release as of a given day. Results are normalized so that the true data release is on day 0 and then averaged.
Source: ALFRED, authors' calculations

Figure 14: LLM Recall of release dates: Relative to release date



Note: Prompt asks LLM to state the reference date of the most recent data release as of a given day. Results are normalized so that the true data release is on day 0 and then averaged.
Source: ALFRED, authors' calculations

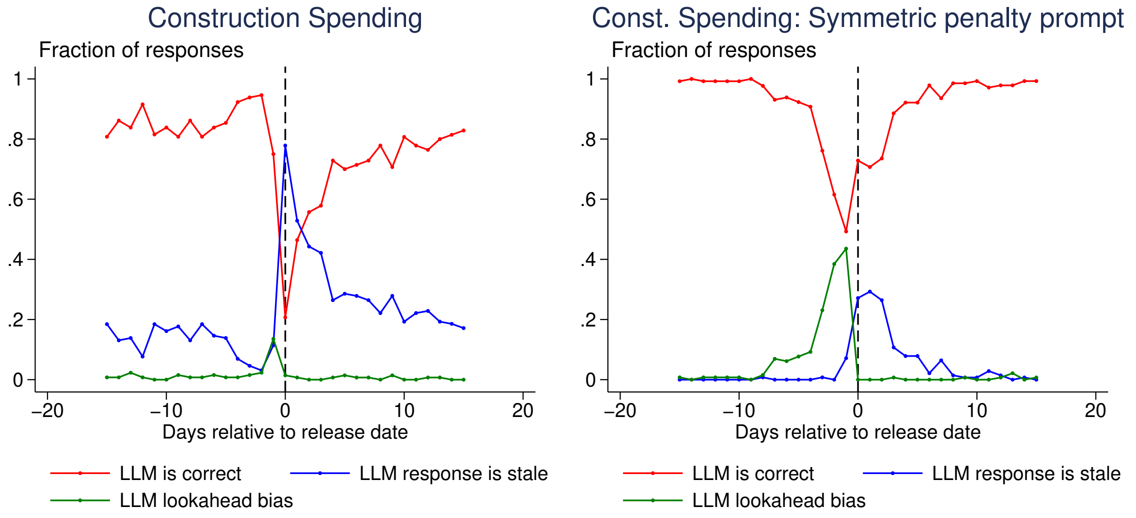
Figure 15: LLM Recall of release dates: Relative to release date


CPI

CPI: Symmetric penalty prompt

Note: Prompt asks LLM to state the reference date of the most recent data release as of a given day. Results are normalized so that the true data release is on day 0 and then averaged.
Source: ALFRED, authors' calculations

Figure 16: Effect of alternative prompt on CPI


Construction Spending

Const. Spending: Symmetric penalty prompt

Note: Prompt asks LLM to state the reference date of the most recent data release as of a given day. Results are normalized so that the true data release is on day 0 and then averaged.
Source: ALFRED, authors' calculations

Figure 17: Effect of alternative prompt on construction spending

The story is different for CPI. Originally, CPI had roughly equal look-ahead and look-behind biases. Under the new prompt, look-ahead bias worsens, look-behind bias is almost nonexistent, and accuracy is lower. Thus, it appears that the new prompt does not necessarily improve accuracy; instead, it trades off more look-ahead bias for less look-behind bias and the outcome depends on the initial balance.

7 Conclusion

LLMs are becoming important tools for economic analysis. Our results paint a complicated picture of current capabilities and shortcomings. We find that current LLMs have excellent retrospective knowledge of some macro variables (like CPI and the unemployment rate), but much noisier knowledge of GDP and IP growth. LLM recall of data release dates is also impressively accurate, but still suffers from noise; and the noise accumulates as more series are considered.

Our results point to problems when LLMs are used as real-time forecasters and evaluated before the knowledge cutoff. Fuzzy knowledge of data release dates, LLM estimates that smooth future data values into current estimates, and the mixture of first-print values with later revisions all suggest look-ahead bias contaminates LLM forecasts. This raises the question of why—if LLMs have significant look-ahead bias—LLM forecasts are only modestly better than SPF forecasts. Our evidence suggests that LLMs may be both too good and too bad: the fuzzyness and imperfect recall that lead to look-ahead bias also *limit* forecast accuracy, since the LLM has limited recall of both the true target value and the historical variables it might use as predictors. These offsetting errors may leave LLMs with in-sample forecasts that are good, but not implausibly good. Some of these issues may be attenuated by more sophisticated prompting strategies, such as providing more information to ground the LLM. We leave exploration of these margins for future work.

References

Bybee, J. Leland, "The Ghost in the Machine: Generating Beliefs with Large Language Models," 2023.

Cajner, Tomaz, Leland D. Crane, Christopher J. Kurz, Norman J. Morin, Paul E. Soto, and Betsy Vrankovich, "Manufacturing Sentiment: Forecasting Industrial Production with Text Analysis," Finance and Economics Discussion Series 2024-026, Board of Governors of the Federal Reserve System (U.S.) May 2024.

