

Universiteit Leiden

Master Computer Science

ChartNarrator: Towards Faithful Chart-to-Text Generation for Financial Time Series

Name:	Yutao Liu
Student ID:	s4213211
Date:	[29/05/2026]
Specialisation:	Artificial Intelligence
1st supervisor:	Prof.dr.ir. J.M.W. Visser
2nd supervisor:	Dr. Niki van Stein

Master's Thesis in Computer Science

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Abstract

Background: Financial institutions, central banks, and policymakers routinely interpret time series charts to communicate quantitative findings to non-specialist audiences; automating this interpretation into coherent textual narratives would reduce the manual effort required and improve the scalability of financial reporting. Generating accurate textual narratives from financial time series charts is a task where factual errors carry consequential downstream effects: a misrepresented trend direction of fabricated numerical claim does not merely reduce fluency but can lead to misinformed economic decisions. Existing multimodal large language model (MLLM) approaches fail along three intertwined dimensions simultaneously: they lack reliable mechanisms for grounding generated statements in the numerical and directional constraints visible in a chart; they cannot produce mechanistic reasoning that remains strictly bounded by chart-visible evidence; and they offer no verified means of evaluating whether generated narratives are factually faithful. No existing framework addresses all three of these challenges in combination.

Aim: This thesis proposes ChartNarrator, a framework for faithful chart-to-text generation applied to financial time series data. Through anchor metadata injection, a two-stage generation pipeline, and an evaluation protocol verified via inter-rater agreement analysis, ChartNarrator systematically targets each of the three gaps identified above.

Method: We construct a dataset of $1,960$ financial time series charts derived from 20 macroeconomic indicators published by Statistics Netherlands (CBS Open Data), covering domains including labour market statistics, price indices, confidence indices, and bankruptcy counts, anchor metadata encoding numerical constraints, temporal boundaries, and variable directional semantics is extracted from each indicator and provided as structured input to the generation pipeline. The pipeline separates factual description from mechanistic reasoning in two consecutive stages: Stage 1 produces a constrained factual narration grounded in chart-extractable values, while Stage 2 generates source-bounded mechanistic reasoning within the boundaries established by Stage 1. This separation prevents interpretive language from overriding observational accuracy before a verifiable factual layer has been established. We fine-tune Qwen2-VL-7B-Instruct via QLoRA on approximately 1,234 pipeline-generated training pairs and evaluate the resulting model against Zero-Shot, No-Anchor, and Text-Only baselines on a held-out set of 124 samples. Evaluation quality is assessed using an LLM-as-a-Judge framework, and judge reliability is independently verified against human annotation via Cohen's Kappa.

Results: We identify four categories of faithfulness violations in MLLM-generated financial chart narratives, including Latent-state Mechanism Wording, a violation type not captured by existing hallucination taxonomies. Anchor metadata injection produces an overall weighted average (WA) improvement of $+1.802$ on a 1–5 integer scoring scale ($p < 0.001$), with the largest gain on the Mechanistic Grounding dimension; we additionally observe that the anchor block functions as a format conditioning trigger during generation. LLM-as-a-Judge reliability reaches Moderate overall agreement ($\kappa = 0.463$), with the Dependency Strength dimension achieving substantial agreement ($\kappa = 0.659$). In the comparative evaluation, the fine-tuned vision–language model (WA = 4.306) performs comparably to the text-only baseline (WA = 4.422), while both conditions substantially outperform the No-Anchor and Zero-Shot conditions.

Conclusion: Reliable faithful chart narration requires the systematic integration of constraint design, task separation, and evaluation verification; no single component produces adequate faithfulness in isolation. The violation taxonomy, anchor design principles, and Kappa verification protocol developed in this thesis offer transferable reference points for faithful generation tasks beyond the financial chart narration setting.

Contents

Abstract . . . 1

1 Introduction **3**
1.1 Problem Statement . . . 3
1.2 Research Questions . . . 5
1.3 Overview of the Thesis . . . 5

2 Related work **6**
2.1 Chart Understanding and Narrative Generation . . . 6
2.2 Financial and Economic Text Analysis . . . 8
2.3 Faithful NLG and Hallucination Evaluation . . . 9

3 The ChartNarrator Framework **10**
3.1 Framework Overview . . . 11
3.2 Data Construction . . . 12
3.3 Anchor Metadata Injection . . . 13
3.4 Two-Stage Generation Pipeline . . . 14
3.5 Evaluation Protocol . . . 15
3.6 Generation and Judge Models . . . 17
3.7 Fine-Tuning the Target Model . . . 17

4 Results **19**
4.1 Dataset Statistics and Training Set Construction . . . 19
4.2 Faithfulness Violation Taxonomy . . . 22
4.3 Effects of Anchor Metadata on Generation Quality . . . 24
4.4 Inter-Rater Reliability of LLM-as-a-Judge . . . 26
4.5 Comparative Model Performance . . . 27
4.6 Qualitative Output Analysis . . . 30

5 Discussion **35**
5.1 Causes and Implications of the Violation Taxonomy . . . 35
5.2 Mechanism and Scope of Anchor Metadata Effects . . . 36
5.3 Interpreting Judge Reliability Across Evaluation Dimensions . . . 37
5.4 Modality, Fine-tuning, and Training Set Effects . . . 39
5.5 Framework-Level Observations . . . 40
5.6 Limitations . . . 42

6 Conclusions and Future Work **44**
6.1 Conclusions . . . 44
6.2 Future work . . . 45

A Appendix **50**
A.1 Data Construction Tables . . . 50
A.2 Two-Stage Anchor Comparison Tables . . . 52
A.3 Two-Stage Generation Prompts . . . 54
A.4 Two-Stage Judge Prompts . . . 57
A.5 Fine-Tuning Parameters . . . 61

A.6 Kappa Stratum 61

Chapter 1

Introduction

In this chapter, we introduce the research problem and our research questions.

1.1 Problem Statement

Financial institutions, central banks, and policymakers often use time series charts to convey the results of quantitative analysis. The accompanying textual descriptions not only describe the data but also offer explanations: they provide the economic significance of visual patterns (trend direction, structural turning points, dependencies between variables) in a way that is understandable to an audience without specialized training [1]. Transforming these visual representations into coherent textual narratives has historically been the work of domain experts, a time-consuming process that is difficult to scale to the data volumes required for modern financial analysis.

The emergence of Multimodal Large Language Models (MLLMs) has opened a practical path to automating this transformation process. MLLMs can jointly process visual and textual inputs to generate natural language output. Recent models and benchmarks, including ChartQA [2] and CharXiv [3], demonstrate that MLLMs can answer questions and generate chart titles for various types of charts. However, most of these benchmarks are based on synthetic or general-domain graphs and evaluate models primarily based on fluency and surface correctness rather than faithfulness. Here, faithfulness denotes the property that every factual statement in the generated narrative can be verified against the source data [3, 4].

Financial time series charts present a class of visual inputs that is qualitatively more demanding than the general-purpose charts on which most MLLMs are trained and evaluated [3]. Unlike bar charts or pie charts that encode discrete categorical comparisons, time series charts require a reader (either human or model) to track continuous numerical trajectories, identify directional inflections, locate extrema within specific temporal windows, and interpret the economic valence of variable movements [5].

Figure 1.1 illustrates the type of chart narrative this thesis aims to generate: a three-paragraph source-bouned account that moves from observed structure to mechanistic interpretation and explicit dependency resolution, grounded throughout in the numerical and temporal evidence visible in the chart.

Three intertwined challenges make faithful narration of such charts resistant to existing approaches. First, general-purpose MLLMs lack reliable grounding mechanisms for the fine-grained numerical and directional relationships encoded in financial time series. A model may correctly identify that a curve is declining, yet misread the magnitude of the decline, misplace the timing of a trough, or, critically, misinterpret the economic direction of a variable: falling unemployment, for instance, represents labour market improvement, not deterioration, a distinction that requires domain-semantic grounding beyond visual perception alone [3].

Second, faithful chart narration in financial contexts demands more than surface description. It requires mechanistic reasoning: identifying structural patterns, inferring causal dependencies between variables, and constructing internally consistent explanatory chains, all while remaining strictly grounded in what the chart itself shows, without importing external event knowledge that cannot be verified against the visual source [6]. This closed-system constraint is both necessary for faithfulness and difficult to enforce in open-ended generation.

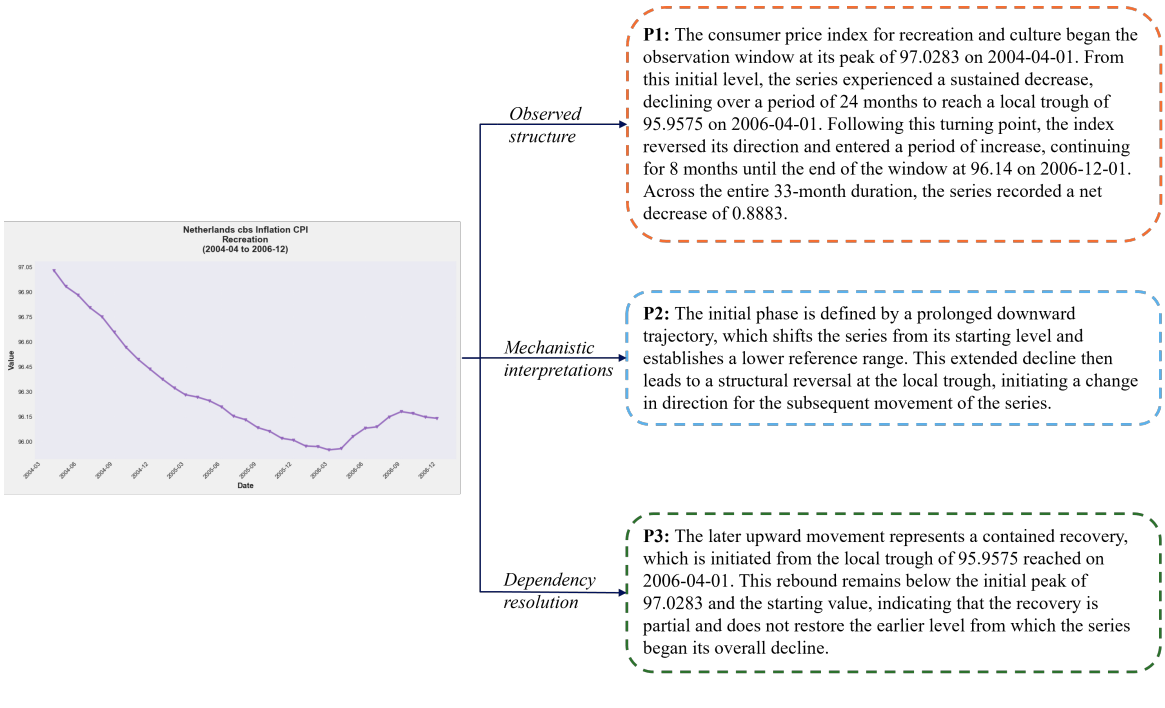


Figure 1.1: An example of the type of chart narrative ChartNarrator aims to generate: a three-paragraph account of a Netherlands CBS Inflation CPI Recreation chart (April 2004 – December 2006), moving from factual description to mechanistic interpretation and dependency resolution, grounded strictly in the chart's visible evidence.

Third, the standard tools for evaluating generation quality are inadequate for this task. Metrics such as BLEU [7] and ROUGE [8] measure lexical overlap rather than factual correctness, and are insensitive to the kinds of semantic errors, such as reversed trend directions, misattributed extrema, fabricated causal claims that constitute faithfulness violations in this domain [4]. LLM-as-a-Judge approaches [9] offer a more flexible alternative, but whether these judges reliably align with human annotators on higher-order dimensions specific to financial chart narration such as dependency and structural coherence, remains an open question.

Three lines of prior work bear on the problem, but none addresses it in full. The first concerns chart understanding and narrative generation. Benchmarks such as ChartQA [2] and CharXiv [3], along with models such as ChartLlama [10] and MatCha [11], have substantially advanced the capacity of MLLMs to interpret and describe charts. However, these systems are developed and evaluated primarily on synthetic or general-domain chart data, and their evaluation criteria prioritize fluency and question-answering accuracy, without requiring the kind of verifiable, data-grounded faithfulness that financial narration demands.

The second line concerns financial and economic text analysis. Work in this area engages with real financial data under strict accuracy requirements, encompassing tasks such as earnings report generation, economic indicator forecasting, and sentiment extraction from textual disclosures [5]. Yet this research stream operates almost exclusively on structured tabular inputs or pre-existing text, and does not address the challenge of generating mechanistic narratives from visual chart inputs.

The third line concerns faithful natural language generation and the evaluation of hallucination in generated text. Maynez et al. [6] established foundational distinctions between faithful and unfaithful abstractive generation, and Ji et al. [4] provide a systematic taxonomy of hallucination types across Natural Language Generation (NLG) tasks. LLM-as-a-Judge frameworks [9] have extended automatic evaluation toward semantic dimensions. However, none of this work is grounded in chart-based or financial contexts, and the reliability of automatic judges on the specific higher-order dimensions relevant to financial chart narration remains unexamined.

No existing framework simultaneously addresses grounded data alignment, mechanistic narrative gener-

ation, and verified faithfulness evaluation for financial time series charts. ChartNarrator is designed to occupy this intersection.

