Assessing the Possibility and Feasibility of Predicting Major US Financial Bubbles Using Deep Learning, LLM, and DL-LLM Architectures Analyzing Macro-Financial Data

Junbum Cho

Received July 28, 2025
Accepted October 05, 2025
Electronic access October 15, 2025

Predicting financial bubbles is essential for preventing market disruptions and economic losses, encouraging the development of various technical approaches for detection. As of July 2025, Deep Learning (DL) models have become the dominant approach for such financial prediction, while LLM research remains limited to basic tasks such as text summarization and question-answering. This paper builds on recent LLM developments to assess AI architectures' feasibility for financial bubble prediction. Three architectures are evaluated: (i) Standalone DL models with BiLSTM and Transformer backbones; (ii) Standalone LLMs using three frontier models (GPT-5, Grok-4, and Gemini 2.5 Pro); and (iii) DL-LLM hybrid architectures pairing each DL backbone with LLM heads. These architectures are compared across three evaluation approaches: (1) bubble-only periods for architectural configuration; (2) non-bubble-only periods; and (3) mixed regimes incorporating both bubble and non-bubble periods. All architectures generate episode-level bubble probabilities evaluated using the Brier score as a proper, threshold-free metric.
Results demonstrate that Standalone LLMs achieve optimal performance in single-regime settings (Brier scores: bubble-only 0.230, non-bubble-only 0.278). In mixed-regime evaluations, Standalone DL with BiLSTM backbone delivers superior probabilistic accuracy (0.179), followed closely by DL-LLM with BiLSTM backbone (0.183). Transformer-based variants underperform significantly in mixed settings (0.60–0.70). Analysis reveals that DL-LLM hybrid outputs are strongly anchored to their DL backbone components. While macro-financial trend features alone prove insufficient for uniformly accurate bubble detection, the findings suggest that more precise computational bubble definitions could enhance LLM-based prediction methods.

1 Introduction

1.1 Destructive Nature of Bubbles and Technical Approaches to Minimize Their Effects

Financial bubbles are recurring economic phenomena where asset prices inflate far beyond intrinsic value before collapsing catastrophically[1]. These cycles consistently devastate economies, wiping out trillions in wealth and triggering severe recessions[2]. Historical examples from the 1637 Tulip Mania[3] to the early 2000s Dot Com bubble demonstrate their capacity for widespread economic destruction[4].

Deep Learning (DL) and Machine Learning (ML) models have dominated financial bubble prediction, consistently employing binary classification frameworks to distinguish bubble from non-bubble states across markets including Indian equity[6], Vietnamese stocks[7], and S&P 500[8]. Recent advances demonstrate that contrastive and self-supervised learning techniques represent the current state-of-the-art[9–12] showing improved performance through enhanced label efficiency and superior generalization capabilities.

However, a critical gap exists in applying Large Language Models (LLMs) to quantitative financial datasets for bubble prediction. Current LLM research in finance focuses largely on unstructured text and audio data—earnings transcripts, news articles—for sentiment analysis and general risk assessment[13–17]. Even in multimodal LLM research, finance-specialized mod-

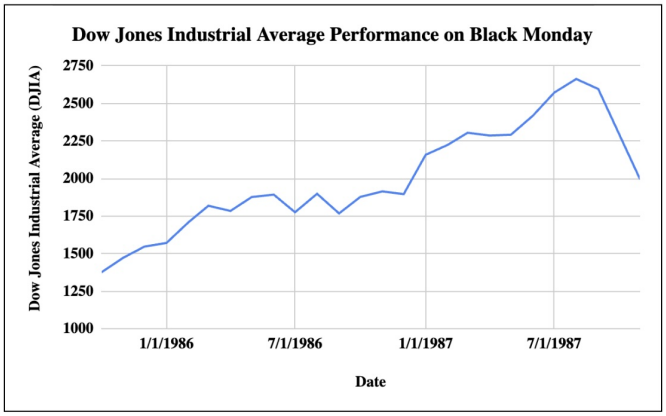

Fig. 1 Dow Jones Industrial Average from January 1986 to late 1987, showing an upward trend through mid-1987 followed by the sharp Black Monday crash in October 1987[5].

els like FinTral perform conventional tasks such as chart/table understanding, report Q&A, KPI extraction, and cross-modal reasoning over price charts—as evaluated by benchmarks such as MME-Finance and FCMR. Bubble identification on raw market windows remains underexplored[18].

This limitation overlooks LLMs' potential for direct quantitative analysis, particularly given recent advances in inference and reasoning capabilities through Chain of Thought (CoT) methods[19].

This study addresses this gap by investigating LLMs' capability to predict financial bubbles directly from quantitative market data, leveraging enhanced reasoning capabilities that traditional ML approaches cannot replicat[20]. This represents the first systematic exploration of LLM-based quantitative bubble prediction, potentially establishing a new paradigm in financial forecasting.

1.2 Research Objective

This paper aims to identify the possibility and the most feasible AI architecture in predicting financial bubbles, constrained within the United States financial market. To do so, the research evaluates three AI architectures specialized for American bubble prediction since the 1960s: a standalone DL model, a standalone LLM, and the DL-LLM architecture that integrates both models.

The three architectures were developed and evaluated under three distinct approaches. The first approach utilized bubble data exclusively. The DL model was trained on macro-financial data from bubble periods, while the LLM was fed with the same data. During evaluation, both architectures predicted the bubble probability on unseen bubble and non-bubble data. The DL-LLM architecture combined the two by enabling the LLM to interpret DL outputs through sophisticated prompts. The second approach utilized only the non-bubble data for DL and LLM configuration, also letting the architectures predict the bubble probability on unseen bubble and non-bubble data. The third approach treats non-bubble periods as noise for bubble detection, combining the bubble and non-bubble data to configure and evaluate the three architectures.

2 Methodology

2.1 Overview

First, macro-financial data from the U.S. bubble and non-bubble periods were collected. Then, different versions of the DL model were trained: a version trained using only bubble period data, only the non-bubble data, and both the bubble and non-bubble data. Subsequently, different versions of standalone LLM prompts were developed to predict bubble probability under the three different approaches. Finally, different versions of the DL-LLM architecture prompts were developed to enable the LLM to interpret DL model outputs and predict bubble probability.

All materials are available at
`https://github.com/Junbum-Cho/US-Financial-Bubble-Prediction-DL-LLM-and-DL-LLM-Code-Prompts-Raw-Data.git`
including: (1) the exact prompt templates used for the Standalone LLM and DL-LLM setups; (2) training code for the Standalone DL models (BiLSTM and Transformer backbones); (3) Raw Data 1—a single Google Doc consolidating per-episode, per-approach tables of all numeric outputs from the three architectures; and (4) Raw Data 2—a Google Drive folder of ChainForge run logs (e.g., Method, Prompt, Response, Batch ID, Var: target/Bubble_Prototypes, Metavar, fully specifying each experiment.

2.2 Computational Environment for Configuring the Standalone DL Model, Standalone LLM, and DL-LLM System

Table 1 Development Platforms and Tools.

Name	Description
Google Colab (Colaboratory)	Google Colab (Colaboratory) is a cloud-based Jupyter Notebook environment that allows users to write and execute Python code in their browser[21] In this research, Google Colab was utilized to train and run the DL models.
Python	Python, a versatile programming language suitable for data science[22] was used for the training and running of the DL model.
ChainForge	ChainForge is an open-source, visual data-flow tool for prompt engineering and rapid hypothesis testing with LLMs[23]. This platform was utilzed to build the running environment for the Standalone LLM and the DL-LLM architectures with different file configurations.
ChatGPT-5, Grok-4, Gemini 2.5 Pro	These three LLMs were chosen for building the standalone and DL-LLM architectures. Specific Model identifiers and versions are specfieidc in section 2.6.
Macbook	A MacBook Pro (14-inch, 2021) equipped with an Apple M1 Pro chip featuring a 10-core CPU (8 performance cores + 2 efficiency cores), 14-core GPU, 16-core Neural Engine, and 32GB unified LPDDR5 memory running macOS 15.2 (24C101) was utlized for this entire research.

2.3 Obtaining the Bubble and Non-Bubble Macro-Financial Data For Training

2.3.1 Definition of a Bubble

This research follows the most widely accepted definition of a bubble: when an asset's market price exceeds its fundamental value—most commonly operationalized as the present value of expected future cash flows, without necessarily including the price plummeting process[24].

This bubble definition serves as the framework for selecting appropriate bubble timelines in the training dataset, as discussed in Section 2.3.2.

2.3.2 Selecting Bubble and Non-Bubble Periods and Obtaining Their Macro-Financial Dataset

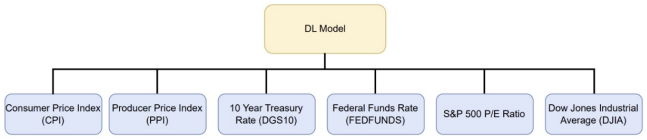

Fig. 2 Six Macroeconomic and Financial Indicators Utilized for DL Models' Training.

As specified in Fig. 2, these six economic and financial indicators were chosen based on their historical relevance to bubble incidents, either as direct or indirect bubble indicators.

Meanwhile, following the bubble definition explained in section 2.3.1, the time periods for the four bubble incidents were chosen to best reflect the formation, overheating, and peak phases of each bubble, ensuring coverage of at least 2 years of macro-financial data to provide sufficient datasets for training and testing. Similarly, ensuring the minimal 2-year criteria, non-bubble periods were selected to avoid overlapping with bubble incidents and to demonstrate relatively healthy financial market conditions. The chosen bubble and non-bubble periods, timelines, and justifications with references are shown in Table 2 and Table 3.

Data were organized into CSV files (detailed in Table 5) for DL model training, or restructured into JSON files containing natural language summaries for Standalone LLM and DL-LLM architectures. During JSON preparation, generic identifiers such as 'Bubble_Prototype_1' and 'Target_Data' replaced original filenames to prevent filename-based bias during LLM bubble prediction, as detailed in Section 2.6.

Once the target bubble and non-bubbles and their respective timelines were confirmed, the macro-financial datasets for each were obtained directly from the following sources: U.S. Department of Labor Bureau of Labor Statistics (BLS), Federal Reserve Bank of St. Louis, and S&P Dow Jones Indices LLC. Direct links to the each economic sources, encompassin all the bubble and non-bubble periods, are organized in table 4 below.

The obtained data were eventually compiled into a single CSV (Comma Separated Values) file, as illustrated in Table 5.

Table 2 Major US Bubble Incidents, Selected Timeline, and Justification.

Bubble Incident	Selected Timeline	Justification
Nifty Fifty	1970/01/01 - 1973-04-01	S&P 500 gained 10.79% in 1971 and 15.63% in 1972, where the euphoria was concentrated in about 50 "blue-chip" stocks[25]. By late 1972, Polaroid's Price-to-Earnings (P/E) ratio hit 91, and Avon's reached 65[26].
Black Monday Stock Surge	1982/08/01 - 1987/09/01	The DJIA rose from 776 in August 1982 to a peak of 2,722 in August 1987[27], achieving a 44% gain in the first eight months of 1987 alone[28]. On October 19, 1987, the DJIA fell by a record 22.6% in a single day[29].
Dot Com	1997/11/01 - 2000/02/01	The NASDAQ Composite index's P/E ratio hit an unprecedented 200, showing an exponential growth crossing 3,000 in November 1999 and 4,000 in just two months later in December 1999[30].
Subprime Housing Crisis	2004/01/01 - 2006/07/01	The S&P/Case-Shiller Home Price Index shows back-to-back years of double-digit growth: 12.8% in 2004 and 12.5% in 2005[31].

During collection, missing macro-financial data points were left blank, and this approach was maintained consistently throughout training and evaluation. All DL models, LLMs, and the DL-LLM architectures used the six selected macro-financial indicators organized in this format, where subsequent processes are detailed in the following sections.