Coibion, Olivier and Yuriy Gorodnichenko, "What Can Survey Forecasts Tell Us about Information Rigidities?," *Journal of Political Economy*, 2012, *120* (1), 116–159.

Cook, Thomas R., Sophia Kazinnik, Anne Lundgaard Hansen, and Peter McAdam, "Evaluating Local Language Models: An Application to Bank Earnings Calls," Research Working Paper RWP 23-12, Federal Reserve Bank of Kansas City November 2023.

Croushore, Dean, "Frontiers of real-time data analysis," *Journal of economic literature*, 2011, *49* (1), 72–100.

Faria-e-Castro, Miguel and Fernando Leibovici, "Artificial Intelligence and Inflation Forecasts," Working Papers 2023-015, Federal Reserve Bank of St. Louis July 2023.

Federal Reserve Bank of St. Louis , "ALFRED, Archival Federal Reserve Economic Data," https://alfred.stlouisfed.org/.

Glasserman, Paul and Caden Lin, "Assessing Look-Ahead Bias in Stock Return Predictions Generated By GPT Sentiment Analysis," 2023.

Hansen, Anne Lundgaard, John J. Horton, Sophia Kazinnik, Daniela Puzzello, and Ali Zarifhonarvar, "Simulating the Survey of Professional Forecasters," November 15 2024. SSRN Working Paper.

He, Songrun, Linying Lv, Asaf Manela, and Jimmy Wu, "Chronologically Consistent Large Language Models," Working paper 2025.

Jha, Manish, Jialin Qian, Michael Weber, and Baozhong Yang, "ChatGPT and Corporate Policies," Working Paper 32161, National Bureau of Economic Research February 2024.

Kazinnik, Sophia, "Bank Run, Interrupted: Modeling Deposit Withdrawals with Generative AI," 2024.

Kim, Alex G., Maximilian Muhn, and Valeri V. Nikolaev, "Financial Statement Analysis with Large Language Models," 2024.

Korinek, Anton, "Generative AI for Economic Research: Use Cases and Implications for Economists," *Journal of Economic Literature*, January 2023, *61* (4), 1281â1317.

Lopez-Lira, Alejandro, Yuehua Tang, and Mingyin Zhu, "The Memorization Problem: Can We Trust LLMs' Economic Forecasts?," *arXiv preprint arXiv:2504.14765*, 2025.

Ludwig, Jens, Sendhil Mullainathan, and Ashesh Rambachan, "Large Language Models: An Applied Econometric Framework," 2025.

Manning, Benjamin S., Kehang Zhu, and John J. Horton, "Automated Social Science: Language Models as Scientist and Subjects," 2024.

Ouyang, Shuyin, Jie M Zhang, Mark Harman, and Meng Wang, "An empirical study of the non-determinism of chatgpt in code generation," *ACM Transactions on Software Engineering and Methodology*, 2025, *34* (2), 1–28.

Pham, Van and Scott Cunningham, "Can Base ChatGPT be Used for Forecasting without Additional Optimization?," 2024.

Phan, Long, Adam Khoja1, Mantas Mazeika, and Dan Hendrycks, "LLMs Are Superhuman Forecasters," 2024.

Research Department, Federal Reserve Bank of Philadelphia, "Survey of Professional Forecastors," https://www.phil.frb.org/research-and-data/real-time-center/survey-of-professional-forecasters/.

Sakar, Suproteem and Keyon Vafa, "Lookahead Bias in Pretrained Language Models," 2024.

Sarkar, Suproteem, "StoriesLM: A Family of Language Models With Time-Indexed Training Data," Mar 2024. Available at SSRN: https://ssrn.com/abstract=4881024.

Schoenegger, Philipp, Peter S. Park, Ezra Karger, and Philip E. Tetlock, "AI-Augmented Predictions: LLM Assistants Improve Human Forecasting Accuracy," 2024.

Shapiro, Adam Hale, Moritz Sudhof, and Daniel J. Wilson, "Measuring news sentiment," *Journal of Econometrics*, 2022, *228* (2), 221–243.

Tranchero, Matteo, Cecil-Francis Brenninkmeijer, Arul Murugan, and Abhishek Nagaraj, "Theorizing with Large Language Models," 2024.

van Binsbergen, Jules H, Svetlana Bryzgalova, Mayukh Mukhopadhyay, and Varun Sharma, "(Almost) 200 Years of News-Based Economic Sentiment," Working Paper 32026, National Bureau of Economic Research January 2024.

Wei, Jason, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou et al., "Chain-of-thought prompting elicits reasoning in large language models," *Advances in neural information processing systems*, 2022, *35*, 24824–24837.

Zarifhonarvar, Ali, "Evidence on Inflation Expectations Formation Using Large Language Models," 2024.