To address these challenges, this thesis proposes ChartNarrator, a framework for faithful chart-to-text generation applied to financial time series data. ChartNarrator operates as a modular pipeline organized around three design principles: grounding generation in structured anchor metadata that encodes numerical constraints, temporal boundaries, and variable directional semantics as explicit constraints; separating factual description from mechanistic reasoning about relationships between variables through a two-stage generation architecture; and verifying evaluation reliability through inter-rater agreement analysis [12] before drawing conclusions from automatic judgments.

The framework culminates in a fine-tuning experiment in which high-quality pipeline outputs are used to train a faithfulness-oriented MLLM via parameter-efficient fine-tuning [13]. The resulting model is then evaluated against Zero-Shot, No-Anchor, and Text-Only baselines under a unified faithfulness evaluation protocol.

1.2 Research Questions

This thesis addresses four research questions:

RQ1. What types of faithfulness violations occur most frequently when MLLMs generate narratives from financial time series charts, and what underlying causes drive each violation type?

RQ2. How does structured anchor metadata affect the faithfulness and explanatory quality of generated narratives?

RQ3. To what extent can LLM-as-a-Judge reliably approximate human judgments across the faithfulness dimensions defined in this framework, and under what conditions does systematic disagreement arise?

RQ4. How does a fine-tuned MLLM trained under the ChartNarrator framework compare against Zero-Shot, No-Anchor, and Text-Only baselines under a unified faithfulness-oriented evaluation framework?

1.3 Overview of the Thesis

This thesis makes the following contributions. We develop a systematic taxonomy of faithfulness violations in MLLM-generated narratives for financial time series charts, identifying four recurring error types and their underlying causes. We propose a structured anchor metadata framework that encodes numerical, temporal, and semantic constraints to ground generation in chart-visible data. We design a two-stage generation pipeline that separates factual description from mechanistic reasoning and enforces closed-system compliance at both stages. We construct an evaluation framework that combines constrained LLM-as-a-Judge scoring with inter-rater reliability verification [12], providing an empirical characterization of the dimensions on which automatic judgment diverges from human assessment. Finally, we present empirical evidence on the capabilities and limitations of current MLLMs in faithful financial time series chart narration, including fine-tuning results and baseline comparisons under a unified evaluation protocol.

The remainder of this thesis is structured as follows. Chapter 2 reviews related work across the three research streams identified above: chart understanding and narrative generation, financial text analysis, and faithful NLG and hallucination evaluation. Chapter 3 describes the ChartNarrator framework, including dataset construction, anchor metadata design, the two-stage generation pipeline, the evaluation protocol, generation and judge model selection, and fine-tuning the target model. Chapter 4 presents the experimental results, covering the hallucination taxonomy, the effects of anchor metadata injection, the reliability analysis of LLM-as-a-Judge against human annotation, and the comparative performance of the fine-tuned model against baselines. Chapter 5 discusses the implications of these findings, the limitations of the current framework, and directions for future work. Chapter 6 concludes the thesis.

Chapter 2

Related work

This chapter reviews three lines of prior work that bear on the ChartNarrator framework: chart understanding and narrative generation, financial and economic text analysis, and faithful NLG and hallucination evaluation.

2.1 Chart Understanding and Narrative Generation

Generating natural language descriptions from charts involves multiple layers of semantic interpretation rather than a single homogeneous task. Lundgard and Satyanarayan [14] proposed a four-level model of semantic content in visualization descriptions, ranging from references to visual construction properties to higher-level domain insights and causal interpretations. Their empirical analysis of more than two thousand human-authored descriptions showed that readers—particularly those who rely on textual descriptions as their primary access channel—tend to consider higher-level content the most informative. However, existing automatic captioning systems remain largely concentrated at the lower levels of this hierarchy. This gap between what readers expect from chart descriptions and what current models produce represents a central challenge in chart narrative generation.

Building on this observation, Kantharaj et al. [1] formalized chart-to-text generation as a large-scale benchmark task. They introduced a dataset containing more than forty thousand chart–summary pairs and evaluated a range of neural generation models. Their results showed that even the strongest systems frequently produced hallucinations and factual inconsistencies, suggesting that faithful narrative generation from charts is a distinct research problem that general-purpose language generation approaches do not adequately address.

Substantial progress in chart understanding has since been driven by the development of specialized benchmarks and pretrained models. ChartQA [2] established a widely adopted evaluation framework for chart-based visual question answering by combining human-authored and template-generated questions across diverse real-world chart sources. Building on this benchmark, MatCha [11] introduced chart-specific pretraining objectives centred on chart derendering and mathematical reasoning, demonstrating that targeted pretraining can significantly improve numerical grounding in vision–language models.

UniChart [15] further extended this direction by proposing a dual-level pretraining strategy that jointly models low-level visual element extraction and higher-level chart comprehension tasks, achieving state-of-the-art performance across several downstream benchmarks. Despite these improvements, persistent failures in factual correctness and complex numerical reasoning have been reported across these systems, indicating that chart-specific pretraining alone does not resolve faithfulness in generated outputs.

ChartLlama [10] explores a complementary direction by adapting a large language model to chart understanding and generation through instruction tuning on synthetically constructed data. Its evaluation relies on GPTScore [16], a probability-based metric that measures generative likelihood rather than factual alignment with the source chart data. A different approach to chart understanding is modality decomposition, as illustrated by DePlot [17]. DePlot first converts a chart image into a linearized data table and then applies a language model to reason over the extracted representation. While this approach improves performance on chart question answering tasks, the intermediate tabular representation may

discard information that is important for time-series charts, where temporal structure and directional semantics are central to interpretation.

Recent research has also begun to document the capability boundaries of multimodal large language models in chart reasoning. CharXiv [3] introduced a benchmark that distinguishes between descriptive questions, which require extracting surface-level visual information, and reasoning questions that require multi-step inference over chart content. Evaluations across several state-of-the-art MLLMs reveal a substantial performance gap: models that perform adequately on descriptive questions show markedly weaker performance on reasoning questions, with the best systems still falling well short of human-level accuracy.

For financial time-series chart narration, this reasoning gap is directly consequential: a reader must track structural patterns in temporal trajectories, infer directional relationships between variables, and construct coherent explanatory narratives grounded in the observed data.

At the level of generation rather than question answering, Huang et al. [18] conducted a systematic analysis of factual errors produced by large vision–language models in chart captioning tasks. Their study identified recurring error categories and proposed a correction framework based on structured error classification. Of the studies reviewed here, this work is most directly comparable to the present research in its focus on factual inconsistency in chart-generated text. It differs, however, in addressing general-purpose charts and framing the problem as post-hoc error correction rather than constraint-guided generation.

Complementary evidence is provided by VisText [19], a dataset of semantically annotated chart captions, which shows that models trained on chart-caption pairs tend to produce outputs concentrated at lower semantic levels, while higher-level interpretations such as trend identification and causal inference remain the least reliably generated.

The challenge of grounding generated text in structured source data is not unique to chart-based tasks. In the broader data-to-text literature, Parikh et al. [20] introduced ToTTo, a controlled table-to-text benchmark in which models generate descriptions of explicitly highlighted table cells. By delimiting the relevant source content, the benchmark makes grounding an explicit task requirement. Results on ToTTo show that faithful generation remains difficult even when the relevant data is clearly specified, and that models trained without explicit grounding supervision frequently produce unsupported statements.

Beyond generation constraints, the evaluation of grounded outputs also requires metrics sensitive to source faithfulness. This grounding principle is closely related to the design of ChartNarrator's anchor metadata. Similar to the task formulation introduced in ToTTo, ChartNarrator injects structured constraints—such as numerical values, temporal boundaries, and variable directional semantics—directly into the generation prompt to restrict the space of permissible narratives. Dhingra et al. [21] examined the limitations of standard automatic metrics for faithfulness evaluation in data-to-text generation and proposed the PARENT metric. PARENT evaluates generated text against both the source data table and a reference string, penalizing statements that cannot be entailed by the input data.

The metric was motivated by the observation that widely used measures such as BLEU [7] and ROUGE [8] reward lexical similarity to reference texts even when generated statements are not factually supported by the underlying data. Although PARENT was developed for tabular inputs and cannot be directly applied to chart-based generation, it formalized the conceptual distinction between surface fluency and source faithfulness, a distinction that is central to the evaluation design in this thesis.

None of these research streams addresses the problem in full. Work on chart understanding has produced increasingly capable models for question answering and summarization, but these systems continue to fall short on the dimensions of faithfulness required for financial narration, including numerical precision, temporal grounding, and directional semantic accuracy.

Data-to-text research has formalized principles of source-grounded generation and highlighted the limitations of standard evaluation metrics, but primarily assumes structured tabular inputs and does not address the perceptual and semantic challenges of chart-based visual data. Meanwhile, work on faithful NLG and hallucination evaluation has developed taxonomies of generation failures and more semantically sensitive evaluation approaches, yet has rarely been applied to chart-based or financial contexts.

As a result, no existing framework simultaneously addresses grounded data alignment, mechanistic narrative generation, and verified faithfulness evaluation for financial time-series charts. ChartNarrator is

designed to occupy this intersection. Anchor metadata injection converts grounded data alignment into an explicit generation constraint, the two-stage pipeline separates factual grounding from mechanistic reasoning in a manner consistent with the semantic hierarchy proposed by Lundgard and Satyanarayan [14], and inter-rater reliability verification [12] establishes the conditions under which automatic faithfulness judgments can be trusted.

2.2 Financial and Economic Text Analysis

Financial English differs systematically from general English at the semantic level. This difference has direct consequences for NLP systems trained on general-domain corpora. Loughran and McDonald [22] show empirically, using a large corpus of SEC 10-K filings, that the Harvard General Inquirer dictionary frequently classifies many neutral financial terms (e.g., *tax*, *liability*, *cost*) as negative words. This misclassification reflects a structural mismatch between general-purpose sentiment lexicons and the semantic conventions of financial language.

Loughran and McDonald [5] further note that domain-specific vocabulary, hedging conventions, and the pragmatic functions of financial disclosure together create a semantic distribution that general-purpose language models are not designed to interpret. The problem extends beyond lower performance on financial text: the underlying semantic categories used in general-domain NLP do not transfer to the financial domain.

In response to this challenge, the dominant strategy in financial NLP has been domain adaptation: continuing to pre-train language models on financial-domain corpora. Araci [23] applies this approach to BERT on a large financial corpus to produce FinBERT, achieving substantial improvements over general-purpose BERT on financial sentiment classification benchmarks. These results demonstrate that domain adaptation can effectively mitigate the semantic mismatch between financial language and general-domain training data.

Wu et al. [24] extend this paradigm to the large-model era by training BloombergGPT, a 50-billion-parameter language model, on a financial corpus containing 363 billion tokens. Despite the scale of these models, the fundamental task formulation remains unchanged. The input consists of existing financial text or structured data, and the output is classification labels, predictions, or extracted insights. None of these systems addresses the generation of narrative text directly from visual financial charts.

Directional changes in financial indicators often have asymmetric economic interpretations. A decline in the unemployment rate signals improvement in labour market conditions, whereas a decline in GDP indicates economic contraction, even though both correspond to numerical decreases. The economic meaning of a directional change therefore depends on the semantic role of the variable rather than on the direction of movement alone. In existing financial NLP research, this phenomenon appears implicitly in the construction of annotated datasets. Sentiment lexicons and classification models rely on polarity labels assigned through human annotation, and these labels already incorporate judgments about the directional semantics of financial variables [22].

However, these judgments remain embedded in annotation practices and are not formulated as explicit operational rules. This limitation becomes more consequential in generation settings. A model generating narrative text must both recognize the directional semantics of a variable and translate that recognition into an appropriate economic interpretation. Errors in this step lead to what this thesis refers to as directional hallucination. ChartNarrator addresses this issue by explicitly encoding the directional semantics of each variable in the anchor metadata, thereby converting an implicit annotation into a formal generation constraint.

Across these approaches, existing financial NLP research operates within a shared paradigm: the input consists of existing text or structured data, and the task centres on understanding, classification, or prediction. This line of work does not address the problem of generating narrative text from visual charts, nor does it examine how numerical constraints and directional semantics can be operationalized as explicit constraints in the generation process. ChartNarrator addresses a task this literature leaves largely unexplored: generating faithful mechanistic narratives directly from financial time-series charts.

2.3 Faithful NLG and Hallucination Evaluation

The core requirement of faithful generation is that every proposition in the generated text must be supported by the source data, establishing faithfulness as a central quality dimension of natural language generation. Maynez et al. [6] systematically analyzed faithfulness violations in abstractive summarization, distinguishing two categories of error: intrinsic violations, which contradict the source content, and extrinsic violations, which introduce information not present in the source. This taxonomy establishes that hallucination comprises structured error types whose defining characteristic is their relationship to the source, rather than occurring as random noise.

Ji et al. [4] extended this analysis to a broader range of NLG tasks, documenting the prevalence of hallucination across diverse generation scenarios and further systematizing the taxonomy of detection approaches. These findings collectively indicate that faithfulness evaluation requires distinguishing source-entailed statements from unsupported ones, a judgment that cannot be reduced to surface-level lexical matching.

Standard automatic evaluation metrics, BLEU [7] and ROUGE [8], assess generation quality through lexical overlap between generated and reference text. The fundamental limitation of this mechanism is that there is no reliable correspondence between lexical overlap and factual accuracy. A narrative that describes a rising unemployment rate as a positive economic signal, for instance, may remain superficially similar to a faithful reference while inverting its factual meaning. This kind of violations including misattributed extrema and errors in variable directional semantic is invisible to metrics that operate purely on lexical overlap.