2.4 Configuration and Evaluation Framework of Standalone DL Model, Standalone LLM, and the DL-LLM System

2.4.1 Overview

The configuration of the three distinct AI architectures (stan-

Table 3 Major US Non-Bubble Timeline Selected and Justifications

Non-Bubble Period	Selected Timeline	Justification
Early 1960s Economic Expansion	1962/01/01 - 1965/12/01	The United States achieved a real GDP growth of 6.1% in 1962, 4.4% in 1963, 5.8% in 1964, and 6.5% in 1965. Meanwhile, the annual inflation rate was just 1.2% in 1962 and 1963, 1.3% in 1964, and 1.6% in 1965. Based on such a 'golden age' for the U.S. economy, the stock market also remained stable[32].
The Late 1970s Expansion	1975/01/01-1/1/1980	The United States experienced a 58-month economic expansion from March 1975 to January 1980, with real GDP growth averaging 4.3% annually and unemployment declining from 9% to 6%. However, unlike the low-inflation growth of the 1960s, this expansion was plagued by persistent inflation averaging over 8%, ultimately leading to its end in January 1980 as the Federal Reserve tightened monetary policy to combat rising prices[33].
Post-Black Monday Recovery	1987/11/01- 1995/12/01	Following the crash, investor sentiment shifted from optimism to caution, driving steady, non-speculative market growth. The economy showed normal business cycle patterns: healthy growth (4.18% in 1988), recession (-0.11% in 1991), then recovery (3.52% in 1992). This reflected typical cyclical behavior, not speculative detachment from economic fundamentals[34].
Post-Dot-Com Recession/Recovery	2001/04/01 - 2003/12/01	The U.S. economy entered recession in March 2001, followed by a weak, jobless recovery lasting through 2003[35]. To combat the downturn, the Federal Reserve aggressively cut interest rates from 6.5% in late 2000 to 1% by June 2003—a 45-year low[36].

Table 4 Sources of Respective Macro-Financial Data

Macro-Financial Data	Sources
Consumer Price Index (CPI)	U.S. Bureau of Labor Statistics. Consumer Price Index for All Urban Consumers: All items in U.S. city average (CPIAUCSL). Federal Reserve Bank of St. Louis (FRED): https://fred.stlouisfed.org/series/CPIAUCSL
Producer Price Index (PPI)	U.S. Bureau of Labor Statistics. Producer Price Index by Commodity: All Commodities (PPIACO). Federal Reserve Bank of St. Louis (FRED): https://fred.stlouisfed.org/series/PPIACO
Federal Funds Rate (FEDFUNDS)	Federal Funds Effective Rate (FEDFUNDS). Federal Reserve Bank of St. Louis (FRED): https://fred.stlouisfed.org/series/FEDFUNDS
10-Year Treasury Yield (DGS10)	Market Yield on U.S. Treasury Securities at 10-Year Constant Maturity, Quoted on an Investment Basis (DGS10). Federal Reserve Bank of St. Louis (FRED): https://fred.stlouisfed.org/series/DGS10
S&P500 P/E Ratio	GuruFocus. S&P 500 P/E Ratio: https://www.gurufocus.com/economic_indicators/57/sp-500-pe-ratio
Dow Jones Industrial Average (DJIA)	S&P Dow Jones Indices. Dow Jones Industrial Average Monthly Performance Report (Excel download): https://www.spglobal.com/spdji/en/web-data-downloads/reports/dja-performance-report-monthly.xls

dalone DL model, standalone LLM, and the DL-LLM system) was completed through three complementary approaches: bubble only, non-bubble only, and both types. Under three different configurations, the efficacy of each architecture in predicting bubbles given unknown bubble or non-bubble data was evaluated.

2.4.2 Approach 1: Bubble Only Configuration with Evaluation on Both Bubble and Non-Bubble Data

- Configuration and Evaluation Overview

Table 5 Macro-Financial Dataset Structure. This figure illustrates the organization of individual macro-financial datasets for both bubble and non-bubble periods within a single CSV file. Each dataset begins with an initial date starting from the second row. Dates increment monthly (e.g., 1971-05-01, 1971-06-01), continuing until the final date. All economic data was in maximum rounded to four decimal places.

Date	DJIA	SP500_PE	FEDFUNDS	DGS10	CPI	PPI
YYYY-MM-DD	val	val	val	val	val	val
YYYY-MM-DD	val	val	val	val	val	val
YYYY-MM-DD	val	val	val	val	val	val
. . .	. . .	. . .	. . .	. . .	. . .	. . .

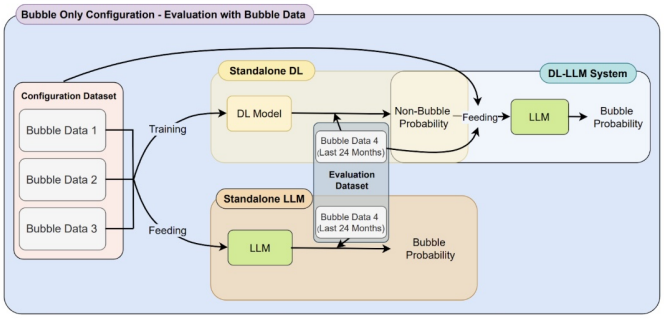

Fig. 3 Configuration and Evaluation Framework for Standalone DL, Standalone LLM, and DL-LLM Architecture Utilizing Bubble Data to Predict Bubble Proability Based on Unseen Bubble Data.

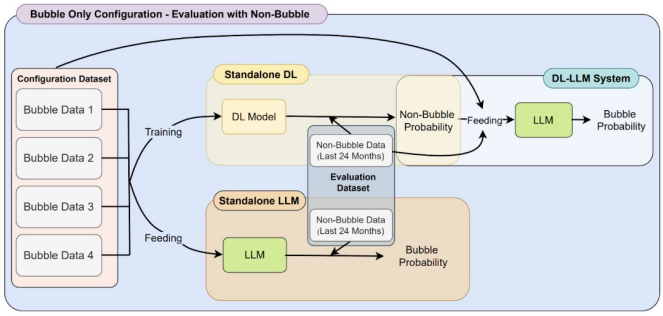

Fig. 4 Configuration and Evaluation Framework for Standalone DL, Standalone LLM, and DL-LLM Architecture Utilizing Bubble Data to Predict Bubble Proability Based on Unseen Non-Bubble Data.

In the first approach, all architectures were configured with only the macro-financial data of the bubble periods while the evaluation let the architectures predict based on unseen bubble and non-bubble data.

To evaluate the architectures based on bubble data, as shown in Fig 3, among the four bubble data sets obtained, three of them were chosen for the three architectural configurations, where the remaining one was utilized for evaluation purposes. As it is detailed in section 2.5, since the DL model was trained on 24-month windows of economic data, each dataset was truncated to its final 24 months.

Continuing, for the evaluation of architectures using non-bubble data, all four bubble datasets were incorporated into the configuration phase, as there was no risk of data contamination when testing exclusively on non-bubble periods. Subsequently, each of the four non-bubble datasets was individually input into the models for bubble probability prediction assessment. The non-bubble data for evaluation was also constrained to the final 24 months.

As detailed in Table 6, the following sections provide a detailed overview of the data compositions and specific evaluation methodology for each architectures.

- Data Preparation for Configuring and Evaluating the Three AI Architectures
  The following table represents different configurations of macro-financial data for evaluating three AI architectures' performance. The first four cells represent the evaluation using the bubble data, while the last four cells represent the evaluation using the non-bubble data.

- Detailed Configuration and Evaluation Methodology for Three AI Architectures
  Different file combinations presented in Table 6 were input individually for each architecture. The following table details the specific configuration and evaluation methodology for individual AI architecture as they have inherent difference: The DL model uses traditional train-test splits, the LLM leverages prompt-based analysis, and the DL-LLM system combines DL outputs with prompted LLM interpretation.

2.4.3 Approach 2: Non-Bubble Only Approach

- Configuration and Evaluation Overview

  In the second approach, all architectures were configured with only the macro-financial data of the non-bubble periods while the evaluation let the architectures predict the non-bubble probability based on unseen bubble and non-bubble data. As the configuration datasets were entirely non-bubble, it naturally drove the models to predict non-bubble probability. Thus, for results analysis, the bubble probability was calculated by subtracting the non-bubble probability values from 1.0.

  To evaluate the architectures based on bubble data, as shown in Fig 5, all four non-bubble data was utilized for the configuration of the three architectures and each four bubble's final 24 months data was used for the evaluation. This

Table 6 Different Data Combinations for Approach 1.

Setup Tag	Macro-Financial Data Processed by Architectures	Macro-Financial Data Used for Evaluation
1-a	Black Monday, Dot Com, Nifty Fifty Bubble	Subprime Housing Bubble Last 24 months
1-b	Black Monday, Dot Com, Subprime Housing Bubble	Nifty Fifty Bubble Last 24 months
1-c	Black Monday, Nifty Fifty, Subprime Housing Bubble	Dot Com Bubble Last 24 months
1-d	Dot Com, Nifty Fifty, Subprime	Black Monday Bubble Last 24 months
1-e	Nifty Fifty, Black Monday, Dot Com, Subprime Housing Bubble	Early 1960s Economic Expansion
1-f	Nifty Fifty, Black Monday, Dot Com, Subprime Housing Bubble	The Late 1970s Expansion
1-g	Nifty Fifty, Black Monday, Dot Com, Subprime Housing Bubble	Post-Black Monday Recovery
1-h	Nifty Fifty, Black Monday, Dot Com, Subprime Housing Bubble	Post-Dot-Com Recession/Recovery

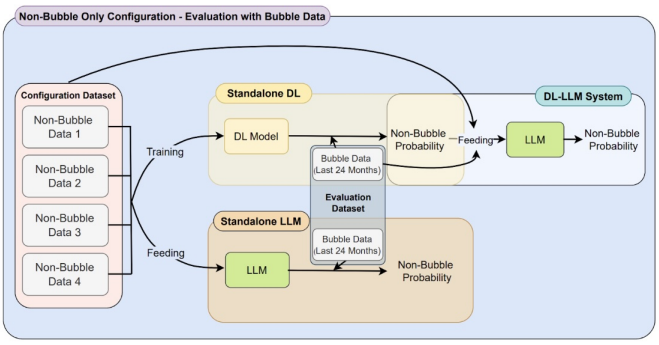

Fig. 5 Configuration and Evaluation Framework for Standalone DL, Standalone LLM, and DL-LLM Architecture Utilizing Non-Bubble Data to Predict Non-Bubble Proability Based on Unseen Bubble Data.

tested the architectures' non-bubble probability predictions on previously unseen bubble data.

Continuing, for the evaluation of architectures using non-bubble data, three of the four non-bubble data was used for the configuration while the remaining data's final 24 months was used for the evaluation, where architectures predicted the non-bubble probability.

Table 7 Configuration and Evalution Details for Individual Architectures Under the First Approach.

AI Architecture	Methodology
Standalone DL Model	The DL model was trained following table 6's different data combinations. For evaluation using the bubble data, training was done with three of the four bubbles' macro-financial data while the remaining bubble's last 24 months of data was utilized for bubble probability prediction. For evaluation using the non-bubble data, training was done with all the four bubble data while each four non-bubble's last 24 months of data was used for bubble probability prediction. The training and evaluation of the DL models were done entirely through Google Colab, where detailed training methodology including the codes are illustrated in section 2.5. Considering that the DL models produce identical outputs for identical inputs once training is complete, the evalutions were not repeated.
Standalone LLM Model	As LLM is a pretrained model, The LLM configuration was implemented through ChainForge using API calls to various chosen LLM models. Evaluations based on specific file compositions were each repeated five times. Model selection and prompt engineering details are illustrated with detail in section 2.6. For evaluation using the bubble data, the LLM was fed with three bubble data and tested on the remaining bubble's last 24 months using a specialized prompt for bubble probability prediction. For evaluation using the non-bubble data, the LLM was fed with four bubble data and tested on the four different non-bubble's final 24 months of data along with a specialized prompt for bubble probability prediction. During testing, each bubble and non-bubble dataset was provided in JSON format containing natural language summaries of the specific bubble or non-bubble data, as detailed in Section 2.6. Standardized labels such as Bubble_Prototype_1, Non-Bubble_Prototype, and Target_Data were assigned to prevent filename-based bias.
DL-LLM System	Similar to Standlaone LLM, DL-LLM configurations were implemented through ChainForge using API calls to various selected LLM models. Each evaluation based on specific file compositions was repeated five times to ensure statistical reliability. Detailed model selection criteria and prompt engineering approaches are presented in Section 2.6. The LLM was provided with the standalone DL model's output, along with a sophisticated prompt for bubble probability prediction. Using the approach employed for the Standalone LLM, during testing, each bubble and non-bubble dataset was provided in JSON format containing natural language summaries of the specific bubble or non-bubble data, as detailed in Section 2.6. Standardized labels such as Bubble_Prototype_1, Non-Bubble_Prototype, and Target_Data were assigned to prevent filename-based bias.