A Prompts

```
You are a resourceful and knowledgeable economic analyst, with deep knowledge
of macroeconomic data and forecasting.  Think step-by-step, writing out your
reasoning, and only write your final answer at the end of your response.
```

Figure 18: Main System Prompt

```
Based on all knowledge available to you, tell me the fully-revised value
of {var} for {reference_quarter}. This should be the value after all
subsequent revisions, not necessarily as initially released.
Do not try to forecast revisions that occur beyond your knowledge cutoff.
If you are unsure, make an estimate based on what you know. Please give me
a numeric point estimate, not a range. Use all of your powers of analysis
and use all of the information you have available to you.
```

Figure 19: Prompt for Fully-Revised Data

```
Based on all knowledge available to you, tell me the first print value
of {var} for {reference_quarter}. This should be the value as initially
released without any subsequent revisions. If you are unsure, make an
estimate based on what you know. Please give me a numeric point estimate,
not a range. Use all of your powers of analysis and use all of the
information you have available to you.
```

Figure 20: Prompt for First Print Data

```
Tell me the first print value of {var} for {reference_quarter}. This should
be the value as initially released without any subsequent revisions.  If you
are unsure, make an estimate based on what you know, but do not base your estimate
on any data for reference periods after {reference_quarter}.  In particular, don’t
use values of {var} from after {reference_quarter} in constructing your estimates.
Please give me a numeric point estimate, not a range. Use all of your powers of
analysis and use all of the information you have available to you, subject to the
constraints above.
```

Figure 21: Prompt for First Print Data—Real Time Information Set

```
You are a specialist in extracting information from the output of other Large
Language Models. You are succinct in your responses and response with exactly
what is asked of you.
```

Figure 22: Summarization System Prompt

```
I have the following output from a large language model:

{llm_output}

This piece of text is an economic forecast. I want you to summarize and extract
the prediction for the economic variable mentioned in the following format for
me, please:

"answer: {...}"

Please replace the placeholders denoted with {...} with the answer over the
requested quarter only. PLEASE ONLY put ONE NUMBER in that location. Refrain
from reporting a range of values; please try to report a single value.

Please only return this format with the right value and NO additional text. Thank
you!
```

Figure 23: Summarization Prompt

A.1 Prompts for Data Release Dates

```
Assume that it is 5pm, close of business on {month} {day_of_month}, {year}.  Tell me
the most recent month for which BLS has released any CPI estimate, i.e. the reference
month for the most recent release on or before {month} {day_of_month}, {year}.  Briefly
explain your reasoning and only give your answer at the end. Give an exact month,
no ranges. Make an estimate if you have to.  When you give your answer, give it in
year, "M", month format,  i.e. 2035M2 for the February 2035, or 2001M11 for the
November 2001. Make sure the final answer is given exactly in that format.
```

Figure 24: Prompt for CPI Release Date

```
Assume that it is 5pm, close of business on {month} {day_of_month}, {year}.
Tell me the most recent month for which BLS has released any CPI estimate, i.e. the
reference month for the most recent release on or before {month} {day_of_month},
{year}.  Briefly explain your reasoning and only give your answer at the end. Give
an exact month, no ranges. Make an estimate if you have to.  It is equally bad to
make mistakes in either direction: if you think there is a 51 percent chance the
more recent release has occured, that should be your answer.  If you think there is
only a 49 percent chance the more recent release has occured, it should not be
your answer.  Do not be "conservative", we only care about raw accuracy.   When
you give your answer, give it in year, "M", month format, i.e. 2035M2 for the
February 2035, or 2001M11 for the November 2001. Make sure the final answer is
given exactly in that format.
```

Figure 25: Prompt for CPI Release Date, Risk Neutral Version

```
You are a resourceful and knowledgeable economic analyst, with deep
knowledge of macroeconomic data and forecasting.  Think step-by-step, writing
out your reasoning, and only write your final answer at the end of your response.
```

Figure 26: Data Release Dates, System Prompt

```
You are a specialist in extracting information from the output of
other Large Language Models. You are succinct in your responses and response with
exactly what is asked of you.
```

Figure 27: Data Release Dates Summarizer, System Prompt

```
I have the following output from a large language model:

{llm_output}

This piece of text contains a monthly date as an answer to a question. I want you to
extract the date, which should be given as a year followed by a "M" followed by a month
number with no spaces.  Examples would be 2021M1, or 2035M11.  Convert the date to that
format if needed.  Reply only with following format, please:
"answer: {...}"

Please replace the placeholders denoted with {...} with the data.
Refrain from reporting a range of values; please try to report a single value.
Make sure you extract the correct date; other dates might be discussed in the passage
but make sure to extract the one given as the answer.

Please only return this format with the right value and NO additional text. Thank
you!
```

Figure 28: Data Release Dates Summarizer Prompt