Zheng et al. [9] introduced the LLM-as-a-Judge paradigm to address this limitation directly, demonstrating through MT-Bench and Chatbot Arena that a capable LLM evaluator correlates substantially with human preference judgments across multiple quality dimensions. Liu et al. [25] further systematized this approach through the G-Eval framework, employing GPT-4 as an evaluator and outperforming traditional metrics on dimensions including consistency and fluency through a chain-of-thought scoring process. However, the reliability of an LLM judge as an evaluation tool requires independent verification, since high automated scores do not imply genuine alignment with human judgment.

Cohen's Kappa [12] provides the standard framework for this verification: it measures inter-rater agreement while correcting for chance, thereby quantifying the degree of alignment between automated and human judgments.

Although prior work such as Zheng et al. [9] and the G-Eval framework [25] demonstrates the feasibility of LLM-as-a-Judge in general evaluation settings, these validation scenarios share an important characteristic: they focus on open-ended dialogue evaluation or the linguistic quality of news summarization. Such tasks place substantially lower demands on numerical accuracy and temporal reasoning than financial time series chart narrative generation.

In financial chart narration, faithful generation demands adherence to numerical constraints, temporal boundaries, and variable directional semantics, requiring the Judge to assess structural reasoning rather than surface quality alone. Preliminary analysis confirms that LLM judges can assign uniformly high scores on these dimensions, masking quality differences that the human annotator identifies; existing work does not systematically quantify this reliability boundary.

ChartNarrator addresses this gap by computing Cohen's Kappa [12] between LLM judges and human annotators across four faithfulness dimensions, identifying where automated judgment is reliable and where it diverges from human assessment. Evaluation reliability is thus treated as an empirical question within the pipeline.

Chapter 3

The ChartNarrator Framework

This chapter describes the ChartNarrator framework across five components: dataset construction, anchor metadata design, the two-stage generation pipeline, the evaluation protocol, and model fine-tuning. The framework we presented here is the result of iterative design cycles: a preliminary single-stage generation system revealed systematic faithfulness failures that motivated the introduction of anchor metadata injection and the two-stage pipeline structure; subsequent rounds of generation & judge prompt iteration and Kappa verification refined the evaluation protocol into its current form. Each design decision reported in this chapter reflects an empirical observed failure modes.

Each annotation round produced systematic observations about recurring failure patterns, most notably the prevalence of latent-state wording, which were used to tighten generation constraints and calibrate Judge threshold setting before the final evaluation round. A dedicated annotation interface was developed to support this process and is described in Section 3.5.

3.1 Framework Overview

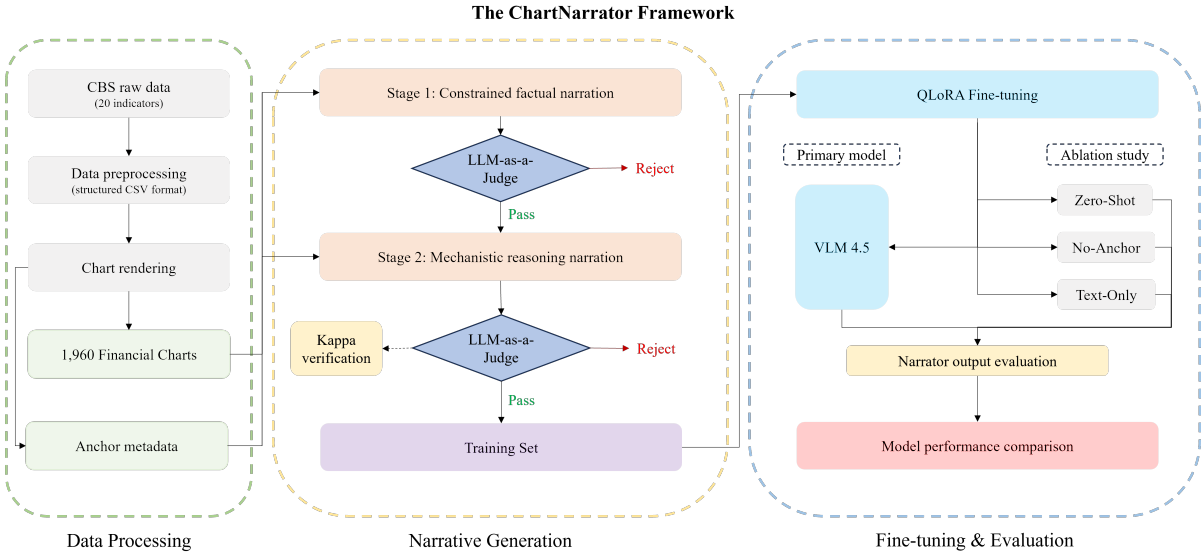


Figure 3.1: Overview of the ChartNarrator pipeline. The pipeline proceeds left to right across three phases. *Data Processing* constructs the input representation: CBS Open Data (20 indicators) is cleaned and rendered into 1,960 financial time series charts, and anchor metadata encoding numerical extrema, temporal boundaries, and variable directional semantics is extracted for each chart. *Narrative Generation* operates in two consecutive stages: Stage 1 produces constrained factual narrations grounded in the Stage 1 anchor schema, and Stage 2 produces source-bounded mechanistic reasoning conditioned on retained Stage 1 outputs and the Stage 2 anchor schema. An LLM-as-a-Judge quality gate follows each stage; outputs that fail the gate are discarded. Following the Stage 2 gate, a Kappa verification step confirms inter-rater reliability between the Judge and a human annotator on a stratified subsample ($n = 80$). *Fine-tuning & Evaluation* uses the filtered narrative pairs as a training set to fine-tune Qwen2-VL-7B-Instruct via QLoRA [13], yielding VLM 4.5. VLM 4.5 and three baselines—Zero-Shot, No-Anchor, and Text-Only—are then compared under a unified faithfulness evaluation protocol (RQ4).

ChartNarrator is organized as a modular pipeline because faithfulness can fail at different points in chart-to-text generation, and each point requires a different form of control. Numerical content must remain aligned with the rendered chart, factual description must be separated from mechanistic reasoning so that interpretation does not override observation (a failure mode observed in preliminary single-stage experiments, where the model entered interpretive language before numerical facts had been stably established), and automatic evaluation must be shown to be reliable before it is used to filter outputs or support model comparison.

Figure 3.1 presents an overview of the complete pipeline. These requirements are addressed through a sequence of connected modules rather than through a single end-to-end generation step. During data processing, charts are paired with anchor metadata that encodes salient numerical values, temporal boundaries, and variable directional semantics as explicit constraints for later generation. During generation, Stage 1 is restricted to constrained factual narration, whereas Stage 2 builds on retained Stage 1 outputs to produce source-bounded mechanistic reasoning. During evaluation, automatic judgments are incorporated only within a protocol that includes inter-rater agreement verification [12].

The three modules thus implement the three design principles introduced in Chapter 1: anchor metadata as explicit constraints, the separation of factual description from mechanistic reasoning, and evaluation reliability verification.

These design choices also determine the order of the pipeline. Data processing establishes the representational basis on which later stages depend by defining the chart as both a visual input and a structured object with accessible constraints. The two-stage generation pipeline then produces candidate narratives under controlled task separation and layered filtering. High-quality outputs are subsequently consolidated into a training set and used to fine-tune Qwen2-VL-7B-Instruct with QLoRA [13].

The resulting model is finally compared with baseline systems within a unified faithfulness-oriented evaluation framework. This chapter follows that progression. Section 3.2 describes dataset construction, Section 3.3 introduces the design of anchor metadata, Section 3.4 details the two-stage generation pipeline, and Section 3.5 presents the evaluation protocol, including LLM-as-a-Judge configuration, inter-rater reliability verification, training set construction, and baseline comparison.

3.2 Data Construction

Data construction is grounded in authoritative macroeconomic and financial time-series data obtained from Statistics Netherlands (CBS).[1] The use of official statistical sources ensures that the numerical values underlying each chart are reliable before any later generation or evaluation step is introduced. Raw time-series data were retrieved from the CBS Open Data platform via its public API and preprocessed before plotting.

The selection of financial indicators follows two criteria. The first is semantic signal strength, meaning that directional changes in the series carry a consistent and interpretable economic meaning. Indicators are excluded when their movements are numerically negligible or when the economic interpretation of a directional change cannot be reliably assigned from the chart alone.

This selection strategy prioritizes narratively informative series over dataset scale, since charts that are excessively flat or weakly negligible do not provide a suitable basis for grounded chart-to-text generation. Selected indicators span several macroeconomic domains, including confidence indices, price indices, labor market statistics, construction activity, and bankruptcy counts, covering both flow and stock variable types. A complete list of selected indicators is provided in Table A.1.

Before plotting and rendering, the raw financial time-series data are preprocessed to reduce seasonal noise and to make medium-term structure more visible. The pre-processing operation depends on the semantic type of the indicator. For flow variables such as monthly counts of bankruptcies or changes in employment, a 12-month rolling sum is applied so that the resulting chart reflects annualized totals rather than seasonally volatile monthly values. For stock or level variables such as price indices or unemployment rates, a 12-month rolling mean is used to smooth short-term variation while preserving the broader trajectory of the series. For observation windows shorter than 24 months, raw monthly values are used directly without smoothing, as the window is too short for a 12-month rolling operation to produce meaningful output. The preprocessing rules applied to each indicator type are summarized in Table A.2.

Each preprocessed series is divided into randomly sampled observation windows drawn from three temporal scales: short-term windows of 12 to 24 months, medium-term windows of 24 to 96 months, and long-term windows of 96 to 240 months, sampled at a 20/60/20 ratio respectively. This distribution covers both localized structural episodes and extended macrooeconnomic cycles. Detailed windowing parameters are provided in Table A.3.

After windowing, all charts are generated through deterministic plotting scripts rather than collected from external reports or manually redrawn. This design choice preserves exact alignment between the rendered chart image and the underlying numerical series. If the visual input were to diverge from the numerical source, the constraints derived from the data would no longer correspond to the chart presented to the model.

To reduce dependence on a fixed visual template, chart rendering incorporates controlled randomization across plot style, line color, marker type, and grid configuration. For series containing negative values, a diverging bar chart format is used in place of a line chart to preserve sign visibility. Detailed rendering parameters are provided in Table A.4.

Each chart is therefore constructed as a controlled instance rather than as an independently sourced figure. For every rendered chart, the pipeline also extracts a structured set of anchor metadata from the underlying series, so that each chart instance can be paired with machine-readable numerical context for later generation. At the dataset construction stage, the role of this step is to ensure that each instance contains both the visual representation and its corresponding structured numerical context. The design of anchor metadata and the way this context is encoded for subsequent use are described in Section 3.3.

[1] `https://opendata.cbs.nl`

3.3 Anchor Metadata Injection

Anchor metadata serves as a structured control layer between chart construction and narrative generation in ChartNarrator. Its purpose is to convert information that is central to generation, yet difficult for a model to read consistently from the chart image alone, into machine-readable constraints. Since Stage 1 and Stage 2 pursue different generation objectives, the framework does not rely on a single static anchor schema. Instead, we implement anchor extraction in two stage-specific scripts.

Stage 1 requires a numerical and temporal scaffold used to constrain factual narration, whereas Stage 2 requires additional structural cues and interpretive boundaries for source-bounded mechanistic reasoning. The difference between the two extraction procedures therefore reflects the distinct informational demands of the two stages. Figure 3.2 provides an overview of the two anchor schemas and their field groups.

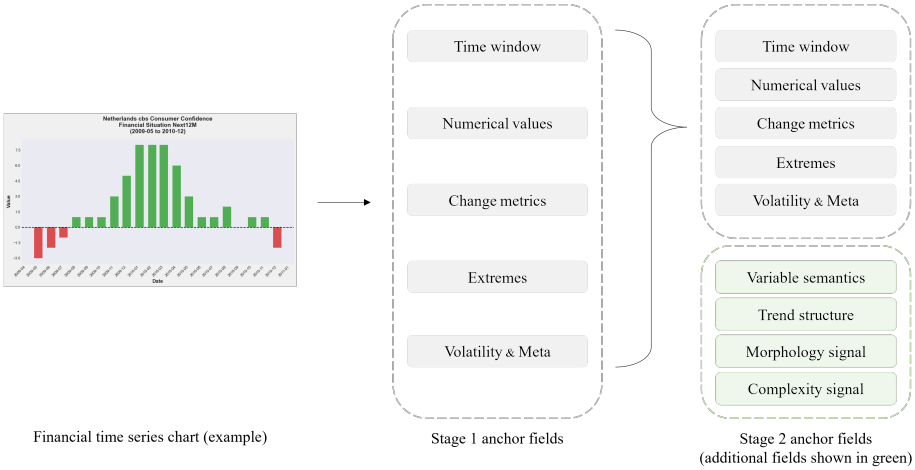


Figure 3.2: Anchor metadata field groups for Stage 1 and Stage 2 generation. Stage 2 inherits all Stage 1 fields and adds four domain-specific groups (shown in green). Full field specifications are provided in Table A.5 and Table A.6.

The Stage 1 anchor schema is designed to provide a low-ambiguity numerical and temporal representation for constrained factual narration. It first specifies the requested time window and valid data range, allowing the generation process to distinguish between the intended observation interval and the interval for which valid data are actually available. This reduces the risk that data unavailability near window boundaries will be misread as genuine trend continuation. The schema also preserves high-precision numerical values, including start and end points, peak and trough values, absolute change, and rate of change.