As detailed in Table 6, the following sections provide a detailed overview of the data compositions and specific

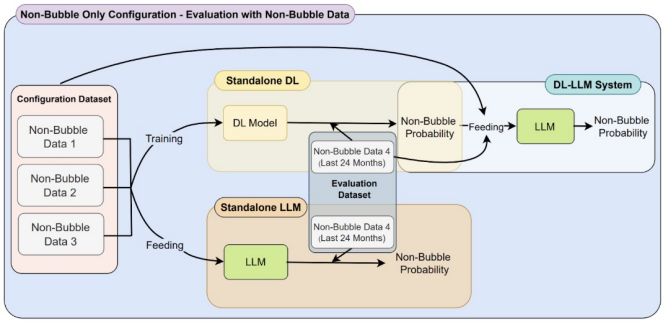

Fig. 6 Configuration and Evaluation Framework for Standalone DL, Standalone LLM, and DL-LLM Architecture Utilizing Non-Bubble Data to Predict Non-Bubble Proability Based on Unseen Non-Bubble Data.

evaluation methodology for each architectures.

- Data Preparation for Configuring and Evaluating the Three AI Architectures

  The following table represents different configurations of macro-financial data for evaluating three AI architectures' performance. The first four cells represent the evaluation using the bubble data, while the last four cells represent the evaluation using the non-bubble data.

- Detailed Configuration and Evaluation Methodology for Three AI Architectures

  Different file combinations presented in Table 8 were input individually for each architecture. The following table details the specific configuration and evaluation methodology for each AI architecture, highlighting their inherent differences.

2.4.4 Approach 3: Bubble and Non-Bubble Approach

- Configuration and Evaluation Overview

  Inspired by the first two approaches, the final approach treats the non-bubble periods as noise for bubble detection, combining bubble and non-bubble data to configure the three architectures. To evaluate the architectures based on bubble data, three of the four bubble data and the all four non-bubble data was used for configuration of the architectures, where the one remaining bubble data was used for the evaluation. This tested the architectures' bubble probability predictions on previously unseen bubble data. To evaluate the architectures based on non-bubble data, three of the four non-bubble data and the all four bubble data was used for configuration of the architectures, where the one remaining non-bubble data was used for the evaluation. This tested the architectures' bubble probability predictions on previously unseen non-bubble data. As detailed in Table 6, the following sections provide a detailed overview of the data compositions and specific evaluation methodology for each architectures.

Table 8 Different Data Combinations for Non-Bubble Testing.

Setup Tag	Macro-Financial Data Processed by Architectures	Macro-Financial Data Used for Evaluation
2-a	Early 1960s Economic Expansion The Late 1970s Expansion Post-Black Monday Recovery Post-Dot-Com Recession/Recovery	Subprime Housing Bubble Last 24 months
2-b	Early 1960s Economic Expansion The Late 1970s Expansion Post-Black Monday Recovery Post-Dot-Com Recession/Recovery	Nifty Fifty Bubble Last 24 months
2-c	Early 1960s Economic Expansion The Late 1970s Expansion, Post-Black Monday Recovery Post-Dot-Com Recession/Recovery	Dot Com Bubble Last 24 months
2-d	Early 1960s Economic Expansion The Late 1970s Expansion, Post-Black Monday Recovery Post-Dot-Com Recession/Recovery	Black Monday Bubble Last 24 months
2-e	The Late 1970s Expansion Post-Black Monday Recovery Post-Dot-Com Recession/Recovery	Early 1960s Economic Expansion Last 24 months
2-f	Early 1960s Economic Expansion Post-Black Monday Recovery Post-Dot-Com Recession/Recovery	The Late 1970s Expansion Last 24 months
2-g	Early 1960s Economic Expansion The Late 1970s Expansion, Post-Dot-Com Recession/Recovery	Post-Black Monday Recovery Last 24 months
2-h	Early 1960s Economic Expansion The Late 1970s Expansion, Post-Black Monday Recovery	Post-Dot-Com Recession/Recovery Last 24 months

- Data Preparation for Configuring and Evaluating the Three AI Architectures
  The following table represents different configurations of macro-financial data for evaluating three AI architectures' performance. The first four cells represent the evaluation using the bubble data, while the last four cells represent the evaluation using the non-bubble data.

- Detailed Configuration and Evaluation Methodology for Three AI Architectures

Table 9 Configuration and Evalution Details for Individual Architectures Under the Second Approach.

AI Architecture	Methodology
Standalone DL Model	The DL model was trained following table 8's eight different data combinations. For evaluation using the bubble data, training was done with all the four non-bubble data while each four bubble's last 24 months of data was used for bubble probability prediction. For evaluation using the non-bubble data, training was done with three of the four non-bubble data while the remaining non-bubble's last 24 months of data was utilized for bubble probability prediction. As DL model was trained on non-bubble datasets, its output was also the probability of the non-bubble period rather than the one of the bubble. Thus, for data analysis, the non-bubble probability converted into bubble by subtracting the output from 1.0. The training and evaluation of the DL models were done entirely through Google Colab, where detailed training methodology including the codes are illustrated in section 2.5. Considering that the DL models produce identical outputs for identical inputs once training is complete, the evalutions were not repeated.
Standalone LLM Model	LLM configurations were implemented through ChainForge using API calls to various selected LLM models. Each evaluation based on specific file compositions was repeated five times to ensure statistical reliability. Detailed model selection criteria and prompt engineering approaches are presented in Section 2.6. For evaluation using the bubble data, the LLM was fed with all the four non-bubble data while each four bubble's last 24 months of data was used for bubble non-probability prediction. For evaluation using the non-bubble data, the LLM was fed with three of the four non-bubble data and tested on remaining one's last 24 months of data along with a specialized prompt for non-bubble probability prediction. During testing, each bubble and non-bubble dataset was provided in JSON format containing natural language summaries of the specific bubble or non-bubble data, as detailed in Section 2.6. Standardized labels such as Bubble_Prototype_1, Non-Bubble_Prototype, and Target_Data were assigned to prevent filename-based bias.
DL-LLM Model	Similar to Standalone LLM, DL-LLM configurations were implemented through ChainForge using API calls to various selected LLM models. Each evaluation based on specific file compositions was repeated five times to ensure statistical reliability. Detailed model selection criteria and prompt engineering approaches are presented in Section 2.6. The LLM was provided with the standalone DL model's output, along with a sophisticated prompt for 'non-bubble' probability prediction. The bubble probability was then calculated as 1.0 minus the non-bubble output. Using the approach employed for the Standalone LLM, during testing, each bubble and non-bubble dataset was provided in JSON format containing natural language summaries of the specific bubble or non-bubble data, as detailed in Section 2.6. Standardized labels such as Bubble_Prototype_1, Non-Bubble_Prototype, and Target_Data were assigned to prevent filename-based bias.

Different file combinations presented in Table 10 were input individually for each architecture. The following table details the specific configuration and evaluation methodology for each AI architecture, highlighting their inherent differences.

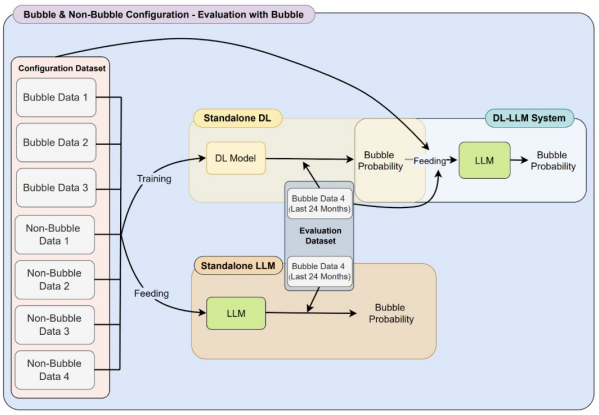

Fig. 7 Configuration and Evaluation Framework for Standalone DL, Standalone LLM, and DL-LLM Architecture Utilizing Bubble and Non-Bubble Data to Predict Bubble Proability Based on Unseen Bubble Data.

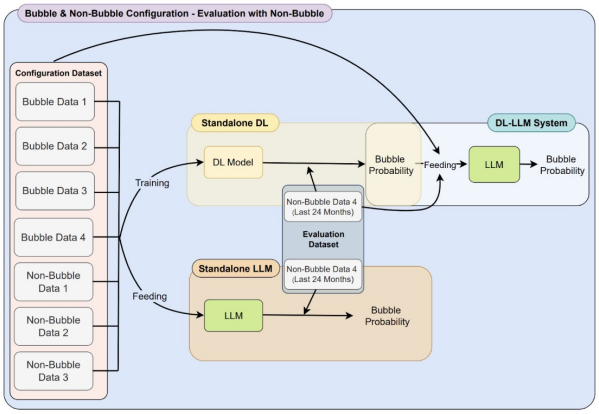

Fig. 8 Configuration and Evaluation Framework for Standalone DL, Standalone LLM, and DL-LLM Architecture Utilizing Bubble and Non-Bubble Data to Predict Bubble Proability Based on Unseen Non-Bubble Data.

2.5 DL Model Development for Standalone DL and DL-LLM Architectures

2.5.1 Overview

As detailed in section 2.4, three versions of the DL models were developed using different approaches: bubble-only, non-bubble-only, and combined datasets. Within each approach, multiple model iterations emerged from varying training file compositions.

All models employed the two different Deep Learning Architecture: BiLSTM and Transformer. The BiLSTM integrated self-supervised representation learning with supervised classification, reflecting the financial industry's current rapidly credited predictive methodology. The Transformer architecture was em-

Table 10 Different Data Combinations for Bubble and Non-Bubble Testing. Each combination represents a different configuration of historical bubble and non-bubble period data used to evaluate AI architectures' performance. The first four cells represent the evaluation using the bubble data, while the last four cells represent the evaluation using the non-bubble data.

Setup Tag	Macro-Financial Data Processed by Architectures		Macro-Financial Data Used for Evaluation
3-a	**Bubble**	Black Monday, Dot Com, Nifty Fifty Bubble	Subprime Housing Bubble Last 24 months
	Non-Bubble	Early 1960s Economic Expansion, The Late 1970s Expansion, Post-Black Monday Recovery, Post-Dot-Com Recession/Recovery	
3-b	**Bubble**	Black Monday, Dot Com, Subprime Housing Bubble	Nifty Fifty Bubble Last 24 months
	Non-Bubble	Early 1960s Economic Expansion, The Late 1970s Expansion, Post-Black Monday Recovery, Post-Dot-Com Recession/Recovery	
3-c	**Bubble**	Black Monday, Nifty Fifty, Subprime Housing Bubble	Dot Com Bubble Last 24 months
	Non-Bubble	Early 1960s Economic Expansion, The Late 1970s Expansion, Post-Black Monday Recovery, Post-Dot-Com Recession/Recovery	
3-d	**Bubble**	Dot Com, Nifty Fifty, Subprime	Black Monday Bubble Last 24 months
	Non-Bubble	Early 1960s Economic Expansion, The Late 1970s Expansion, Post-Black Monday Recovery, Post-Dot-Com Recession/Recovery	
3-e	**Bubble**	Nifty Fifty, Black Monday, Dot Com, Subprime Housing Bubble	Early 1960s Economic Expansion Last 24 months
	Non-Bubble	The Late 1970s Expansion, Post-Black Monday Recovery, Post-Dot-Com Recession/Recovery	
3-f	**Bubble**	Nifty Fifty, Black Monday, Dot Com, Subprime Housing Bubble	The Late 1970s Expansion Last 24 months
	Non-Bubble	Early 1960s Economic Expansion, Post-Black Monday Recovery, Post-Dot-Com Recession/Recovery	
3-g	**Bubble**	Nifty Fifty, Black Monday, Dot Com, Subprime Housing Bubble	Post-Black Monday Recovery Last 24 months
	Non-Bubble	Early 1960s Economic Expansion, The Late 1970s Expansion, Post-Dot-Com Recession/Recovery	
3-h	**Bubble**	Nifty Fifty, Black Monday, Dot Com, Subprime Housing Bubble	Post-Dot-Com Recession / Recovery Last 24 months
	Non-Bubble	Early 1960s Economic Expansion, The Late 1970s Expansion, Post-Black Monday Recovery,	

Table 11 Configuration and Evalution Details for Individual Architectures Under the Third Approach.

AI Architecture	Methodology
Standalone DL Model	The DL model was trained following table 10's eight different data combinations. For evaluation using the bubble data, training was done with all the four non-bubble data and three of the four bubble data. Then, the one reamining bubble's last 24 months of data was used for bubble probability prediction. For evaluation using the non-bubble data, training was done with three of the four non-bubble data and all four bubble data. Then, the one reamining non-bubble's last 24 months of data was used for bubble probability prediction. The training and evaluation of the DL models were done entirely through Google Colab, where detailed training methodology including the codes are illustrated in section 2.5. Considering that the DL models produce identical outputs for identical inputs once training is complete, the evalutions were not repeated.
Standalone LLM Model	LLM configurations were implemented through ChainForge using API calls to various selected LLM models. Each evaluation based on specific file compositions was repeated five times to ensure statistical reliability. Detailed model selection criteria and prompt engineering approaches are presented in Section 2.6. For evaluation using the bubble data, the LLM was fed with all the four non-bubble data and three of the four bubble data. Then, the one reamining bubble's last 24 months of data was used for bubble probability prediction. For evaluation using the non-bubble data, the LLM was fed with three of the four non-bubble data and all four bubble data. Then, the one reamining non-bubble's last 24 months of data was used for bubble probability prediction. During testing, each bubble and non-bubble dataset was provided in JSON format containing natural language summaries of the specific bubble or non-bubble data, as detailed in Section 2.6. Standardized labels such as Bubble_Prototype_1, Non-Bubble_Prototype, and Target_Data were assigned to prevent filename-based bias.
DL-LLM System	Similar to Standalone LLM, DL-LLM configurations were implemented through ChainForge using API calls to various selected LLM models. Each evaluation based on specific file compositions was repeated five times to ensure statistical reliability. Detailed model selection criteria and prompt engineering approaches are presented in Section 2.6. The LLM was provided with the standalone DL model's output, along with a sophisticated prompt for 'bubble' probability prediction. Using the approach employed for the Standalone LLM, during testing, each bubble and non-bubble dataset was provided in JSON format containing natural language summaries of the specific bubble or non-bubble data, as detailed in Section 2.6. Standardized labels such as Bubble_Prototype_1, Non-Bubble_Prototype, and Target_Data were assigned to prevent filename-based bias.

ployed as an ablation study to conduct a deeper analysis of DL model behavior under different design configurations.

As noted at the 2.1 Overview section, all materials, including the training codes for the DL models, are available at the following GitHub Repository: `https://github.com/Junbum-Cho/US-Financial-Bubble-Prediction-DL-LLM-and-DL-LLM-Code-Prompts-Raw-Data.git`

2.5.2 Training DL Model Based on the BiLSTM Architecture

- **Contrastive Learning**
  The model universally consists of a two-part system. First, an Encoder, built with a bidirectional LSTM, processes a 24-month window of economic data. Its function is to compress the time series into a fixed-size numerical representation, or embedding, that captures the sequence's essential patterns. This embedding is then fed into a Classifier, a simple feed-forward network with a Sigmoid output, which produces the final probability score between 0 and 1. The training data is prepared into (anchor, alternative) pairs.

An anchor is an original 24-month data window, while an alternative is a semantically identical but slightly distorted version of the anchor, created using data augmentation techniques like TimeWarp, Drift, and AddNoise. This pairing is fundamental to the contrastive learning aspect of the training.

- **Dual-Loss Optimization**
  The model is trained by optimizing a combined loss function that includes two components, forcing it to learn two skills simultaneously. The total loss is a weighted sum of a contrastive loss and a classification loss.

- **Contrastive Loss (NT-Xent):**
  This self-supervised component trains the Encoder. It works by encouraging the embeddings of an anchor and its alternative to be similar while pushing them apart from the embeddings of other windows in the batch. This teaches the model to create robust representations that are invariant to minor noise and distortions.

- **Classification Loss (Binary Cross-Entropy):**
  This supervised component trains the Classifier. It measures the difference between the model's output probability and a ground-truth label. In the single-class models, this label is statically 1 to measure similarity to the training set. For the DL models trained exclusively on bubble-only and non-bubble-only macro-financial data respectively, all training data was labeled 1.0 within each model's specific dataset. This allowed the model to output ranging from 0.0 to 1.0 based on the input that represented the bubble probability or non-bubble probability. In the dual-class model, which integrated both non-bubble and bubble macro-financial data, labels were set to 0.0 for non-bubble data and 1.0 for bubble data to perform direct binary classification.

2.5.3 Training the DL Model Based on the Transformer Architecture

- **Contrastive Learning**
  The model consists of a two-part system. First, an Encoder built with a Transformer processes a fixed-length economic time-series window. Inputs are linearly projected to the embedding dimension and enriched with sinusoidal positional encodings; stacked self-attention layers then capture long-range temporal dependencies and cross-feature interactions. A pooling operation (e.g., last/mean/CLS) converts the sequence into a single, L2-normalized embedding that summarizes the window. This embedding is then passed to a Classifier—a small feed-forward network with a Sigmoid output—that produces a probability between 0 and 1.

  Training samples are organized into (anchor, alternative) pairs. The anchor is an original window, while the alternative is a stochastically augmented view of the same window using time-series perturbations (e.g., time warping, drift, and additive noise). This pairing is central to the contrastive objective, encouraging invariance to benign distortions while preserving the underlying economic signal.

- **Dual-Loss Optimization**
  The model is optimized with a combined objective that teaches two complementary skills at once. The total loss is a weighted sum of a contrastive loss and a supervised classification loss, ensuring the Encoder learns robust sequence representations that are also discriminative for the downstream task.

- **Contrastive Loss (NT-Xent):**
  This self-supervised term compares the anchor and alternative embeddings within a batch. After normalization, pairwise similarities are scaled by a temperature and fed to a cross-entropy objective that pulls each anchor toward its own alternative while pushing it away from other windows. This shapes the Transformer's embedding space to be stable under realistic time-series augmentations.

- **Classification Loss (Binary Cross-Entropy):**
  This supervised term trains the Classifier head on the Encoder's embeddings to predict a binary label (0 for non-bubble, 1 for bubble). Both the anchor and its augmented view contribute to this loss, reinforcing consistency of the predicted probability across views and tying the representation to the task boundary.

Through repeated training cycles each employing BiLSTM and Transfomer, the model learns to balance two synergistic goals: creating signatures that capture general economic patterns while also making them highly effective for the specific classification task.

Training code snippets and direct links to all deep learning models are provided in the appendices. Appendices 1-6 contain code snippets for two models, both trained on bubble data but evaluated on different datasets: one on bubble data and one on non-bubble data. Each model's training process is divided into three sections for convenience, resulting in six appendices total. Similarly, Appendices 7-12 provide training code for models trained solely on non-bubble data and evaluated on both bubble and non-bubble data, while Appendices 13-18 contain code for models trained on both bubble and non-bubble data. The codes used to run the DL model for evaluation are provided from Appendices 19 - 24. Appendices 19 and 20 contain evaluation code for the DL model trained on the bubble only data using bubble data and non-bubble data, respectively. Appendices 21 and 22 provide the same codes for the DL models trained on the non-bubble only data. Appendices 23 and 24 provide the ones for the DL models trained on both datasets. For the detailed

configuration and evaluation methods of the DL models, refer to Section 2.4.

2.6 LLM Selection, Prompt Development, and Runtime Environment for Standalone LLM and DL-LLM Architectures

2.6.1 Selecting an Optimal Inference LLMs for Standalone and DL-LLM architectures

Given LLMs' demonstrated capabilities in Chain of Thought (CoT) reasoning and complex analytical tasks required for this research, three inference LLMs were selected for both the standalone LLM and DL-LLM hybrid systems. All LLM operations were executed through the ChainForge platform, which enabled direct API calls to systematically generate and collect outputs across multiple evaluation scenarios.

- Gemini 2.5 Pro: The Google Generative AI API released on June 17, 2025 was configured with a 10,000-token output limit, temperature set to 0.5 for balanced creativity-determinism outputs, no system instructions, automatic function calling enabled for direct tool invocation, and no explicit tools array configured.
- ChatGPT-5 Thinking: The OpenAI API released on August 7, 2025 was configured with a 10,000-token output limit for extended reasoning, temperature set to 0.5 for balanced deterministic-variable outputs, no system instructions, and auto tool selection enabled for automatic function calling when tools are available.
- Grok-4: The xAI OpenAI-compatible endpoint was configured with the grok-4-0709 model, 10,000-token output limit, temperature at 0.5 for balanced determinism creativity, no system instructions, and no additional tools array in baseline setup

2.6.2 Developing Standalone LLM Prompts Based on Different Configurations and Evaluation Framework

As detailed in Section 2.4, the standalone LLM was evaluated using three approaches: bubble-only, non-bubble-only, and combined datasets. Each approach included eight testing sessions with different macro-financial data configurations. Consequently, prompting details varied based on the approach and data configuration, as detailed below.

As noted at the 2.1 Overview section, all materials, including the raw Standalone and DL-LLM outputs, are available at the following GitHub Repository: `https://github.com/Junbum-Cho/US-Financial-Bubble-Prediction-DL-LLM-and-DL-LLM-Code-Prompts-Raw-Data.git`

The prompt defines the LLM's role as a macro-financial analyst, specifies the experimental objective and data sources, and explicitly prohibits external assistance such as internet searches. This framework enables Gemini 2.5 Pro to rely solely on its inherent analytical capabilities for bubble prediction.

In order to see the different file configurations set for under different approaches, refer to previous section 2.4's Tables 6, 8, and 10.

2.6.3 Developing DL-LLM Prompts Based on Different Configurations and Evaluation Framework

As detailed in Section 4.1, the DL-LLM was evaluated using three approaches: bubble-only, non-bubble-only, and combined datasets. Each approach included eight testing sessions with different DL model outputs performed with different macro-financial data configurations. Consequently, prompting details varied based on the approach and data configuration, as detailed below.

2.6.4 ChainForge for Standalone LLM and DL-LLM Running Environment

This section details the ChainForge environment configuration for the Standalone LLM and DL-LLM architectures. ChainForge was downloaded directly from the GitHub repository and run locally on a Mac environment[37].

To establish stable API connections, custom functions for API calls to the three selected LLMs were developed and integrated into the system, as demonstrated in Figure 11. Subsequently, JSON files were uploaded to the ChainForge network and directly embedded into the prompts, providing natural language summaries of the relevant data to the LLMs. Figure 12 the complete prompt format used for both Standalone LLM and DL-LLM architectures.

For statistical reliability, API calls were repeated five times, with outputs presented in the Results section.

As noted at the 2.1 Overview section, all materials, including the raw data outputs from ChainForge, are available at the following GitHub Repository:

`https://github.com/Junbum-Cho/US-Financial-Bubble-Prediction-DL-LLM-and-DL-LLM-Code-Prompts-Raw-Data.git`

Fig. 10 Example of Bubble or Non-Bubble Data Input. Natural language formatted bubble or non-bubble data was appended to the prompts shown in Figures 9 and 10.

3 Results

3.1 Overview

The following sections present bubble probability estimations generated by the three architectural frameworks, applying the methodologies detailed previously. All results are standardized for consistent interpretation: values approaching 1.0 indicate high bubble probability, while values near 0.0 signify low bubble probability (i.e., high non-bubble probability). Model accuracy

Table 12 Standalone LLM Prompt. Bold sections indicate prompt components that varied across approaches (bubble-only, non-bubble-only, or combined). Within bold sections: text in "quotation marks" represents actual prompt wording, with alternatives separated by slashes (*/*); unquoted text describes the content inserted.