As a result, factual narration can be grounded in structured numerical evidence instead of approximate visual reading alone. For descriptions that depend on boundary conditions, such as percentage change, annualized change, or the position of extrema, the Stage 1 schema additionally records whether these expressions are valid under the observed window. The anchor layer constrains both the content of factual description and the condition under which particular claims may be made.

The Stage 2 anchor schema extends this factual layer into a representation suitable for source-bounded mechanistic reasoning. As shown in Figure 3.2, We construct the Stage 2 anchor schema by extending the Stage 1 fields with four additional field groups covering variable semantics, trend structure, morphology signal, and complexity signal.

In addition to the basic numerical fields, the schema encodes variable semantics, including unit type, directional meaning, and the interpretation associated with increases or decreases in the value of the series. These fields help fix the semantic direction of the variable in advance, so that similar curve movements are not interpreted in the same way across indicators whose meanings differ.

These structural fields allow the model to distinguish *endpoint continuation*, *structural reversal*, *bounded oscillation*, and *periodic repetition* as different temporal configurations, each corresponding broadly to the morphology families assigned during anchor extraction.

Within Stage 2, anchor metadata also supports morphology-aware routing. This design follows from the observation that different chart morphologies impose different demands on narrative generation. A structurally simple sequence and a sequence with clearly differentiated internal phases do not call for the same form of generative control. We therefore assign each chart to one of four morphology families, including `trend_drift`, `structural_swing`, `oscillatory_or_seasonal_regime`, `simple` and route it to either a simple or a complex generation path on the basis of structural complexity signals. This decision is not derived from surface phase counting alone, but on a combination of temporal structure, boundary behavior, and morphological cues, which helps separate genuinely low-complexity sequences from series that appear visually smooth but remain internally constrained.

Anchor metadata in ChartNarrator thus functions as a stage-conditioned representational interface rather than as a uniform set of annotations attached to the chart. The Stage 1 schema translates core numerical facts into controllable factual constraints, while the Stage 2 schema adds variable semantics, structural form, and complexity signals required for reasoning constraints. Each generation stage thus receives an anchor representation calibrated to its specific task objective. The next subsection therefore turns to the two-stage generation pipeline and explains how anchor metadata are incorporated as explicit constraints at each generation stage.

3.4 Two-Stage Generation Pipeline

We introduced the two-stage generation pipeline in response to a recurring pattern observed in preliminary single-stage generation experiments: the model often entered interpretive language before numerical and temporal facts had been stably established, which made it difficult to maintain a clear boundary between observational description and structural interpretation within a single output step. In the context of financial time series charts, this boundary instability manifests in four types of failures: incorrect trend direction, time misalignment, numerical or magnitude errors, and the introduction of external-knowledge causal events not supported by the chart.

These patterns suggested that single-stage output lacked a mechanism for establishing a factual layer before proceeding to structural explanation. ChartNarrator therefore separates these two objectives into consecutive generation tasks operating under different control conditions: establishing what is visible in the series, and explaining how the observed structure develops.

Stage 1 takes the chart image and the Stage 1 anchor schema as input, and is responsible for constrained factual narration. In practice, this means that factual claims must agree with the extrema, start and end values, and magnitudes of change encoded in the anchors; descriptions must remain within the valid data window; and directional language must follow the semantic settings associated with the variable. Stage 1 is therefore limited to observable structure within the chart window, including the start point, end point, direction of change, and key turning points.

We intentionally suppress explanatory extensions at this stage; external events, entities, and causal language without anchor support do not enter the output. The resulting text provides the factual basis on which Stage 2 reasoning depends.

Evaluating whether a Stage 1 output satisfies factual grounding requirements demands semantic judgment that rule-based or lexical overlap cannot provide; we therefore employ LLM-as-a-Judge [9] as the quality gate. Within the generation pipeline, this module functions as a binary quality gate that screens out outputs failing to satisfy basic factual constraints before they are passed to Stage 2. The Judge configuration, evaluation dimensions, and reliability verification are described in Section 3.5.

Stage 2 produces source-bounded mechanistic reasoning based on retained Stage 1 outputs, aiming to maximize structural explanatory power within the bounds of source faithfulness. All interpretations must therefore be recoverable from chart shape and Stage 2 anchor metadata. The generation space consequently excludes external economic agents, external events, and background world knowledge not grounded in the source. At this stage, the model relies on both the retained Stage 1 factual layer and the Stage 2 anchor schema. The former provides the established observational basis, while the latter provides structural patterns, semantic direction and explanatory boundaries.

When the sequence has sufficient internal stage relationships, Stage 2 adopts a complete reasoning format, organizing the output through paragraph division: the first paragraph *P1* (*Observed Structure*) describes the observable patterns of the sequence, the second paragraph *P2* (*Mechanistic Interpretations*) proposes

structural interpretations under anchor constraints, and the third paragraph *P3* (*Dependency Resolution*) constructs explicit inter-stage dependencies, following the semantic hierarchy proposed by Lundgard and Satyanarayan [14]. The complete Stage 2 generation prompts, including the full paragraph-level instructions for each route, are provided in Appendix A.3.

The organization reflects the internal sequence of mechanistic reasoning, moving from observation to interpretation to explicit dependency resolution; it constitutes the full reasoning format of the complex route, while the simple route adopts a condensed two-paragraph organization, as described below.

In the final system, Stage 2 is route-conditioned by `route_label`, where the simple route is used for low-complexity structural interpretation, and the complex route is used for series with stronger internal phase structure. When an instance is routed as simple, the output organization shrinks to a two-paragraph format, retaining only *P1* and *P2*; when an instance is routed as complex, the output retains the full three-paragraph format, where *P3* provides the necessary expression space for dependency resolution.

A single-phase or low-complexity window cannot reliably support a complete dependency chain. Forcing such sequences to produce *P3* risks introducing phase dependencies that are not supported by the chart, a failure mode this framework terms mechanistic overreach. The shorter simple-route format therefore reflects the structural depth that the input can genuinely sustain. Within the complex route, the morphology family continues to act as a secondary control signal, adjusting the narrative focus and interpretation based on differences in `trend_drift`, `structural_swing`, and `oscillatory_or_seasonal_regime`.

The two stages form a controlled progression from factual grounding to source-bounded structural explanation. Stage 1 establishes a verifiable factual layer, and Stage 2 builds on that layer to produce chart-bounded mechanistic reasoning under route-conditioned control. Filtering after each stage constitutes generation-side quality control, ensuring that later training-set construction is based on outputs that already satisfy stage-specific faithfulness requirements. The next subsection turns to the evaluation protocol and explains how these outputs are judged, how their reliability is verified through inter-rater agreement, and how they are used for training-set selection and model comparison.

3.5 Evaluation Protocol

In ChartNarrator, the evaluation protocol serves as a unified judgment and verification framework, supporting both generation-stage quality screening and systematic comparison among final models. The core design principle is that automatic judgment must be validated for reliability before it governs sample selection, training set construction, or model comparison. The protocol therefore spans the full pipeline: it screens stage-wise outputs during generation and defines the reference frame for final model comparison. Reliability verification is accordingly treated not as an optional audit step, but as a prerequisite integrated into the experimental design.

For Stage 1, the judge evaluates outputs along three dimensions following the syntactic-semantic-pragmatic framework for data-to-text evaluation [26]: *Semantic Correctness*, *Syntactic Coverage*, and *Pragmatic Appropriateness*. The three dimensions of Stage 1 play different roles in the filtering process: Semantic Correctness is the only hard filtering threshold because factual errors constitute irreversible failures; once a sample is semantically unreliable, it loses its basis for entering subsequent stages. Syntactic Coverage and Pragmatic Appropriateness are not used as filtering conditions for different reasons. Syntactic Coverage reflects the degree of narrative completeness rather than a binary pass-or-fail condition, and therefore does not constitute a hard failure signal.

Pragmatic Appropriateness is not used as a filtering threshold because its score distribution in practice is closer to a binary signal than a continuous quality gradient, as confirmed by the results reported in Section 4.1.

These three dimensions are not carried forward to Stage 2 because mechanistic reasoning introduces a qualitatively different class of evaluation demands: not whether facts are correctly stated, but whether structural interpretations remain grounded within chart-bounded evidence, which requires a distinct set of dimensions centered on reasoning quality. The two sets of dimensions address qualitatively distinct generation tasks: Stage 1 dimensions assess basic linguistic faithfulness, while Stage 2 dimensions assess the quality of structural reasoning within chart-bounded evidence.

Stage 2 evaluation therefore adopts four faithfulness-oriented dimensions: *Mechanistic Grounding* (MG), *Dependency Strength* (DS), *Closed-System Compliance* (CSC), and *Structural Coherence* (SC). These

four dimensions were developed specifically for this framework, derived from the violation taxonomy established in RQ1; each dimension carries a traceable design motivation grounded in observed failure modes as discussed in Section 5.1. Table 3.1 defines each dimension and its weight in the weighted average score (WA).

Table 3.1: Stage 2 evaluation dimensions and weights. The weighted average score is WA = 0.35×MG + 0.35×DS + 0.20×CSC + 0.10×SC. Full scoring rubrics are provided in Appendix A.4.

Dimension	**Weight**	**Definition**
Mechanistic Grounding (MG)	0.35	Extent to which P2 mechanisms are grounded in chart structure rather than vague latent-state labels or domain inferences not visible in the chart.
Dependency Strength (DS)	0.35	Extent to which P3 resolves a structural dependency between the two trajectory stages, rather than restating P2.
Closed-System Compliance (CSC)	0.20	Degree to which the narrative remains within the chart boundary, introducing no external events, actors, or unplotted causal variables.
Structural Coherence (SC)	0.10	Degree to which each paragraph performs its designated structural function, with mechanism count consistent with the assigned route.

All evaluation dimensions in both stages share a common 1–5 integer scoring scale. Table 3.2 summarizes the intended interpretation at each score level; full per-dimension rubrics are provided in Appendix A.4.

Table 3.2: Scoring scale shared across all Stage 1 and Stage 2 evaluation dimensions. Each dimension is scored on a 1–5 integer scale by the LLM Judge. Full per-dimension rubrics are provided in Appendix A.4.

Score	**Stage 1 (general)**	**Stage 2 (general)**
5	No errors; all constraints fully satisfied	Fully chart-specific, falsifiable, structurally coherent
4	At most one minor error or violation	Mostly specific; minor generic phrasing or blur
3	One major error or 2–3 minor errors	Partially specific; some direction labels or uneven structure
2	One critical error or 2+ major errors	Mostly direction labels, latent-state language, or role confusion
1	Multiple critical errors or systematic misalignment	No mechanism; pure restatement or incoherent structure

The staged judge design keeps evaluation criteria aligned with the boundaries of each generation task: Stage 1 criteria cover factual narration, and Stage 2 criteria cover mechanistic reasoning. Within this framework, LLM-as-a-Judge is treated as a protocol component rather than a substitute for human judgment, and its outputs are used only after reliability verification [9].

To determine whether automatic judgments can be considered reliable signals, we incorporate inter-rater agreement verification into the evaluation protocol. We compare LLM-as-a-Judge outputs with human ratings dimension by dimension on a stratified validation sample of 80 complex-route narratives, drawn via purposive stratified sampling across eight strata defined by WA tier, `anchor_borrowing` flag, and `morphology_mismatch` flag; WA = 5.0 samples are capped at $n = 20$ to prevent ceiling-scoring entries from dominating the estimate, and simple-route samples are excluded owing to their structural DS ceiling and compressed SC scope.

Agreement is measured with linear-weighted Cohen's Kappa (κ_w) [12] per dimension; full results are reported in Section 4.4. Kappa verification serves to identify which evaluation dimensions show stronger alignment with human judgment, and which continue to exhibit systematic disagreement. Only judgments that pass this reliability check are used in subsequent selection and comparison procedures. Reliability verification is therefore not a post hoc addition to the results section, but an integral part of the

evaluation protocol itself.

Once judge configuration and reliability verification are in place, we construct a high-quality training set from pipeline outputs through layered filtering. This process follows the logic of the two-stage generation pipeline: Stage 1 outputs are first filtered by basic fact constraints, Stage 2 outputs are then subjected to faithfulness-oriented judging, and the reliability analysis determines how these judgments can be used in final sample selection. Training-set inclusion therefore depends on whether stage-specific faithfulness requirements have been satisfied, rather than on fluency or surface naturalness alone.

Training set construction is thus a direct extension of the evaluation protocol, linking judgment, verification, and selection into a single quality-control chain. We use the retained high-quality samples to fine-tune Qwen2-VL-7B-Instruct [27] and form the core experimental objects for subsequent comparative evaluation.

In the final model comparison phase, we compare the fine-tuned ChartNarrator model and baseline systems within the same unified faithfulness-oriented evaluation framework. This unified framework ensures that different types of models, despite differences in input formats and capability boundaries, maintain consistency in evaluation dimensions, judgment procedures, and validation logic. Baseline comparison serves both as a performance ranking and as an examination of how different model classes behave under a shared faithfulness criterion, especially with respect to factual grounding and source-bounded reasoning.

The evaluation protocol in ChartNarrator thus performs four linked functions: judgment, validation, selection, and comparison. It governs both the admission of stage-wise outputs into the training pipeline and defines the unified reference frame for the final model comparison.

The next chapter reports the empirical outcomes of this protocol in actual experiments, including judge reliability, filtering results, training-set statistics, and performance comparisons between the fine-tuned model and the baselines. The complete Stage 1 and Stage 2 judge prompts, including full scoring rubrics and constraint specifications, are provided in Appendix A.3 and Appendix A.4 respectively.

3.6 Generation and Judge Models

Stage 1 generation uses Gemini 2.0 Flash [28] as the generator. Gemini Flash series models are optimized for high-throughput concurrent inference at low latency, making them well-suited to the Stage 1 task profile: the output space is tightly bounded by anchor metadata, and the generation objective is factual coverage of chart-extractable values within a constrained narrative structure.

For Stage 2, generation upgrades to Gemini 2.5 Flash [29], which retains the throughput characteristics of the Flash series while offering substantially greater expressive capacity. This upgrade was motivated by an empirical ceiling observed during iterative development: under Gemini 2.0 Flash, Stage 2 outputs converged to a small set of fixed templates regardless of input variation, indicating that the model had reached its capacity boundary for open-ended mechanistic reasoning. Switching to Gemini 2.5 Flash eliminated this pattern. With generator configurations in place, the evaluation protocol governs how outputs are judged, verified, and selected.

Stage 1 LLM-as-a-Judge uses Gemini 2.5 Flash [29] whose high-throughput design is appropriate for the well-structured, dimension-bounded scoring task that Stage 1 evaluation constitutes. Stage 2 judgment upgrades to Gemini 2.5 Pro [29], a model designed for tasks requiring extended reasoning and fine-grained discrimination. This choice reflects the demands of the four-dimensional faithfulness rubric: assessing degrees of mechanistic grounding and structural dependency requires the Judge to maintain consistent calibration across subtle qualitative distinctions that lie beyond the reliable range of lighter models.

3.7 Fine-Tuning the Target Model

ChartNarrator uses Qwen2-VL-7B-Instruct [27] as the base model for fine-tuning. This model was chosen based on three considerations: the model natively supports joint image and text input without requiring additional modality alignment engineering; the 7B parameter size is feasible within the resource constraint of a single A100 GPU (40 GB); and the instruction-tuned version aligns with common practices for task-adaptive fine-tuning, providing a more stable starting point for subsequent specialized training.

The fine-tuning objective is to specialize the model's output style to produce three-paragraph narratives conforming to the source-bounded mechanistic reasoning format described in Section 3.4.

Considering computational resource constraints, we employ QLoRA (Quantized Low-Rank Adaptation) [13] for parameter-efficient fine-tuning rather than full fine-tuning. QLoRA loads the base model with 4-bit NF4 quantization while maintaining full-precision gradient updates in the adapter layer. This makes fine-tuning of a 7B parameter model possible on a single GPU, while retaining task adaptability similar to full fine-tuning [13]. Complete training hyperparameters are listed in Appendix A.5.

Training set construction proceeds in two filtering stages applied to the Stage 2 output pool. Simple-route pairs are excluded first, as their structural DS ceiling prevents them from producing informative training signal for complex-route generation. Pairs with `anchor_borrowing` = `True` are then excluded, as the deterministic DS cap on these pairs suppresses score variance. The resulting eligible pool of complex-route pairs is then filtered by the quality threshold described below.

The quality threshold for the training set was set to WA ≥ 4.5, while requiring all individual dimension scores to be greater than 2. Details regarding the training set partitioning and construction are provided in Section 4.1. Between the two candidate thresholds WA ≥ 4.5 and WA ≥ 5.0, the former was chosen because the benefit of moderately relaxing the threshold on sample diversity outweighs the benefit on quality purity from forced selection.

Quantitative verification of this judgment through ablation experiments is provided in Section 4.5. We refer to the fine-tuned model trained on the WA ≥ 4.5 subset as *VLM 4.5*; this abbreviation will be used to refer to this model in subsequent sections.

Chapter 4

Results

This chapter reports the experimental results across four research questions, beginning with dataset statistics and training set construction.

4.1 Dataset Statistics and Training Set Construction

This section reports the complete data flow from the initial generation to the final training set, providing a common data foundation for the experiments reported in Sections 4.2–4.5. Figure 4.1 presents the sample size and filtering logic at each stage; subsequent sections explain the morphology routing distribution, simple-route exclusion decisions, and training set construction in turn.

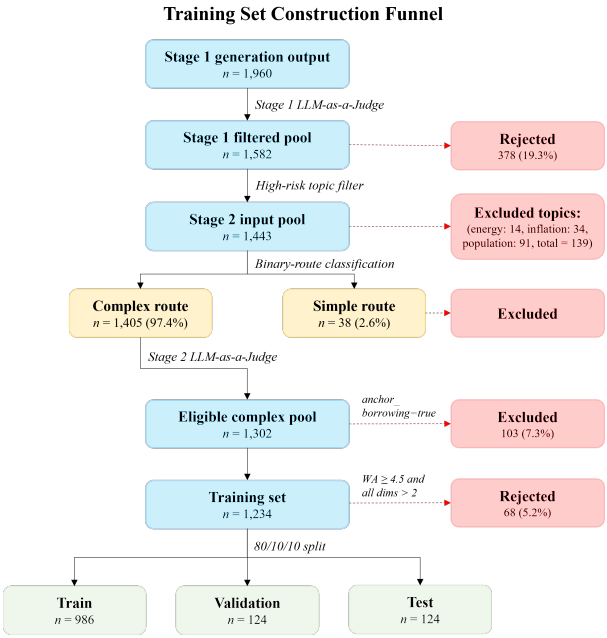


Figure 4.1: Training set construction funnel. Starting from 1,960 Stage 1 generation outputs, samples are filtered through four sequential stages: Stage 1 LLM-as-a-Judge filtering retains the semantic pool (n = 1,582); a high-risk topic filter removes indicators with ambiguous variable semantics (n = 1,443); binary-route classification separates complex-route (n = 1,405) from simple-route pairs (n = 38, 2.6%), with simple-route pairs excluded due to their structural DS ceiling. Following Stage 2 LLM-as-a-Judge scoring, `anchor_borrowing=True` pairs are excluded (n = 103), yielding the eligible complex pool (n = 1,302). A quality threshold of WA $\geq$ 4.5 with all individual dimension scores $>$ 2 produces the final training set (n = 1,234), partitioned into train (n = 986), validation (n = 124), and test (n = 124) subsets via stratified split (seed = 42).

Stage 1 generation produced 1,960 chart-narrative pairs. After Stage 1 LLM-as-a-Judge filtering, 1,582 samples that passed the Stage 1 Semantic Correctness threshold (Stage 1 filtered pool) were retained (retention rate 80.7%). Before entering Stage 2, we applied a high-risk topic filter to the semantic pool: the `energy` (14 pairs), `inflation` (34 pairs), and `population` (91 pairs) topic categories were excluded because the directional change of these indicators cannot be assigned a single consistent economic interpretation from the chart alone, making source-bounded narrative generation unreliable for this indicator class, totaling 139 pairs. After filtering, the Stage 2 input pool comprised 1,443 pairs, all of which entered Stage 2 generation and Judge evaluation.

Following the filtering procedure described in Section 3.7, we first excluded all 38 simple-route pairs and the 103 pairs with `anchor_borrowing` = `True`, the latter of which incur a deterministic DS cap (mean DS = 2.922). These two exclusions remove 141 pairs, leaving an eligible pool of 1,302 complex-route pairs. From this eligible pool, we retained pairs with WA ≥ 4.5 and all individual dimension scores > 2, yielding a final training set of 1,234 pairs.

The 1,443 pairs divide into two route categories: 1,405 complex-route pairs (97.4%) and 38 simple-route pairs (2.6%), as shown in Figure 4.2. The complex routes are further divided into three buckets based on their morphology family: `trend_drift` (191 routes), `structural_swing` (698 routes), and `oscillatory_or_seasonal` (554 routes). The remaining 38 simple-route pairs form a separate fourth category outside the morphology classification. This distribution is referenced in Sections 4.4 and 4.5.

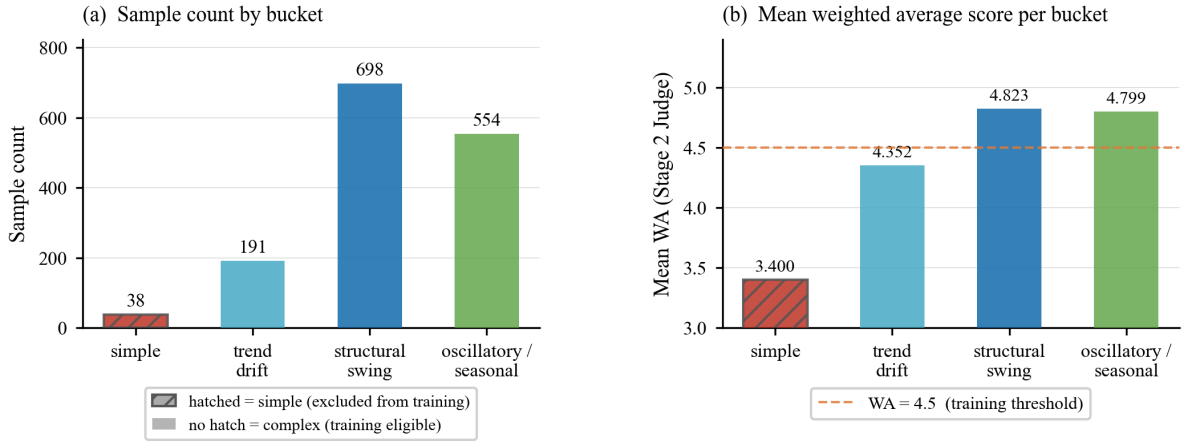


Figure 4.2: Morphology routing distribution of the Stage 2 input pool (n = 1,443). Panel (a) shows sample counts per bucket; panel (b) shows mean weighted average score per bucket as evaluated by the Stage 2 Judge, with the WA = 4.5 training threshold indicated by the dashed line. The hatched bar denotes the simple route, which is excluded from training set construction.

Following the Judge reliability study reported in Section 4.4, we excluded all 38 simple-route pairs from the training candidates. Simple routes correspond to single-phase time-series windows; under the three-paragraph generation framework, a deterministic cap forces the Dependency Strength score to ≤ 3, as single-phase windows cannot support multi-stage dependency. Simple-route pairs therefore lack the structural conditions required for high-quality dependency learning. Judge evaluation results corroborate this structural analysis (Figure 4.3): the mean WA for simple-route pairs is 3.400 against 4.788 for complex-route pairs, with mean Mechanistic Grounding of 2.789 and mean Dependency Strength of 3.000; only 10.5% of simple-route pairs reach the training tier. Including these pairs in the training set would offer limited learning value and risk contaminating the learning signal of complex-route pairs.

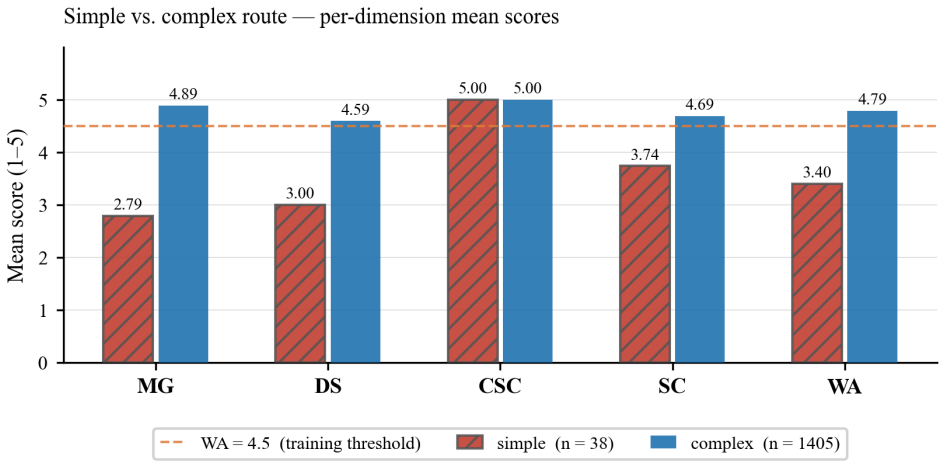


Figure 4.3: Per-dimension mean scores for simple-route ($n = 38$) and complex-route ($n = 1{,}405$) pairs as assigned by the Stage 2 Judge. The dashed line marks the WA = 4.5 training threshold. Simple-route pairs fall below the threshold on all dimensions except Closed-System Compliance (CSC), which is at ceiling for both routes.

The training set was partitioned into train (986 pairs), validation (124 pairs), and test (124 pairs) subsets using a fixed random seed of 42, with morphology proportions held consistent across all three splits. The human annotation subset used in Section 4.4 comprises 80 complex-route pairs drawn from the Stage 2 full output pool via purposive stratified sampling across eight strata, defined by WA tier, `anchor_borrowing` flag, and `morphology_mismatch` flag (Stratum A: $n = 20$; Strata B–D: $n = 10$ each; Stratum E: $n = 8$; Stratum F: $n = 3$; Stratum G: $n = 12$; Stratum H: $n = 7$). WA = 5.0 samples are capped at $n = 20$ (Stratum A) to prevent ceiling-scoring samples from dominating the Kappa estimate; simple-route samples are excluded because their structural DS ceiling (≤ 3) and compressed SC evaluation scope artificially inflate expected agreement. This subset serves as the basis for Judge reliability verification [12].

Table 4.1: Experimental conditions used in Sections 4.3 and 4.5. All fine-tuned VLM settings use Qwen2-VL-7B-Instruct as the base model, while Text-Only uses Qwen2.5-7B-Instruct. VLM 4.5 is the primary reference condition. n_{train} and n_{test} denote the numbers of training and test samples, respectively.