Act as an impartial, closed-world evaluator: using only the de-identified natural-language summaries of six U.S. macro-financial indicators provided below—without external or historical knowledge—assign a probability that the target 24-month window ["is a bubble" / "NOT a bubble"] by strictly following the rules specified in the continuing sections.
1) Operational constraints (read carefully):
- Closed world. Use only the information provided in this prompt. Do not use external tools, web search, prior knowledge of specific events, or any information not explicitly included here.
- No historical inference. Do not name or infer specific eras, crises, tickers, or dates. Treat the reference as an anonymized, widely recognized **["bubble prototype" / "non-bubble prototype" / "bubble and non-bubble prototypes"]**; treat the target as an anonymized window.
- Perform any internal reasoning privately; output only the requested fields. Solely rely on your reasoning capabilities.
- All data describe the United States equity market and U.S. macro-financial indicators. The target is an anonymized 24-month U.S. window; the reference is an anonymized, widely recognized U.S. **["bubble prototype" / "non-bubble prototype" / "bubble and non-bubble prototypes"]**.
- Do not invent numbers or facts. In the rationale, where you should justify your final decision, base your explanation on the NL summaries of the reference and target files.
- If you regard the given reference and target evidence are weak or mixed, avoid confident extremes.
2) What you will be given (all de-identified):
- A reference **["bubble prototypes" / "non-bubble prototypes" / "bubble and non-bubble prototypes"]** represented as natural-language summaries for each of six indicators.
- A target 24-month window is represented the same way. This target window is based on an unknown period.
- Indicators (six) in those reference and target windows: Consumer Price Index (CPI), Producer Price Index (PPI), Federal Funds Rate (FEDFUNDS), 10-Year Treasury Yield (DGS10), S&P 500 P/E Ratio (SP500_PE), Dow Jones Industrial Average (DJIA).
3) Example Natural-language summary format (per indicator, already computed for you):
- net_change_pct: The total percentage change over the full series (e.g., +18.2).
- slope_ols_pct_per_mo: OLS slope of monthly % change (e.g., +0.75).
- trend_r2: R^2 of the overall trend (e.g., 0.61).
- up_month_share: The fraction of up months (e.g., 0.63).
- vol_std_pct: The standard deviation of monthly % changes (e.g., 4.2).
- vol_late_minus_early_pct: The standard deviation of the late period minus the standard deviation of the early period, in % (e.g., +1.1).
- max_drawup_pct: The largest peak-to-prior-trough rise within the full series, % (e.g., +22.0).
- t_peak: The month index of that max drawup in the full series [1..N] (e.g., 19).
- max_drawdown_pct: The largest trough-from-prior-peak fall within the full series, % (e.g., -9.0).
- t_trough: The month index of that max drawdown in the full series [1..N] (e.g., 7).
- acf1: The lag-1 autocorrelation of monthly % changes (e.g., 0.31).
4) Guidance for the NL summaries (attached to this prompt)
• Format: The Data block contains an anonymized 24-month target window. It also includes the full reference data. Both the target and reference data are encoded as an indicator-to-feature pair. No specific filenames or dates are included. Filenames are designed with a specific convention to assist you in identifying whether a file is a target or a reference.
• Values: Each feature is a formatted string (e.g., "+7.4") or "uncertain" when it cannot be computed without imputing missing values.
• Missing data: If present, missing_data lists month indices in [1..N] where a level is missing. This happens when a particular indicator was not officially published for that month. Do not infer across gaps. Drawup/drawdown is computed only within contiguous valid segments.
5) Your task (free qualitative reasoning):
- Inputs to use: the de-identified NL summaries for the Target 24-month window and the Reference **["bubble prototype" / "non-bubble prototype" / "bubble and non-bubble prototypes"]** (pattern-level, anonymized).
- Goal: Assign a single probability that the Target 24-month window is in a **["bubble" / "non-bubble"]** state, using only the information provided here.
- Method: Read the provided summaries carefully and compare them for the stated goal. Do not rely on any external knowledge, numeric thresholds, or unstated calibration rules.
- Uncertainty: If the evidence is weak, mixed, or marked uncertain, say so explicitly and avoid confident extremes.
6) Output (exactly this format):
- Probability: ¡a single number in [0,1] with four decimals representing **["P(bubble)" / "P(non-bubble)"]**.
- Rationale: Justify your decision based only on the provided natural summaries of the macro-financial indicators of the references.
7) Data (de-identified NL summaries):
The de-identified NL summaries are given as a JSONL format with specific tags attached that will allow you to distinguish the reference and target test files. The bubble data has tag **["Bubble Prototype" / "Non-Bubble Prototpye" / "Bubble and Non-Bubble Prototpye"]**, while your target data for output has tag "Target Data". It can be assessed here:
target in JSON, **["Bubble Prototype in JOSN" / "Non-Bubble Prototpye in JSON" / "Bubble and Non-Bubble Prototpye in JSON"]**

Table 13 The DL-LLM Prompt. This figure shows the prompt structure for the first experimental approach, where bubble-only data was used to train the BiLSTM-based DL-LLM model and evaluate its performance on bubble detection tasks. Due to space constraints, individual prompts for each DL-LLM configuration across different approaches and training architectures are provided in a supplementary document available at the following link:

Act as an impartial, closed-world evaluator: using only (i) the DL model's bubble score computed from a standardized 24-month window and (ii) the de-identified natural-language summaries for that same window plus other reference data, assign a probability that the target 24-month window is a bubble, strictly following the rules below and without any external or historical knowledge.
1) Operational constraints (read carefully):
- Closed world. Use only the information provided in this prompt. Do not use external tools, web search, prior knowledge of specific events, or any information not explicitly included here.
- No historical inference. Do not name or infer specific eras, crises, tickers, or dates. Treat any "reference bubble prototype" as an anonymized, pattern-level abstraction; treat the target strictly as an anonymized evaluation window.
- Perform any internal reasoning privately; output only the requested fields. Solely rely on your reasoning capabilities.
- Do not invent numbers or facts. In the rationale, where you should justify your final decision, base your explanation on the architecture and information about the DL model.
- If you regard the given reference and target evidence are weak or mixed, avoid confident extremes.
2) What you will be given (all de-identified):
A single probability pDL$\varepsilon[0,1]$ produced by a DL model trained on multiple historical bubble episodes. This score pertains only to the target 24-month window.
- Target window (24 months) summaries: De-identified natural-language summaries for the six macro-financial indicators computed over one contiguous 24-month U.S. window.
- Reference bubble prototype summaries: De-identified natural-language summaries of the same six indicators and feature schema, representing a bubble pattern (learned at training time).
- Indicators (six) in those reference and target windows: Consumer Price Index (CPI), Producer Price Index (PPI), Federal Funds Rate (FEDFUNDS), 10-Year Treasury Yield (DGS10), S&P 500 P/E Ratio (SP500_PE), Dow Jones Industrial Average (DJIA).
3) About the DL Model and Its Data
- What the DL model does (generic): A supervised time-series model that ingests a standardized 24-month window of six macro-financial indicators—CPI, PPI, Federal Funds Rate (FEDFUNDS), 10-Year Treasury Yield (DGS10), S&P 500 P/E (SP500_PE), and DJIA—and outputs a scalar bubble probability p$\varepsilon[0,1]$ for that window. The model is trained on multiple historical bubble episodes to learn their common temporal patterns and validated on a held-out bubble episode to assess out-of-sample generalization; for reporting, we use the probability from the last 24 months of the evaluation period.
- Architecture for the DL Model Training: Inputs are standardized 24×6 windows over CPI, PPI, FEDFUNDS, DGS10, SP500_PE, DJIA. A 2-layer bidirectional LSTM with hidden size 128 per direction encodes each window; the final forward/backward states are concatenated (256) and passed through a Linear(256$\to$128), then ℓ 2-normalized to produce the embedding z. A classifier MLP (128$\to$64$\to$32$\to$1, ReLU, dropout 0.2) outputs pbubble$\varepsilon[0,1]$ via Sigmoid. Training draws windows that do not cross file ("prototype") boundaries; for each window it forms an augmented view using TimeWarp + Drift + AddNoise. The objective is NT-Xent contrastive loss ($\tau = 0.05$) between the two views + $0.5 \times$ BCE with label 1 (bubble) on both views. Optimization uses Adam (lr=3e-4) for 300 epochs, batch size ≤ 64. Feature scaling uses StandardScaler (macro vs. market columns fit on train data). At inference, the model reports pbubble for each sliding window (typically the last 24-month window or a summary such as the mean).

4) Example Natural-language summary format (per indicator, already computed for you):
- net_change_pct: The total percentage change over the full series (e.g., +18.2).
- slope_ols_pct_per_mo: OLS slope of monthly % change (e.g., +0.75).
- trend_r2: R^2 of the overall trend (e.g., 0.61).
- up_month_share: The fraction of up months (e.g., 0.63).
- vol_std_pct: The standard deviation of monthly % changes (e.g., 4.2).
- vol_late_minus_early_pct: The standard deviation of the late period minus the standard deviation of the early period, in % (e.g., +1.1).
- max_drawup_pct: The largest peak-to-prior-trough rise within the full series, % (e.g., +22.0).
- t_peak: The month index of that max drawup in the full series [1..N] (e.g., 19).
- max_drawdown_pct: The largest trough-from-prior-peak fall within the full series, % (e.g., -9.0).
- t_trough: The month index of that max drawdown in the full series [1..N] (e.g., 7).
- acf1: The lag-1 autocorrelation of monthly % changes (e.g., 0.31).

5) Guidance for the NL summaries (attached to this prompt)
• Format: The Data block contains an anonymized 24-month target window. It also includes the full reference data. Both the target and reference data are encoded as an indicator-to-feature pair. No specific filenames or dates are included. Filenames are designed with a specific convention to assist you in identifying whether a file is a target or a reference.
• Values: Each feature is a formatted string (e.g., "+7.4") or "uncertain" when it cannot be computed without imputing missing values.
• Missing data: If present, missing_data lists month indices in [1..N] where a level is missing. This happens when a particular indicator was not officially published for that month. Do not infer across gaps. Drawup/drawdown is computed only within contiguous valid segments.

6) Your task (free qualitative and rigorous reasoning):
- Inputs to use: the DL model output for the Target 24-month window, plus de-identified NL summaries for the Target (24 months) and the Reference Bubble Prototype (pattern-level, anonymized).
- Goal: Decide whether the Target 24-month window is more consistent with a bubble or a non-bubble state, using only the provided inputs.
- Method (qualitative): Read the Target summaries and the Reference prototype summaries and compare their overall patterns. Do not use any external knowledge, numeric thresholds, or unstated calibration rules.
- Use of DL output: Treat the DL output as one piece of evidence. If the textual comparison supports or contradicts it, state that plainly in your rationale.
- Uncertainty: If the evidence is weak, mixed, or marked uncertain, say so and avoid confident extremes. Do not invent numbers or facts, and do not infer dates, eras, or tickers.

7) Output (exactly this format):
- Probability: a single number in 0,1 with four decimals representing P(bubble) for the Target 24-month window
- Rationale: Justify your decision based only on the provided DL model output and natural summaries of the macro-financial indicators of the references, noting whether that quoted evidence supports or contradicts the DL output.

8) Data (DL Model Output and de-identified NL summaries):
The DL Model outputted:
The de-identified NL summaries are given as a JSONL format with specific tags attached that will allow you to distinguish the reference and target test files. The bubble data has tag "Bubble Prototype", while your target data for output has tag "Target Data". It can be assessed here:
target in JSON
Bubble_Prototypes in JOSN

Table 14 Example of Bubble or Non-Bubble Data Input. Natural language formatted bubble or non-bubble data was appended to the prompts shown in Figures 9 and 10.

Prompt Excerpt
Act as an impartial, closed-world evaluator: using only the de-identified (...) Your target data for output has tag "Target Data". It can be assessed here: `{"label":"Target Data","role":"target","filename":"Target Data 1 (4).json","data":{"window": "24_months","indicators":{"CPI":{"net_change_24m_pct":" +2.9","slope_ols_pct_per_mo":"+0.01", "trend_r2":"0.16","up_month_share":"0.83","vol_std_pct":"+0.1", "vol_late_minus_early_pct":"+0.2" ,"max_drawup_pct":"+3.0","t_peak":24,"max_drawdown_pct":"-0.2","t_trough":20,"acf1":"0.24"}, (...)}}}`

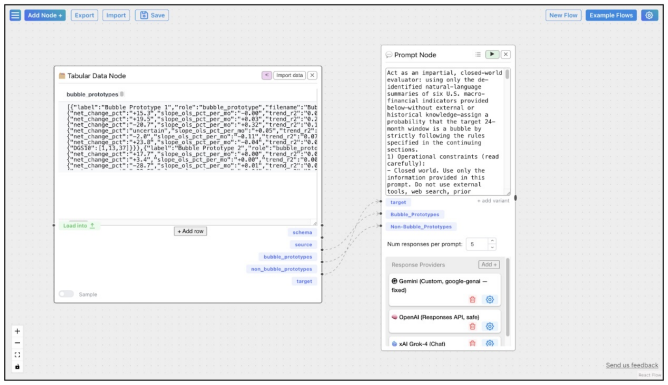

Fig. 9 ChainForge Configuration for Standalone LLM under Approach 3: Bubble and Non-Bubble Configuration Evaluated on Bubble Data. The prompts introduced in Figure 9 were provided along with specific files for analysis. The connecting lines represent the data flow between components.

can be interpreted through the alignment between prediction values and the evaluated dataset: for bubble evaluation datasets, accuracy increases as predictions approach 1.0, while for non-bubble evaluation datasets, accuracy increases as predictions approach 0.0. Additional contextual information necessary for data interpretation is provided in each respective section.

As specified in Section 2.4, the LLM experiments were repeated five times, while the DL model experiments were conducted in single test sessions. Rather than presenting all raw data, this paper provides a concise overview of outputs from the three architectures. For the Standalone LLM and DL-LLM models, mean and standard deviation values are reported due to their repeated testing. For the Standalone DL model, complete outputs are provided since it underwent only single-session testing.