Condition	**Input modality**	**Training set**	n_{test}	**Purpose**
VLM 4.5	Image + anchor	WA $\geq$ 4.5 ($n_{\text{train}} = 986$)	124	Primary reference condition
VLM 5.0	Image + anchor	WA $\geq$ 5.0 ($n_{\text{train}} = 705$)	88	Training-quality ablation
No-Anchor	Image only	WA $\geq$ 4.5 ($n_{\text{train}} = 986$)	124	Anchor-injection ablation
Zero-Shot	Image + anchor	None (no fine-tuning)	124	Fine-tuning lower bound
Text-Only	CSV + anchor (no image)	WA $\geq$ 4.5 ($n_{\text{train}} = 986$)	124	Input-modality ablation

All experimental results in this chapter are reported under a unified evaluation framework, which adopts a Stage 2 LLM-as-a-Judge configuration and scores the generated narratives along four dimensions: Mechanistic Grounding (MG, weight 0.35), Dependency Strength (DS, weight 0.35), Closed-System Compliance (CSC, weight 0.20), Structural Coherence (SC, weight 0.10); the weighted average score is denoted WA. To systematically answer RQ2 and RQ4, we ran five experimental conditions; four conditions share the same test set ($n = 124$, complex-route, seed = 42). VLM 5.0 uses a separate test split derived from the WA $\geq$ 5.0 pool ($n = 88$), with the same 80/10/10 stratified partition.

Table 4.1 defines all five conditions: VLM 4.5 (main experiment, image + anchor metadata, WA $\geq$ 4.5 training set), VLM 5.0 (data quality threshold ablation, WA $\geq$ 5.0 training set), No-Anchor (anchor injection ablation, image input only, no anchor metadata), Zero-Shot (untuned Qwen2-VL-7B-Instruct), and Text-Only (input modality ablation, CSV time series data + anchor, no image). The four ablation comparisons are as follows: Zero-Shot vs. VLM 4.5 measures the contribution of fine-tuning; No-Anchor vs. VLM 4.5 isolates the effect of anchor metadata injection (RQ2); VLM 5.0 vs. VLM 4.5 examines the impact of training data quality thresholds; Text-Only vs. VLM 4.5 examines the role of visual input. The latter three comparisons, together with an overall performance summary, are reported in Section 4.5.

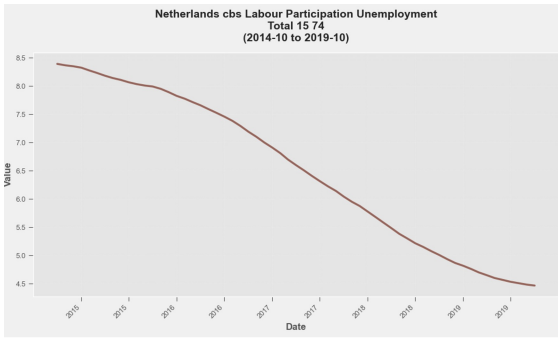
4.2 Faithfulness Violation Taxonomy

To answer RQ1, asking what types of faithfulness violations occur most frequently when MLLMs generate narratives from financial time series charts, and what underlying causes drive each violation type, we conducted a preliminary study using a single-stage unconstrained generation system that generated narratives directly from chart images without anchor metadata injection to identify recurring faithfulness violation patterns in MLLMs applied to financial time-series chart narration. Output were evaluated using the *Semantic Correctness*, *Syntactic Coverage*, and *Pragmatic Appropriateness* rubric described in Section 3.5; failure was concentrated in the semantic dimension, with a substantial proportion of outputs receiving low Semantic Correctness scores.

We manually inspected a small exploratory subset of low-scoring outputs, including an initial set of ten low-semantic-score cases and additional failure examples examined during prompt debugging. through inductive grouping of the observed failure patterns, we identified three main violation patterns (Trend Direction, Numerical Faithfulness, External Causal Hallucination); a fourth type (Latent-state Mechanism Wording) was subsequently identified through systematic analysis during Stage 2, forming a complete four-type taxonomy. Table 4.2 summarizes the taxonomy, and Figure 4.4 presents one representative example per type.

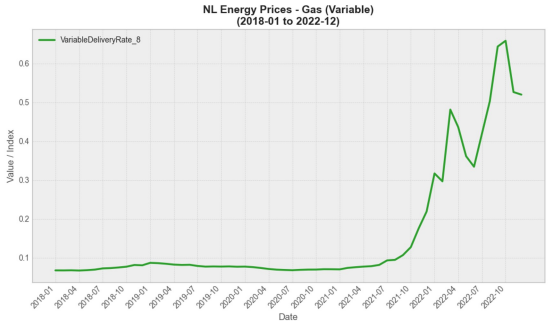
Table 4.2: Faithfulness violation taxonomy identified in ChartNarrator. The first three types were identified in the Preliminary Study; Type 4 was identified through systematic analysis during Stage 2.

Type	**Definition**	**Sub-types**
Trend Direction Error	Misreading the overall direction or internal reversal structure of a time-series curve; in financial contexts, also includes inverting the economic implications of a variable's directional change.	(a) Misjudging overall upward/-downward movement (b) Misinterpreting local reversals (c) Inverting variable directional semantics
Numerical Faithfulness Violation	Mismatches between model-generated numerical or temporal claims and the visual content of the chart.	(a) Peak/trough timestamp misalignment (b) Numerical magnitudes inconsistent with y-axis scale
External Causal Hallucination	Introducing macroeconomic events, policy changes, or named institutions absent from the chart as causal explanations; linguistically plausible but visually ungrounded.	(a) Introducing macroeconomic events (crises, policy shifts) (b) Introducing named institutions or implied economic agents
Latent-state Mechanism Wording	Using language that invokes unobservable internal states to explain trend changes, attributing visible structural changes to hidden conditions unverifiable from the chart image or anchor metadata.	(a) Energy-depletion framing (*exhaustion of upward momentum*) (b) Pressure-release framing (*release from an elevated state*) (c) Accumulation framing (*build-up of structural pressure*)



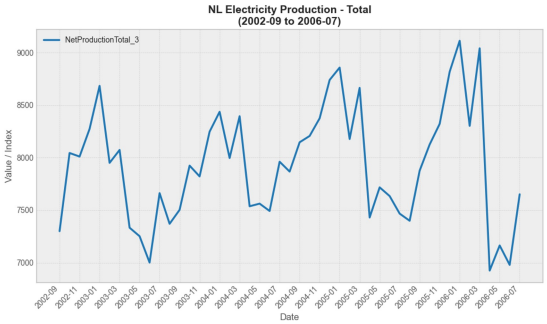
Output:
the proportion of the population actively participating in the labor force decreasing over time…

(a) Type 1: Trend Direction Error



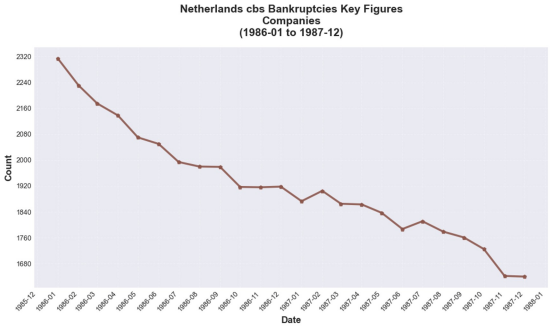
Output:
The price then experiences a significant drop starting in late 2021 and continuing into early 2022, before rising again in late 2022…

(b) Type 2: Numerical Faithfulness Violation



Output:
The initial increase ... was likely driven by the recovery from the dot-com bust and the subsequent economic expansion.

(c) Type 3: External Causal Hallucination



Output:
The decline... indicates a release from a previously elevated state of business failures.

(d) Type 4: Latent-state Mechanism Wording

Figure 4.4: Representative examples of the four faithfulness violation types identified in the ChartNarrator taxonomy. Highlighted spans (shown in orange) mark the violation-triggering phrases in each model output.

Trend Direction Errors refer to models misreading the overall direction or internal reversal structure of a time-series curve. Three subtypes are observed: misjudging overall upward or downward movement, misinterpreting local reversals and inverting the economic implications of a variable's direction which is unique to the financial domain. The proximate cause is that the model relies on visual perception to judge curve direction but lacks an explicit mechanism for binding directional change to the economic semantics of the specific variable. Figure 4.4a illustrates sub-type (c) on a Labour Participation Unemployment chart spanning 2014–2019: the series declines monotonically from 8.39 to 4.47, yet the model describes the labour force participation proportion as "decreasing over time", inverting the improvement signal carried by a falling unemployment rate.

Numerical Faithfulness Violations refer to mismatches between model-generated numerical and temporal claims and the visual content of the chart. Two subtypes are observed: peak or trough timestamp misalignment, where event times are shifted earlier or later than the visual evidence supports and occasionally fall outside the chart's time range entirely; and numerical magnitudes inconsistent with the y-axis scale, including fabricated or implausible value ranges. These errors reflect the inherent limitation of models performing precise numerical reads on continuous time-series data: without a numerical anchoring mechanism, the model resorts to visual approximation. Figure 4.4b shows a representative instance: the first local peak occurs in early 2022, yet the model reports a "significant drop starting in late 2021 and continuing into early 2022", a timestamp and direction mismatch relative to the visible chart structure.

External Causal Hallucinations refer to models introducing macroeconomic events, policy changes, or named institutions absent from the chart as causal explanations. Such statements are linguistically plausible but visually ungrounded. Figure 4.4c illustrates a representative instance on an NL Electricity Production chart (2002–2006): the model attributes the initial increase to "recovery from the dot-com bust and the subsequent economic expansion", neither of which appears in the chart. The proximate cause is the absence of a boundary between the model's world knowledge and the visual content: the model defaults to economic explanation templates from pre-training rather than grounding its account strictly in the chart.

Latent-state Mechanism Wording refers to models using language that invokes unobservable internal states to explain trend changes, for instance, "exhaustion of upward momentum", "build-up of structural pressure", or "pent-up demand", thereby attributing visible structural changes on the chart to hidden conditions that cannot be verified from the chart image itself or anchor metadata. Unlike Type 3, which introduces named external entities, Type 4 introduces internal, unobservable causal states; both extend beyond the chart's evidentiary boundaries, but through different pathways.

Figure 4.4d illustrates this pattern on a Netherlands CBS Bankruptcies chart (1986–1987): the model describes a sustained monotonic decline as indicating "a release from a previously elevated state of business failures", a hidden condition framing with no basis in the chart. The proximate cause is that models tend to rely on an energy-accumulation-and-release metaphor when narrating trend reversals; this language reads naturally in economic commentary but constitutes a faithfulness violation in any generation task where the chart is the sole evidentiary source.

The prevalence of this violation type was subsequently confirmed during an earlier round of manual annotation conducted as part of the iterative Judge development process. In that annotation round, approximately 62.5% of 80 complex-route outputs examined contained at least one latent-state wording flag in the Mechanistic Grounding annotation notes, indicating that this pattern is not an isolated edge case but a systematic tendency of the model when producing mechanistic interpretations of trend reversals. This finding informed the design of the final Stage 2 generation constraint and the Mechanistic Grounding rubric used in the evaluation protocol reported in Section 3.5.

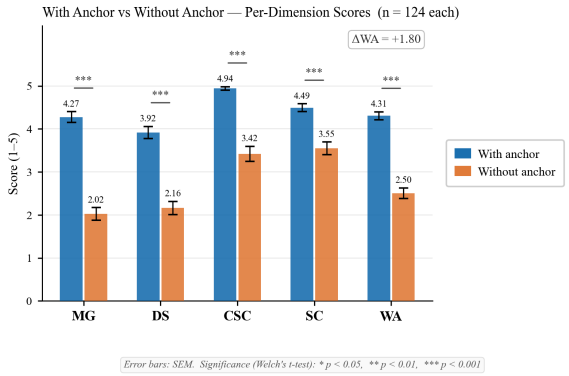
4.3 Effects of Anchor Metadata on Generation Quality

To answer RQ2, asking how structured anchor metadata affects the faithfulness and explanatory quality of generated narratives, we extracted the pair VLM 4.5 and No-Anchor from the five experimental conditions defined in Table 4.1 to measure the overall contribution of anchor metadata to the system. The No-Anchor condition removes anchor metadata simultaneously at both the training and inference levels: the `[CHART CONTEXT]` block is removed from all samples in the training set, and the model is trained with independent LoRA weights; similarly, no anchor metadata is injected during inference. Both conditions are evaluated on the same 124 complex-route test samples using the four-dimensional evaluation framework (MG, DS, CSC, SC) and weighted average score WA described in Section 3.5. The Judge retains access to anchor metadata when scoring to ensure that the evaluation criteria do not change due to variations in the generation conditions.

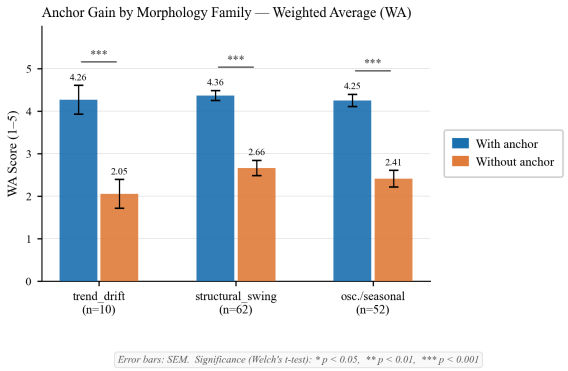
Table 4.3: Per-dimension scores under with-anchor and without-anchor conditions ($n = 124$ each). Δ denotes the mean gain attributable to anchor metadata injection. All differences are statistically significant (Welch's t-test, $p < 0.001$).

Condition	**MG**	**DS**	**CSC**	**SC**	**WA**
With anchor	4.274	3.919	4.944	4.492	4.306
Without anchor	2.024	2.161	3.419	3.548	2.504
Δ (gain)	+2.250	+1.758	+1.525	+0.944	+1.802

Table 4.3 presents the mean scores and gains across all dimensions for both conditions. The with-anchor condition achieves a mean WA of 4.306 against 2.504 for the without-anchor condition, ΔWA = +1.802. Welch's t-test confirms statistical significance across all five metrics ($p < 0.001$).



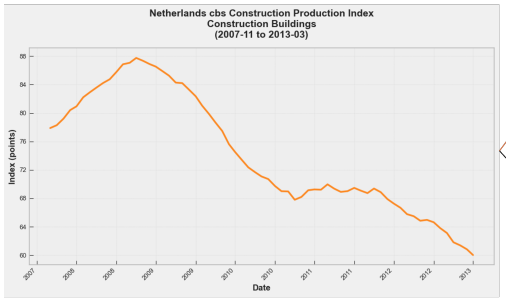
(a) Per-dimension scores under with-anchor and without-anchor conditions ($n = 124$ each). Error bars indicate SEM. All differences are statistically significant (Welch's t-test, $p < 0.001$).



(b) Weighted average scores by morphology family. All three families show statistically significant anchor gain (Welch's t-test, $p < 0.001$). Results for `trend_drift` should be interpreted with caution ($n = 10$).

Figure 4.5: Effects of anchor metadata injection on generation quality. The No-Anchor condition removes anchor metadata at both training and inference stages (joint ablation); the Judge retains access to anchor metadata in both conditions to ensure consistent evaluation.

Figure 4.5a shows that the absence of anchor metadata produces larger score reductions in MG and DS than in CSC and SC. Figure 4.5b breaks down the anchor gain by morphology family. All three families show positive and statistically significant anchor gain ($p < 0.001$). The `structural_swing` family shows the smallest gain ($\Delta = +1.70$, $n = 62$), followed by the `oscillatory_or_seasonal_regime` family ($\Delta = +1.84$, $n = 52$), and `trend_drift` shows the largest gain ($\Delta = +2.21$), though its small sample size ($n = 10$) warrants caution.



Narrator Output(Without Anchor):
The series began at its peak value of 305.875 on January 1, 2009, then increased to reach a local peak of approximately 346.5 by early 2010. Following this, the series reversed direction and declined sharply over several months, reaching a trough around 220 in late 2010 or early 2011. From this low point, the series exhibited a partial recovery, rising again to approximately 250 by mid-2011 before declining once more towards the end of the observation period.

Narrator Output(With Anchor):
The production volume index began at 77.8833 in November 2007 and rose to its highest point of 87.7417 on December 1, 2008. Following this peak, the index reversed direction and declined steadily over approximately two years, reaching a local trough of 69.0833 by October 1, 2011. From this point, the series continued its downward trajectory, ending the observation period at its lowest level of 60.025 on March 1, 2013.

Figure 4.6: Qualitative comparison of model outputs on the CBS Construction Production Index (Construction Buildings, 2007–2013) under with-anchor and without-anchor conditions. Without anchor metadata, the model fabricates a value range of approximately 220-347, rough four times the true range of 60–88. With anchor metadata, all cited values correspond to anchor constrained extrema and time boundaries. This pair constitutes a controlled contrast: the same fine-tuned model is used in both conditions, with anchor metadata removed at both training and inference stages in the No-Anchor condition.

To illustrate the effect of anchor injection on suppressing numerical drift at the output text level, Figure 4.6 shows a comparison of the first paragraph of the same index (CBS Construction Production

Index Construction Buildings, `structural_swing`, ΔWA = +3.70) under two generation conditions. Without anchor metadata, the model-generated narrative references a completely fictitious numerical system, starting at 305.875, with a local peak of approximately 346.5 and a trough of approximately 220, while the actual numerical range of the chart is approximately 60 to 88, a difference of about four times. With anchor metadata injected, the generated content directly references the extrema and time boundary fields in the anchor schema, and all values are within the observable range of the chart: a starting value of 77.88 (November 2007), a peak of 87.74 (December 2008), a local trough of 69.08 (October 2011), and a terminal value of 60.03 (March 2013).

4.4 Inter-Rater Reliability of LLM-as-a-Judge

This section reports addresses RQ3, asking to what extent LLM-as-a-Judge can reliably approximate human judgments across the faithfulness dimensions defined in this framework, and under what conditions systematic disagreement arises, by reporting the inter-rater reliability results between the LLM Judge and a human annotator across the four Stage 2 evaluation dimensions. Human annotation was conducted using the dedicated interface described in Section 3.5.

The validation sample comprises 80 complex-route narratives drawn from the Stage 2 full output pool via purposive stratified sampling across 8 strata, defined by WA tier, `anchor_borrowing` flag, and `morphology_mismatch` flag. Samples with WA = 5.0, which account for approximately 62.5% of the complex-route corpus, are capped at $n = 20$ (Stratum A) to prevent ceiling-scoring samples from dominating the Kappa estimate. Simple-route samples are excluded from the validation set because their structural DS ceiling (≤ 3) and compressed SC evaluation scope artificially inflate expected agreement.

The human annotator completed all four dimension scores independently before consulting the Judge output, a sequencing designed to reduce anchoring bias. To enforce this sequencing, we developed a dedicated web-based annotation interface.

The tool presents each narrative alongside its source chart, with the LLM Judge's dimension scores before the Judge output is revealed. The interface is additionally displays stratum metadata (`anchor_borrowing` flag, morphology family, WA tier) to support systematic coverage of all validation samples, and exports completed annotations as structured JSON for downstream Kappa computation.

Agreement is measured with linear-weighted Cohen's Kappa (κ_w) [12] per dimension; the comprehensive κ_w is computed as $MG \times 0.35 + DS \times 0.35 + SC \times 0.10$ (normalized to the retained weight sum of 0.80), with CSC excluded owing to score degeneracy.

Table 4.4: Inter-rater reliability between the LLM Judge and the human annotator across four Stage 2 evaluation dimensions on the validation subset ($n = 80$, complex-route only). Agreement is measured with linear-weighted Cohen's Kappa (κ_w) [12]. Int. denotes interpretation band; Adj. counts scores within one point as agreement. The weighted-average κ_w excludes CSC and is computed as MG × 0.35 + DS × 0.35 + SC × 0.10, normalised to the retained weight sum of 0.80. Interpretation bands follow Cohen [12]: Slight (< 0.20), Fair (0.20–0.40), Moderate (0.40–0.60), Substantial (0.60–0.80), and Almost Perfect (> 0.80).

Dimension	**LLM**	**Human**	κ_w	**Int.**	**Exact**	**Adj.**	**MAE**
MG	4.53	3.84	0.306	Fair	50.0	73.8	0.863
DS	3.99	3.62	0.659	Substantial	68.8	96.2	0.362
CSC	5.00	5.00	N/A[†]	Degenerate	100.0	100.0	0.000
SC	4.12	3.85	0.323	Fair	46.2	86.2	0.675
WA	—	—	0.463	Moderate	—	—	—

[†]CSC is excluded from Kappa computation because the Judge assigned score 5 to all 80 entries, yielding zero variance in one rater and rendering linear-weighted Kappa undefined. The human annotator independently confirmed score 5 for all 80 entries.

Table 4.4 reports per-dimension LLM and human mean scores, κ_w, interpretation band, exact agreement rate, adjacent agreement rate, and MAE.

Overall, the composite $\kappa_w = 0.463$ corresponds to Moderate agreement. Dependency Strength reached the highest agreement among the four dimensions ($\kappa_w = 0.659$, Substantial), while Mechanistic Grounding and Structural Coherence both fall in the Fair range ($\kappa_w = 0.306$ and 0.323 respectively). Closed-System Compliance is excluded from Kappa computation because the Judge assigned score 5 to all 80 entries, yielding zero variance in one rater and rendering linear-weighted Kappa undefined. As a substitute measure, the human annotator independently confirmed score 5 for all 80 entries, giving a confirmation rate of 100% and corroborating the Judge's assessment.

The Fair Kappa for Mechanistic Grounding is accompanied by a pronounced distributional asymmetry: the Judge assigned score 5 in 91.5% of cases, whereas the human annotator's scores were more widely distributed (score 5 in 40.0% of cases; full distribution: 2:13, 3:19, 4:16, 5:32). Adjacent agreement nonetheless reached 73.8% and MAE = 0.863, indicating that the disagreement reflects a systematic upward shift in Judge scoring, rather than random noise.

Stratifying DS by `anchor_borrowing` flag reveals a substantial difference between the two subgroups: entries flagged `anchor_borrowing=True` ($n = 25$) yield $\kappa_w = 0.609$ (Substantial), whereas non-flagged entries `anchor_borrowing=False` ($n = 55$) yield $\kappa_w = 0.386$ (Fair). Mean scores diverge in the same direction: the True subgroup shows lower means on both sides (LLM 2.84, Human 2.68, MAE =0.160), while the False subgroup shows higher means (LLM 4.51, Human 4.05, MAE =0.455). This pattern suggests that the Judge and the human annotator converge closely when identifying the DS reductions associated with anchor-borrowing cases, but agree less on DS judgments where anchor-borrowing is absent.

Table A.8 in the Appendix reports per-stratum κ_w for MG, DS, and SC across all eight strata.

Two boundary cases warrant attention. In Stratum A (*ceiling*, $n = 20$), $\kappa_w = 0.000$ across all three dimensions is an instance of the Kappa paradox: when both raters' score distributions are highly concentrated, linear-weighted Kappa approaches zero regardless of the observed agreement rate, and should be read as a statistical artefact rather than substantive disagreement. In Stratum F (*low*, $n = 3$) $\kappa_w = 1.000$ for DS and SC reflects the extremely small sample size and carries no interpretable reliability signal.

4.5 Comparative Model Performance

This section answers RQ4 through a system comparison under four conditions: VLM 4.5 is the primary model; No-Anchor targets the anchor metadata contribution; Text-Only targets the input modality; and Zero-Shot serves as an unsupervised baseline to measure the contribution of fine-tuning. No-Anchor uses joint ablation, whereby anchor metadata is removed simultaneously during both training and inference phases with independent LoRA weights. Therefore, it isolates the total contribution of anchor metadata to the system, rather than its localized effect at inference time. Text-Only replaces image input with structured CSV text, with the same training set size as VLM 4.5 ($n = 1{,}234$), and is used to measure the marginal contribution of visual input relative to structured numerical text. Zero-Shot uses a base model (without fine-tuning), with input consisting of a chart image and anchor metadata, serving as an unsupervised reference.

Table 4.5 presents the complete numerical data for the four conditions across five dimensions. Figure 4.7, panel (a) shows the total WA score, and panel (b) shows the per-dimension distribution. The overall ranking is: Text-Only (4.422) $\approx$ VLM 4.5 (4.306) $\gg$ No-Anchor (2.504) $>$ Zero-Shot (1.806); the difference between Text-Only and VLM 4.5 (ΔWA $= -0.117$) is not statistically significant and the two conditions are treated as equivalent.

The difference between VLM 4.5 and No-Anchor (ΔWA $= +1.802$, $p < 0.001$) is significant across all four dimensions, with the largest absolute differences in MG and DS (approximately 2.250 and 1.758, respectively), while the differences in CSC and SC are relatively smaller.

VLM 4.5 outperforms Zero-Shot across all dimensions (ΔWA $= +2.500$, $p < 0.001$), with a particularly low DS score (approximately 1.51), reflecting the base model's inability to produce cross-segment dependency reasoning.

Table 4.5: Per-dimension and weighted-average scores for the four experimental conditions evaluated on the held-out test set ($n = 124$, while Text-Only scores are averaged over three repeated evaluation runs on the same held-out set). FC denotes three-paragraph format compliance rate. Significance vs. VLM 4.5 (Primary) assessed via Welch's t-test: $^{***}p < 0.001$; n.s. = not significant.