As noted at the 2.1 Overview section, all materials, including the entire three AI architecture outputs, are available at the following GitHub Repository:

`https://github.com/Junbum-Cho/US-Financial-Bubble-Prediction-DL-LLM-and-DL-LLM-Code-Prompts-Raw-Data.git`

Moving forward, in order to more visually represent the data, the brier score was utilized. A brier score measures the accuracy of probabilistic predictions by calculating the mean squared difference between predicted probabilities and actual outcomes, with scores ranging from 0 (perfect) to 1 (worst possible). Section 3.6 illustrates the brier scores calculated for each architectures under unique three approaches.

3.2 Performance Evaluation of the Three Architectures Configured with Bubble Only Data

Tables 15–18 present an overview of outputs from the three architectures: mean and standard deviation values for both the Standalone LLM and LLM-DL models, along with the complete output data for the DL model.

The Standalone DL and DL-LLM variants assign very high bubble probabilities on bubble episodes, but they also produce elevated probabilities on non-bubble episodes (false positives), indicating over-flagging outside the training regime. The Standalone LLM shows mixed calibration on bubble episodes—correctly elevating some (e.g., 1-a to 1-c) but under-detecting Subprime (1-d, $\sim$0.14)—and yields moderate probabilities on several non-bubble episodes (e.g., 1-f, 1-g). Within the DL-LLM hybrids, the LLM component largely defers to the DL backbone's probability, so the backbone remains the dominant determinant of the final estimate.

3.3 Performance Evaluation of the Three Architectures with Non-Bubble Only Data

Tables 19–22 present an overview of outputs from the three architectures: mean and standard deviation values for both the Standalone LLM and LLM-DL models, along with the complete output data for the DL model.

As detailed in section 2.4, architectures configured with exclusively non-bubble data naturally generate non-bubble probability values. Consequently, bubble probability was derived by calculating (1.0 - non-bubble probability) for all presented results. Thus, the interpretation rule remains the same: for evaluation done through bubble datasets, accuracy increases as predictions

Table 15 Bubble Probability Estimation. Standalone DL Models Trained on Historical Bubble Data Based on BiLSTM and Transformer Architecture Evaluated on Unseen Bubble and Non-Bubble Data.

Case Tag	BiLSTM	Transformer	Case Tag	BiLSTM	Transformer
1-a	99.1%	90.26%	1-e	95.0%	99.59%
1-b	99.7%	99.19%	1-f	96.23%	94.53%
1-c	95.52%	95.67%	1-g	95.94%	99.48%
1-d	87.16%	96.27%	1-h	89.62%	99.78%

Table 16 Bubble Probability Estimation. Standalone LLMs Fed with Historical Bubble Data and Evaluated on Unseen Bubble and Non-Bubble Data.

Case Tag	Gemini 2.5 Pro		ChatGPT-5 Thinking		Grok-4	
	Mean ± SD	Range	Mean ± SD	Range	Mean ± SD	Range
1-a	0.61 ± 0.27	[0.15-0.85]	0.60 ± 0.18	[0.28-0.69]	0.66 ± 0.08	[0.60-0.80]
1-b	0.63 ± 0.26	[0.35-0.88]	0.74 ± 0.05	[0.67-0.82]	0.70 ± 0.06	[0.65-0.80]
1-c	0.64 ± 0.11	[0.45-0.70]	0.57 ± 0.10	[0.39-0.65]	0.62 ± 0.04	[0.55-0.65]
1-d	0.14 ± 0.05	[0.05-0.20]	0.21 ± 0.05	[0.18-0.30]	0.33 ± 0.06	[0.25-0.40]
1-e	0.23 ± 0.08	[0.15-0.35]	0.23 ± 0.04	[0.18-0.29]	0.48 ± 0.09	[0.35-0.60]
1-f	0.69 ± 0.15	[0.45-0.85]	0.56 ± 0.12	[0.36-0.68]	0.56 ± 0.11	[0.45-0.70]
1-g	0.60 ± 0.17	[0.35-0.82]	0.39 ± 0.07	[0.32-0.47]	0.63 ± 0.07	[0.60-0.75]
1-h	0.19 ± 0.10	[0.05-0.30]	0.25 ± 0.08	[0.16-0.34]	0.51 ± 0.08	[0.45-0.65]

Table 17 Bubble Probability Estimation. DL-LLM Based on BiLSTM Architecture Configured with Historical Bubble Data and Evaluated on Unseen Bubble and Non-Bubble Data.

Case Tag	Gemini 2.5 Pro		ChatGPT-5 Thinking		Grok-4	
	Mean ± SD	Range	Mean ± SD	Range	Mean ± SD	Range
1-a	0.988 ± 0.003	[0.985-0.991]	0.988 ± 0.004	[0.982-0.991]	0.968 ± 0.021	[0.950-0.991]
1-b	0.996 ± 0.001	[0.995-0.997]	0.986 ± 0.010	[0.970-0.996]	0.982 ± 0.018	[0.950-0.990]
1-c	0.948 ± 0.004	[0.940-0.950]	0.946 ± 0.005	[0.940-0.950]	0.926 ± 0.019	[0.900-0.950]
1-d	0.778 ± 0.093	[0.625-0.876]	0.704 ± 0.083	[0.610-0.800]	0.750 ± 0.061	[0.700-0.850]
1-e	0.925 ± 0.056	[0.825-0.950]	0.886 ± 0.051	[0.800-0.920]	0.900 ± 0.061	[0.800-0.950]
1-f	0.954 ± 0.006	[0.950-0.962]	0.952 ± 0.008	[0.940-0.960]	0.936 ± 0.026	[0.900-0.962]
1-g	0.950 ± 0.000	[0.950-0.950]	0.950 ± 0.000	[0.950-0.950]	0.890 ± 0.063	[0.800-0.950]
1-h	0.864 ± 0.022	[0.850-0.900]	0.878 ± 0.013	[0.860-0.890]	0.804 ± 0.036	[0.750-0.850]

Table 18 Bubble Probability Estimation. DL-LLM Based on Transformer Architecture Configured with Historical Bubble Data and Evaluated on Unseen Bubble and Non-Bubble Data.

Case Tag	Gemini 2.5 Pro		ChatGPT-5 Thinking		Grok-4	
	Mean ± SD	Range	Mean ± SD	Range	Mean ± SD	Range
1-a	0.9026 ± 0.0018	[0.9000-0.9050]	0.9026 ± 0.0018	[0.9000-0.9050]	0.8310 ± 0.0770	[0.7500-0.9026]
1-b	0.9915 ± 0.0009	[0.9900-0.9920]	0.9892 ± 0.0034	[0.9850-0.9919]	0.9919 ± 0.0000	[0.9919-0.9919]
1-c	0.9440 ± 0.0119	[0.9250-0.9550]	0.9550 ± 0.0023	[0.9520-0.9567]	0.9280 ± 0.0464	[0.8500-0.9567]
1-d	0.8675 ± 0.0939	[0.7500-0.9627]	0.9037 ± 0.0713	[0.7800-0.9627]	0.8400 ± 0.0962	[0.7000-0.9500]
1-e	0.9842 ± 0.0196	[0.9500-0.9959]	0.9566 ± 0.0427	[0.8900-0.9950]	0.9675 ± 0.0427	[0.9000-0.9959]
1-f	0.9352 ± 0.0143	[0.9150-0.9453]	0.9417 ± 0.0049	[0.9350-0.9453]	0.9301 ± 0.0139	[0.9200-0.9453]
1-g	0.9950 ± 0.0000	[0.9950-0.9950]	0.9929 ± 0.0018	[0.9910-0.9948]	0.9948 ± 0.0000	[0.9948-0.9948]
1-h	0.8944 ± 0.1657	[0.6000-0.9970]	0.9836 ± 0.0071	[0.9730-0.9910]	0.9496 ± 0.0346	[0.9000-0.9978]

approach 1.0, while for non-bubble datasets, accuracy increases as predictions approach 0.0.

Training solely on non-bubble data produces distinct behaviors across architectures. Standalone DL outputs remain near zero across both regimes, which is desirable on non-bubble episodes but leads to systematic under-detection of bubbles

Table 19 Bubble Probability Estimation. DL Models Trained on Historical Non-Bubble Data and Evaluated on Unseen Bubble and Non-Bubble Data.

Case Tag	BiLSTM	Transformer	Case Tag	BiLSTM	Transformer
2-a	0.0072	0.0383	1-e	0.1086	0.0032
2-b	0.015	0.0052	1-f	0.0141	0.0958
2-c	0.0076	0.0222	1-g	0.0280	0.0388
2-d	0.024	0.0256	1-h	0.0025	0.0090

Table 20 Bubble Probability Estimation. Standalone LLMs Fed with Historical Non-Bubble Data and Evaluated on Unseen Bubble and Non-Bubble Data

Case Tag	Gemini 2.5 Pro		ChatGPT-5 Thinking		Grok-4	
	Mean ± SD	Range	Mean ± SD	Range	Mean ± SD	Range
2-a	0.119 ± 0.051	[0.075-0.200]	0.146 ± 0.029	[0.110-0.180]	0.250 ± 0.071	[0.150-0.350]
2-b	0.595 ± 0.184	[0.275-0.700]	0.558 ± 0.142	[0.380-0.680]	0.610 ± 0.065	[0.500-0.650]
2-c	0.172 ± 0.052	[0.080-0.200]	0.188 ± 0.046	[0.140-0.260]	0.260 ± 0.055	[0.200-0.300]
2-d	0.175 ± 0.075	[0.100-0.300]	0.168 ± 0.023	[0.130-0.190]	0.330 ± 0.045	[0.250-0.350]
2-e	0.355 ± 0.339	[0.075-0.750]	0.180 ± 0.016	[0.160-0.200]	0.430 ± 0.192	[0.200-0.650]
2-f	0.430 ± 0.236	[0.150-0.700]	0.208 ± 0.047	[0.140-0.250]	0.290 ± 0.055	[0.250-0.350]
2-g	0.336 ± 0.266	[0.100-0.650]	0.200 ± 0.040	[0.160-0.260]	0.280 ± 0.057	[0.200-0.350]
2-h	0.710 ± 0.370	[0.050-0.900]	0.304 ± 0.076	[0.220-0.390]	0.390 ± 0.082	[0.300-0.500]

Table 21 Bubble Probability Estimation. DL-LLM Based on BiLSTM Architecture Configured with Historical Non-Bubble Data and Evaluated on Unseen Bubble and Non-Bubble Data.

Case Tag	Gemini 2.5 Pro		ChatGPT-5 Thinking		Grok-4	
	Mean ± SD	Range	Mean ± SD	Range	Mean ± SD	Range
2-a	0.1543 ± 0.3056	[0.0072-0.7000]	0.1480 ± 0.0390	[0.1200-0.2000]	0.4180 ± 0.2787	[0.1500-0.7200]
2-b	0.0150 ± 0.0000	[0.0150-0.0150]	0.0202 ± 0.0029	[0.0180-0.0250]	0.2100 ± 0.0962	[0.0500-0.3000]
2-c	0.0076 ± 0.0000	[0.0076-0.0076]	0.2960 ± 0.1688	[0.1800-0.5800]	0.3420 ± 0.2501	[0.1000-0.7600]
2-d	0.0314 ± 0.0114	[0.0240-0.0500]	0.1060 ± 0.0358	[0.0700-0.1500]	0.0996 ± 0.0775	[0.0240-0.2000]
2-e	0.1132 ± 0.0072	[0.1086-0.1250]	0.3640 ± 0.1596	[0.2200-0.6200]	0.1852 ± 0.1265	[0.1086-0.4000]
2-f	0.0143 ± 0.0004	[0.0141-0.0150]	0.2030 ± 0.1944	[0.0200-0.5200]	0.1700 ± 0.1151	[0.0500-0.3000]
2-g	0.0304 ± 0.0054	[0.0280-0.0400]	0.1716 ± 0.2107	[0.0280-0.5400]	0.2700 ± 0.0758	[0.2000-0.3500]
2-h	0.0050 ± 0.0056	[0.0025-0.0150]	0.0143 ± 0.0112	[0.0005-0.0300]	0.2000 ± 0.0707	[0.1000-0.2500]

Table 22 Bubble Probability Estimation. DL-LLM Based on Transformer Architecture Configured with Historical Non-Bubble Data and Evaluated on Unseen Bubble and Non-Bubble Data.

Case Tag	Gemini 2.5 Pro		ChatGPT-5 Thinking		Grok-4	
	Mean ± SD	Range	Mean ± SD	Range	Mean ± SD	Range
3-a	0.0383 ± 0.0000	[0.0383-0.0383]	0.3157 ± 0.2383	[0.0383-0.5700]	0.2177 ± 0.1782	[0.0383-0.5000]
3-b	0.0242 ± 0.0424	[0.0052-0.1000]	0.0107 ± 0.0062	[0.0052-0.0200]	0.1010 ± 0.0591	[0.0052-0.1500]
3-c	0.0222 ± 0.0000	[0.0222-0.0222]	0.2820 ± 0.1293	[0.1200-0.4600]	0.2100 ± 0.0652	[0.1000-0.2500]
3-d	0.0222 ± 0.0000	[0.0222-0.0222]	0.3940 ± 0.2017	[0.1800-0.6000]	0.2900 ± 0.0822	[0.2000-0.4000]
3-e	0.0032 ± 0.0000	[0.0032-0.0032]	0.1680 ± 0.1460	[0.0600-0.4200]	0.1313 ± 0.1238	[0.0032-0.2500]
3-f	0.0966 ± 0.0019	[0.0958-0.1000]	0.1260 ± 0.0345	[0.0958-0.1800]	0.1732 ± 0.0636	[0.0958-0.2500]
3-g	0.0388 ± 0.0000	[0.0388-0.0388]	0.3000 ± 0.0886	[0.1500-0.3700]	0.2100 ± 0.0418	[0.1500-0.2500]
3-h	0.0102 ± 0.0027	[0.0090-0.0150]	0.0240 ± 0.0167	[0.0090-0.0500]	0.3900 ± 0.2881	[0.2000-0.9000]

when evaluated on bubble episodes. Standalone LLM assigns substantially higher probabilities overall: this elevates bubble episodes relative to DL but also raises many non-bubble episodes into a moderate range, yielding false positives. The DL-LLM hybrids sit between these patterns: their outputs are anchored to the DL backbone yet exhibit an upward shift from the LLM

component. This adjustment improves bubble sensitivity versus DL alone but does not fully resolve overestimation on several non-bubble episodes. In short, single-regime training exposes a trade-off: DL favors non-bubble specificity; LLMs favor bubble sensitivity; the hybrid predominantly reflects the DL signal with modest LLM-driven lift.

False positives on non-bubble episodes likely stem from overlapping macro-financial signatures—late-cycle expansions (rising P/E and equity levels with benign rate dynamics) resemble the bubble prototypes in the summaries, raising $P(\text{bubble})$ even when labels are non-bubble. In the single-regime setting, the LLM acquires an upward prior on bubble likelihood from pattern comparisons, which—together with the hybrid's measured +0.12 upward adjustment relative to DL in Approach 2—elevates several non-bubble cases into the moderate range.

3.4 Performance Evaluation of the Three Architectures with Bubble and Non-Bubble Data

Tables 23–26 present an overview of outputs from the three architectures: mean and standard deviation values for both the Standalone LLM and LLM-DL models, along with the complete output data for the DL model.

When trained on both regimes, Standalone DL (BiLSTM) produces the most balanced probabilities: it assigns high bubble probabilities in three of the four bubble episodes and low probabilities in most non-bubble episodes, with two notable errors—Dot-Com (3-c) is under-estimated, and 1987–1995 (3-g) is over-estimated. The DL-LLM (BiLSTM backbone) is very close to these patterns and tracks the DL backbone, consistent with the hybrid's output being anchored by the DL probability. By contrast, Transformer-backbone variants (Standalone DL and DL-LLM) tend to assign very high bubble probabilities even on non-bubble episodes, indicating poor calibration in this mixed setting. The Standalone LLM (mean of three heads) remains moderate across episodes—less prone to extreme over-flagging than the Transformer-backbone DLs, but not as well aligned with the mixed-regime labels as the BiLSTM DL.

3.5 Brier Scores For the Three AI Architectures Under Three Approaches

Each architecture is evaluated using the Brier score, a proper scoring rule for probabilistic predictions. For a set of N evaluation episodes, the score is

$$\text{Brier} = \frac{1}{N}\sum_{i=1}^{N}(p_i - y_i)^2 \tag{1}$$

where p_i is the model's predicted probability of "bubble" for episode i, and $y_i \in \{0,1\}$ is the ground-truth label (1 for bubble episodes; 0 for non-bubble episodes). In this setting, all model outputs are already $P(\text{bubble})$. For Standalone DL models, p_i is the single deterministic probability produced for that episode. For Standalone LLM models, the five evaluation runs per LLM head (Gemini, GPT-5 Thinking, Grok-4) were averaged to obtain one per-head probability per episode, then the mean across the three heads was taken to yield a single episode-level p_i for the architecture. For DL-LLM models, the same two-stage averaging (five runs per head $\rightarrow$ mean across three heads) was used separately for each backbone (BiLSTM and Transformer). Brier scores were computed within each approach (Bubble-only, Non-bubble-only, Both) by averaging the eight episode errors (four bubble + four non-bubble). These per-approach Brier scores are reported without any additional thresholding or calibration; lower values indicate better probabilistic accuracy and calibration.

Table 27 presents the mean, standard deviation, and the range for the Brier scores obtained for each AI architecture.

Across the three evaluation settings, the Brier scores reveal a clear pattern in probabilistic accuracy. Under Approach 1 (Bubble-only) and Approach 2 (Non-bubble-only), the Standalone LLM (mean of 3) achieves the lowest error (A1: 0.2303; A2: 0.2775), outperforming both DL baselines by $\sim$0.20–0.26 absolute Brier points (e.g., vs. DL-BiLSTM: 0.4463$\rightarrow$0.2303 in A1; 0.4883$\rightarrow$0.2775 in A2). In contrast, when exposed to both regimes (Approach 3), the Standalone DL (BiLSTM) yields the best overall calibration (0.1790), with DL-LLM (BiLSTM backbone) a close second (0.1830, $\Delta \approx 0.004$), while the Standalone LLM trails (0.2489). Notably, Transformer-backbone variants perform substantially worse in A3 (Standalone DL: 0.7030; DL-LLM: 0.6025), consistent with a tendency to over-predict bubbles during non-bubble episodes. Taken together, these results suggest that LLMs excel when trained/evaluated within a single regime, whereas sequence DL with a BiLSTM backbone (alone or as the DL-LLM backbone) offers superior, more stable probability estimates when both regimes are present. All scores were computed directly from raw $P(\text{bubble})$ outputs using the Brier rule without thresholding or post-calibration; lower values indicate better calibration and accuracy.

4 Discussion

4.1 Conclusion

- **Standalone DL Trained on Both the Bubble and Non-Bubble Data Yields Superior but Still Limited Performance for Financial Bubble Prediction:** The evaluation results indicate that the standalone DL architecture, when trained on both the bubble and non-bubble period's macro-financial data, demonstrates superior performance in predicting financial bubbles among the three architectures configured. However, its performance remains limited, where the standalone architecture consistently completely failing to distinguish one bubble (Case 3-c) and one non-bubble

Table 23 Bubble Probability Estimation. DL Models Trained on Historical Bubble and Non-Bubble Data and Evaluated on Unseen Bubble and Non-Bubble Data.

Case Tag	BiLSTM	Transformer	Case Tag	BiLSTM	Transformer
3-a	0.6977	0.7061	1-e	0.0017	0.9899
3-b	0.9236	0.9936	1-f	0.2187	0.9423
3-c	0.1824	0.0601	1-g	0.7509	0.9152
3-d	0.7763	0.0060	1-h	0.00685	0.9800

Table 24 Bubble Probability Estimation. Standalone LLMs Fed with Historical Bubble and Non-Bubble Data and Evaluated on Unseen Bubble and Non-Bubble Data

Case Tag	Gemini 2.5 Pro		ChatGPT-5 Thinking		Grok-4	
	Mean ± SD	Range	Mean ± SD	Range	Mean ± SD	Range
3-a	0.572 ± 0.245	[0.150-0.785]	0.464 ± 0.193	[0.220-0.610]	0.557 ± 0.127	[0.333-0.650]
3-b	0.387 ± 0.306	[0.150-0.785]	0.662 ± 0.071	[0.600-0.780]	0.470 ± 0.205	[0.200-0.650]
3-c	0.425 ± 0.103	[0.350-0.600]	0.280 ± 0.048	[0.230-0.340]	0.630 ± 0.125	[0.450-0.800]
3-d	0.182 ± 0.046	[0.150-0.250]	0.172 ± 0.019	[0.140-0.190]	0.300 ± 0.035	[0.250-0.350]
3-e	0.300 ± 0.071	[0.200-0.350]	0.216 ± 0.025	[0.180-0.240]	0.640 ± 0.042	[0.600-0.700]
3-f	0.375 ± 0.302	[0.150-0.900]	0.372 ± 0.116	[0.270-0.560]	0.610 ± 0.065	[0.500-0.650]
3-g	0.260 ± 0.042	[0.200-0.300]	0.334 ± 0.155	[0.220-0.600]	0.480 ± 0.135	[0.350-0.650]
3-h	0.315 ± 0.246	[0.150-0.750]	0.252 ± 0.052	[0.200-0.320]	0.500 ± 0.137	[0.300-0.650]

Table 25 Bubble Probability Estimation. DL-LLM Based on BiLSTM Architecture Configured with Historical Bubble and Non-Bubble Data and Evaluated on Unseen Bubble and Non-Bubble Data.

Case Tag	Gemini 2.5 Pro		ChatGPT-5 Thinking		Grok-4	
	Mean ± SD	Range	Mean ± SD	Range	Mean ± SD	Range
3-a	0.683 ± 0.032	[0.650-0.720]	0.681 ± 0.047	[0.600-0.720]	0.684 ± 0.019	[0.650-0.698]
3-b	0.907 ± 0.032	[0.850-0.925]	0.888 ± 0.043	[0.820-0.920]	0.862 ± 0.081	[0.750-0.920]
3-c	0.100 ± 0.000	[0.100-0.100]	0.188 ± 0.004	[0.180-0.190]	0.214 ± 0.076	[0.180-0.350]
3-d	0.720 ± 0.064	[0.625-0.770]	0.429 ± 0.030	[0.400-0.460]	0.590 ± 0.055	[0.500-0.650]
3-e	0.0242 ± 0.0428	[0.0017-0.1000]	0.0039 ± 0.0035	[0.0017-0.0100]	0.1442 ± 0.0967	[0.0010-0.2500]
3-f	0.2187 ± 0.0000	[0.2187-0.2187]	0.2205 ± 0.0058	[0.2140-0.2300]	0.2412 ± 0.0355	[0.2187-0.3000]
3-g	0.7565 ± 0.0131	[0.7500-0.7800]	0.6220 ± 0.1031	[0.4600-0.7400]	0.6400 ± 0.0224	[0.6000-0.6500]
3-h	0.0701 ± 0.0028	[0.0685-0.0750]	0.0722 ± 0.0082	[0.0650-0.0850]	0.1114 ± 0.0410	[0.0685-0.1500]

Table 26 Bubble Probability Estimation. DL-LLM Based on Transformer Architecture Configured with Historical Bubble and Non-Bubble Data and Evaluated on Unseen Bubble and Non-Bubble Data.

Case Tag	Gemini 2.5 Pro		ChatGPT-5 Thinking		Grok-4	
	Mean ± SD	Range	Mean ± SD	Range	Mean ± SD	Range
3-a	0.7019 ± 0.0094	[0.6850-0.7061]	0.5504 ± 0.1817	[0.2900-0.7061]	0.6949 ± 0.0251	[0.6500-0.7061]
3-b	0.9936 ± 0.0000	[0.9936-0.9936]	0.9904 ± 0.0051	[0.9820-0.9936]	0.9087 ± 0.0804	[0.8000-0.9936]
3-c	0.6304 ± 0.0268	[0.6010-0.6500]	0.3340 ± 0.3609	[0.0700-0.8000]	0.0981 ± 0.0849	[0.0601-0.2500]
3-d	0.0060 ± 0.0000	[0.0060-0.0060]	0.0266 ± 0.0073	[0.0180-0.0350]	0.1624 ± 0.1682	[0.0060-0.4000]
3-e	0.9878 ± 0.0025	[0.9850-0.9899]	0.9068 ± 0.1018	[0.7300-0.9890]	0.9698 ± 0.0197	[0.9500-0.9899]
3-f	0.9408 ± 0.0033	[0.9350-0.9423]	0.9400 ± 0.0033	[0.9350-0.9423]	0.9105 ± 0.0183	[0.9000-0.9423]
3-g	0.9161 ± 0.0022	[0.9150-0.9200]	0.7420 ± 0.0716	[0.6500-0.8400]	0.8391 ± 0.1366	[0.6000-0.9152]
3-h	0.9780 ± 0.0027	[0.9750-0.9800]	0.9470 ± 0.0205	[0.9200-0.9700]	0.9740 ± 0.0134	[0.9500-0.9800]

(Case 3-g) data.

- **Configuring Architectures Using Bubble Only or Non-Bubble Only Data are the Least Effective Approach for Financial Bubble Prediction:** The evaluation results demonstrate that DL models trained exclusively on single-

Table 27 Brier scores by architecture and approach (lower = better).

Architecture	Approach 1 (Bubble-only)	Approach 2 (Non-bubble-only)	Approach 3 (Both)
Standalone DL (BiLSTM)	0.446341	0.488262	0.179014
Standalone DL (Transformer)	0.485432	0.478852	0.702961
Standalone LLM (mean of 3)	0.230274	0.277544	0.248869
DL-LLM (BiLSTM backbone, mean of 3)	0.421102	0.372311	0.183022
DL-LLM (Transformer backbone, mean of 3)	0.436932	0.364864	0.602470

class data (either bubble or non-bubble) consistently fail to recognize market conditions diverging from their training distribution. A similar pattern emerged with the Large Language Model when exclusively provided bubble data, as it subsequently misclassified non-bubble scenarios. While this misclassification tendency diminished when the LLM was configured with non-bubble data, its performance remained suboptimal, accurately identifying only five of eight market conditions with estimation output exceeding 80 percent.

- **DL Model Output Tends To Dominate LLM Reasoning and Output in the DL-LLM System:** DL signal dominates the DL-LLM output. Across approaches, the DL model's probability strongly anchors the DL-LLM estimate. Pooling all 24 episode pairs per backbone (8 episodes $\times$ 3 approaches), DL vs. DL-LLM probabilities are highly correlated (BiLSTM backbone: $r = 0.984$; Transformer backbone: $r = 0.989$) with small deviations (mean absolute difference, MAD $\approx$ 0.070–0.077; median absolute difference $\approx$ 0.049–0.051), and 50% of all pairs lie within ± 0.05. This dominance is most evident in Approach 1 and Approach 3: for DL-BiLSTM vs. DL-LLM(BiLSTM) we find $r = 0.972$, MAD=0.037 with 7/8 episodes within ± 0.05 (A1), and $r = 0.985$, MAD=0.051 with 5/8 within ± 0.05 (A3); for the Transformer backbone $r = 0.974$, MAD=0.073 with 4/8 within ± 0.05 (A3). The only notable departure occurs in Approach 2, where DL emits near-zero probabilities; DL-LLM applies a systematic upward shift (mean $\Delta \approx +0.12$ for both backbones), which lowers r (to 0.36 and 0.24) but still reflects an adjustment around the DL baseline rather than independent behavior. Overall, these statistics indicate the LLM largely transmits the DL signal with modest, regime-dependent adjustments, consistent with DL dominance in the DL-LLM system.

The practical implications above are illustrative and grounded in the present dataset and labeling scheme, focusing exclusively on post 1960s American stock market conditions. They should not be taken as implementation guidance; before use in investment or oversight settings, models require prospective, out-of-time evaluation, larger and more diverse features/labels, robustness checks, and governance controls.

4.2 Insights and Implications

- **Six Macro-Financial Data is Insufficient for Detecting Financial Bubbles Using AIs:** Across all approaches, none of the architectures achieved uniformly low Brier scores or perfect separation of bubble vs. non-bubble episodes (e.g., best case 0.179 in Approach 3, while others exceed 0.6). Persistent errors—such as the DL–BiLSTM's low probability on the Dot-Com bubble (case 3-c) and high probability on the 1987–1995 non-bubble interval (case 3-g)—suggest that the current feature set (CPI, PPI, Federal Funds Rate, 10-Year Treasury Yield, S&P 500 P/E, DJIA) may not fully capture the mechanisms underlying speculative regimes in this historical sample. This does not rule out value in these indicators; rather, it indicates that these six, by themselves, were not sufficient to drive consistently accurate probabilistic forecasts here.

- **Single-regime configurations generalize poorly; mixed-regime training improves DL robustness but not universally:** Training on only bubble or only non-bubble data produced clear distribution-shift errors for DL and DL-LLM variants (higher Brier in Approaches 1–2). The Standalone LLM was comparatively resilient in single-regime settings (best scores in Approaches 1–2) but still exhibited directional biases on out-of-regime episodes. When exposed to both regimes (Approach 3), the DL–BiLSTM achieved the strongest overall calibration (Brier 0.179), while Transformer-backbone variants degraded markedly (e.g., 0.603–0.703), underscoring architecture sensitivity to training regime composition.

- **Precise and Objective Computational Definitions of Financial Bubbles and Relevant Macro-Financial Data are Required for Effective AI-Based Bubble Prediction:** Recent research has demonstrated the sophisticated analytical capabilities of Large Language Models when guided by meticulously designed prompts. For instance, recent research shows that complex problem-solving can be achieved through prompt engineering that decomposes

challenges into sequential reasoning steps while clearly defining the LLM's role and behavior[?] . Drawing from this methodological insight, developing a objective computational definition of financial bubbles and relevant macro-financial concepts emerges as a essential prerequisite for effective prompt engineering in the financial domain. The current broad and subjective definitions of bubbles need to be standardized and made more objective[?] . As LLM capabilities rapidly advance, such definitional clarity is necessary to transform financial bubble prediction from a historically challenging endeavor into a partially tractable analytical task.

- **Practical implication for industry and policy:** The results suggest using AI bubble probabilities as complementary early-warning indicators rather than stand-alone triggers: LLM ensembles are comparatively sharper when the data reflect a single regime (our Approaches 1–2), while sequence DL with a BiLSTM backbone is more stable when regimes mix (Approach 3). Because DL-LLM outputs largely anchor to the DL backbone, oversight and governance should focus primarily on the backbone's behavior and calibration, not the LLM wrapper. For investors, this points to integrating probabilities into risk budgeting and scenario monitoring (not binary buy/sell rules); for regulators and policy makers, it supports requiring standardized bubble labels, out-of-sample calibration reporting, and change-management disclosures to reduce pro-cyclicality and improve comparability across models.

4.3 Limitations

The definition of financial bubbles remains a subject of persistent debate among economists, financial analysts, and market specialists, with considerable disagreement regarding which historical events genuinely constituted bubble phenomena. As noted in section 4.1, precisely delineating bubble characteristics and identifying specific timeframes for model training presents a formidable challenge that transcends technical implementation concerns, extending into fundamental theoretical disputes within both financial and economic disciplines.

5 Acknowledgements

I would like to express my sincere gratitude to Mr. Sim Joongwon for his invaluable guidance and mentorship throughout this research. His technical insights, methodological guidance, and consistent support were instrumental in overcoming key challenges and bringing this work to fruition.

References

1 D. Petersen, *A brief history of financial bubbles*, 2014, https://www.gsb.stanford.edu/insights/finance-investing/a-brief-history-financial-bubbles, Accessed: 2025-10-11.

2 S. R. H. Han and M. M. R. Han, *Journal of Urban Economics*, 2015, **88**, 160–170.

3 F. Rad, *MSc thesis*, University of Manitoba, Department of Agribusiness and Agricultural Economics, Winnipeg, Canada, 2018.

4 Financial Crisis Inquiry Commission, *The Financial Crisis Inquiry Report: Final Report of the National Commission on the Causes of the Financial and Economic Crisis in the United States*, U.S. Government Publishing Office, Washington, DC, 2011, p. 662.

5 F. Biagini, L. Gonon, A. Mazzon and T. Meyer-Brandis, *arXiv*, 2024.

6 M. Manian and P. Kayal, *Decision Analytics Journal*, 2025, **15**, 100165.

7 K. L. Tran, H. A. Le, C. P. Lieu and D. T. Nguyen, *International Journal of Financial Studies*, 2023, **11**, 133.

8 F. B. Kabran and K. D. Ünlü, *Journal of Applied Statistics*, 2020, **48**, 2776–2794.

9 S. Anbaee Farimani, M. Vafaei Jahan, P. Soltani and A. Milani Fard, Proceedings of the 38th Canadian Conference on Artificial Intelligence, 2025, pp. 1–11.

10 R. Dolphin, B. Smyth and R. Dong, *arXiv*, 2024.

11 RBC Borealis Research, *Self-supervised learning in time-series forecasting — a contrastive learning approach*, 2024, https://rbcborealis.com/research-blogs/self-supervised-learning-in-time-series-forecasting-a-contrastive-learning-approach/, Accessed: 2025-10-11.

12 N. Vinden, R. Saqur, Z. Zhu and F. Rudzicz, *arXiv*, 2025.

13 A. G. Kim, M. Muhn and V. V. Nikolaev, *arXiv*, 2025.

14 X. Wang and M. Brorsson, Proceedings of the Joint Workshop of the 9th Financial Technology and Natural Language Processing (FinNLP), the 6th Financial Narrative Processing (FNP), and the 1st Workshop on Large Language Models for Finance and Legal (LLMFinLegal), 2025, pp. 196–206.

15 Y. Cao, Z. Chen, P. Kumar, Q. Pei, Y. Yu, H. Li, F. Dimino, L. Ausiello, K. P. Subbalakshmi and P. M. Ndiaye, *arXiv*, 2025.

16 S. Mahdavi, J. Chen, P. K. Joshi, L. Huertas Guativa and U. Singh, *arXiv*, 2025.

17 D. T. Pele and D. S. Costea, *Expert Systems with Applications*, 2025, **236**, 121444.

18 G. Bhatia, E. M. B. Nagoudi, H. Cavusoglu and M. Abdul-Mageed, *arXiv*, 2024.

19 J. Wei and D. Zhou, *Language models perform reasoning via chain of thought*, 2022, https://research.google/blog/language-models-perform-reasoning-via-chain-of-thought/, Google Research Blog.

20 A. Lopez-Lira, *arXiv*, 2025.

21 Google Colaboratory Team, *Google Colab*, 2025, https://colab.google/.

22 M. S. Vyas and R. K. Chatterjee, *SSRN Working Paper*, 2025.

23 I. Arawjo, C. Swoopes, P. Vaithilingam, M. Wattenberg and E. L. Glassman, *arXiv*, 2024.

24 M. H. Baumann and A. Janischewski, *arXiv*, 2025.

25 Macrotrends, *S&P 500 historical annual returns (1927–2025)*, 2025, https://www.macrotrends.net/2526/sp-500-historical-annual-returns.

26 B. Carlson, *The Nifty Fifty and the old normal*, 2020, https://awealthofcommonsense.com/2020/07/the-nifty-fifty-and-the-old-normal/.

27 A. Hill, *Historic run of the Dow Jones from the October '87 crash to 20,000*, 2016, https://www.tradingsim.com/blog/historic-run-dow-jones-october-87-crash-20000.

28 D. Bernhardt and M. Eckblad, *Stock market crash of 1987*, 2013, https://elischolar.library.yale.edu/ypfs-documents/1539/.

29 J. Kagan, *Black Monday*, 2024, https://www.investopedia.com/terms/b/blackmonday.asp.

30 P. R. Koudijs, *Dot-com bubble*, 2025, https://handwiki.org/wiki/Dot-com_bubble.

31 Federal Reserve Bank of St. Louis, *S&P CoreLogic Case-Shiller U.S. National Home Price Index (CSUSHPINSA)*, 2025, https://fred.stlouisfed.org/series/CSUSHPINSA.

32 Macrotrends, *U.S. GDP growth rate (1961–2024)*, 2025, https://www.macrotrends.net/global-metrics/countries/usa/united-states/gdp-growth-rate.

33 K. R. Kliesen, *The Great Inflation*, 2013, https://www.federalreservehistory.org/essays/great-inflation.

34 K. A. McWhirter, *U.S. recessions throughout history: Causes and effects*, 2024, https://www.investopedia.com/articles/economics/08/past-recessions.asp.

35 National Bureau of Economic Research Business Cycle Dating Committee, *The Business Cycle Peak of March 2001*, 2001, https://www.nber.org/reporter/fall-2001/business-cycle-peak-march-2001.

36 K. Bahr, *The Federal Reserve and interest rate changes*, 2025, https://blog.uwsp.edu/cps/2025/03/25/the-federal-reserve-and-interest-rate-changes/.

37 I. Arawjo, *ChainForge (GitHub repository)*, https://github.com/ianarawjo/ChainForge, 2025.