Condition	**MG**	**DS**	**CSC**	**SC**	**WA**	σ	**FC (%)**
VLM 4.5 (Primary)	4.274	3.919	4.944	4.492	4.306	0.991	100.0
No-Anchor***	2.024	2.161	3.419	3.548	2.504	1.388	81.5
Zero-Shot***	1.863	1.508	2.202	1.829	1.806	0.669	0.0
Text-Only$^{\text{n.s.}}$	4.368	4.113	4.987	4.567	4.422	0.801	100.0

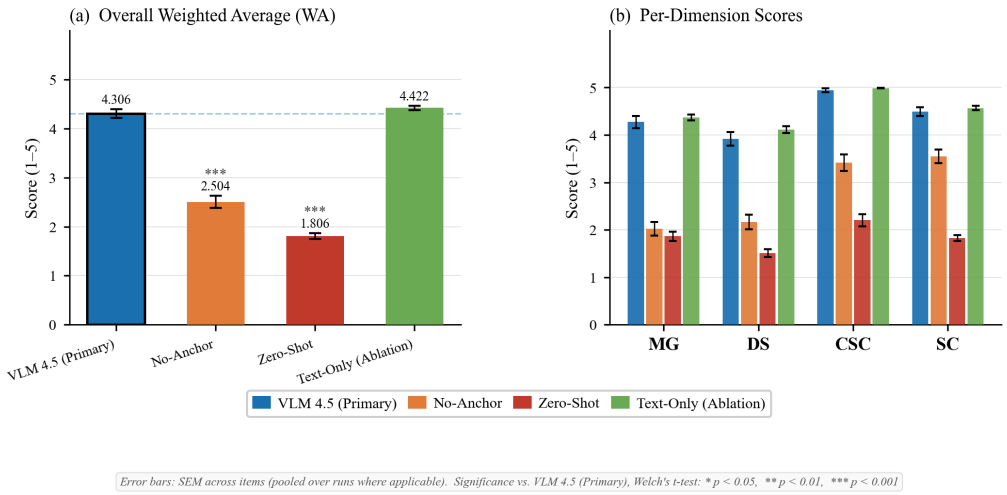


Figure 4.7: Overview of evaluation results across the four experimental conditions. Panel (a) shows the overall weighted average (WA) for each condition; asterisks denote significance vs. VLM 4.5 (Primary) assessed via Welch's t-test ($^{***}p < 0.001$). Panel (b) shows mean per-dimension scores across Mechanistic Grounding (MG), Dependency Strength (DS), Closed-System Compliance (CSC), and Structural Coherence (SC). Error bars represent SEM across items, pooled over evaluation runs where applicable.

Figure 4.8 presents pairwise per-dimension comparisons between VLM 4.5 and each of the three conditions: panel (a) vs. No-Anchor, panel (b) vs. Zero-Shot, and panel (c) vs. Text-Only. Text-Only and VLM 4.5 show no significant difference across any dimension (panel c), but Text-Only scores slightly higher than VLM 4.5 on DS and SC, while both are close to the ceiling on CSC (approximately 4.94 vs. 5.00). Regarding format compliance, all fine-tuned conditions (VLM 4.5 and Text-Only) have a 100% three-segment format compliance rate; No-Anchor is approximately 81.5%.

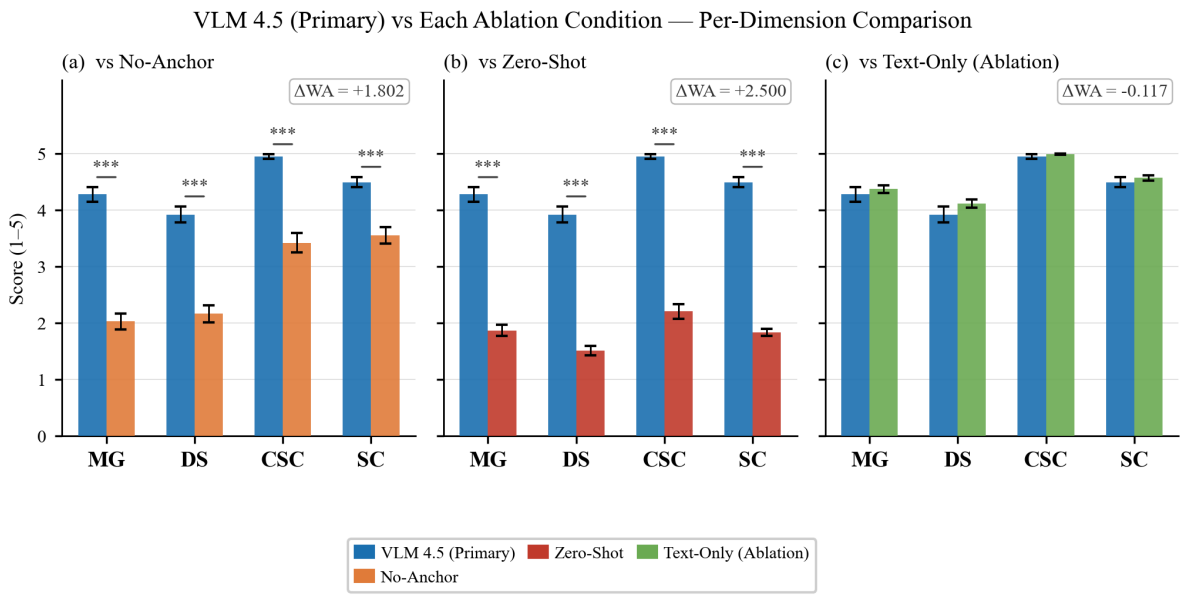


Figure 4.8: Per-dimension comparison of VLM 4.5 (Primary) against each ablation condition. Panel (a) shows VLM 4.5 vs. No-Anchor (ΔWA = +1.802), panel (b) shows VLM 4.5 vs. Zero-Shot (ΔWA = +2.500), and panel (c) shows VLM 4.5 vs. Text-Only (ΔWA = −0.117). Asterisks denote significance vs. VLM 4.5 (Primary) assessed via Welch's t-test ($^{***}p < 0.001$); no significance markers appear in panel (c) as no dimension reaches $p < 0.05$. Error bars represent SEM across items, pooled over evaluation runs where applicable.

To separately examine the effect of the training set quality threshold, we drew 881 samples from the WA $\geq$ 4.5 training pool, matched to the morphology distribution of VLM 5.0 with scores pooled over three evaluation runs. Figure 4.9 presents the results of this sub-ablation: the difference between VLM 4.5 (WA = 4.055) and VLM 5.0 (WA = 4.004) is (ΔWA = +0.051), which is statistically insignificant (n.s.).

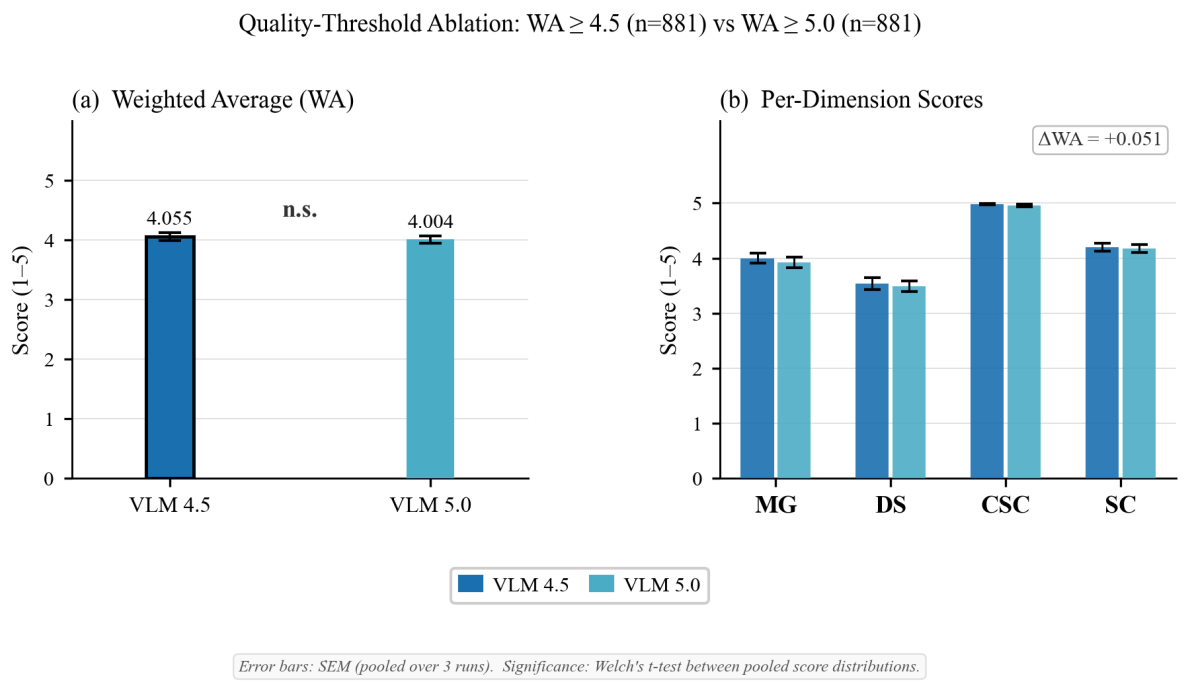


Figure 4.9: Quality-threshold sub-ablation: VLM 4.5 Matched ($n_{\text{train}} = 881$, sampled from the WA ≥ 4.5 pool to match the morphology distribution of VLM 5.0) vs. VLM 5.0 ($n_{\text{train}} = 881$, WA ≥ 5.0 threshold). Both conditions share an identical held-out test set and are evaluated over three pooled runs ($n = 264$ each). Panel (a) shows overall WA; the difference (ΔWA $= +0.051$) is not statistically significant (n.s., Welch's t-test). Panel (b) shows per-dimension scores; no dimension reaches significance. Error bars represent SEM pooled over three evaluation runs.

4.6 Qualitative Output Analysis

The aggregate scores reported in Section 4.5 characterize average performance across the test set, but they do not convey how the four conditions differ in the texture of their outputs. This section presents two representative examples drawn from the test set to illustrate the qualitative patterns that underlie the quantitative findings. For each example, truncated outputs from all four conditions are shown alongside brief annotations.

Example 1: Consumer Economic Climate Assessment (October 2007 – October 2012)

Figure 4.10 shows the chart for this example. The series exhibits a three-phase oscillatory pattern: an initial sharp decline from a positive range into deeply negative territory, a sustained recovery that approaches the zero line without establishing a positive range, and a renewed decline that reaches the overall trough near the end of the window.

This example is selected because it makes the aggregate findings in Section 4.5 visible at the output level: VLM 4.5 and Text-Only both recover the main three-phase structure and a dependency relation, whereas No-Anchor loses numerical grounding and Zero-Shot fails to produce the required dependency paragraph.

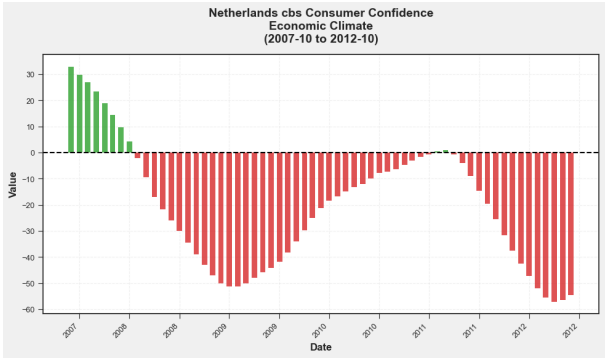


Figure 4.10: Consumer assessment of economic climate (points; positive = optimistic), October 2007 – October 2012. The series is classified as *oscillatory_or_seasonal_regime*, complex route.

Table 4.6 presents verbatim diagnostic excerpts from each condition. The excerpts are copied from the original JSON outputs and shortened only with omission markers. The annotations evaluate the full original output, not only the displayed excerpt.

Table 4.6: Verbatim diagnostic excerpts for Example 1 across the four conditions (oscillatory pattern, complex route). Excerpts are copied from the original JSON outputs and shortened only by omission markers [. . .]. Annotations evaluate the full original output and its corresponding held-out test score.

Condition	**Verbatim diagnostic excerpt from original output**	**Annotation based on full output**
VLM 4.5 WA = 5.0	*P1:* "The consumer assessment of economic climate began at its highest point of 32.9167 on 2007-10-01, then declined steadily over approximately two years to reach a local trough of −50.0833 on 2009-07-01. Following this, the series reversed direction and increased significantly, reaching a local peak of −0.6667 on 2011-08-01. [. . .]" *P2:* "The initial decline established a lower reference level for the series, shifting the overall range downwards from its starting positive value. This downward shift was interrupted by a strong reversal that initiated an upward swing, which temporarily restored the series to a higher negative band around −0.6667 before a renewed decline." *P3:* "The later decline represents a contained retracement, as the series did not fully recover the earlier local peak of −0.6667 reached on 2011-08-01. Instead, it remained bounded below this level [. . .]"	The full output captures the three-phase oscillatory structure and uses the 2011 local peak as an explicit ceiling anchor for the final decline. This supports the strong MG and DS performance of the primary VLM condition. One caveat is that the full P1 later describes the October 2012 endpoint as the lowest point, while the anchor metadata places the trough in August 2012.
Text-Only WA = 4.65	*P1:* "The series begins at its highest point of 32.9167 in October 2007, then declines sharply over approximately two years, crossing the zero threshold and reaching a local trough of −50.0833 by July 2009. Following this, the series reverses direction, rising significantly to reach a local peak of −0.6667 by August 2011. [. . .]" *P2:* "The initial sharp decline establishes an extended negative range for the series, with the zero threshold crossed and the lowest point reached at −57.1667 in August 2012. The subsequent rise from the local trough of −50.0833 on July 2009 represents a significant reversal that temporarily lifts the series above the zero level, establishing a new reference band for later movement." *P3:* "The later decline, beginning from the local peak of −0.6667 on August 2011, unfolds entirely within the negative range established earlier, failing to recover the positive territory seen initially. [. . .]"	The full output identifies the same broad decline–recovery–decline structure and forms a ceiling-style dependency in P3. This supports the aggregate result that Text-Only is close to VLM 4.5 when structured numerical text is available. Its lower score is consistent with grounding errors in P2, especially the claim that the recovery moved above zero, which contradicts the stated local peak of −0.6667.

Continued on next page

Condition	Verbatim diagnostic excerpt from original output	Annotation based on full output
No-Anchor WA = 1.1	*P1:* "The financial series began at 3054.7833 on January 2, 2020, and initially increased to a local peak of 3169.6667 by February 10, 2020. Following this, the value entered a sustained decline that persisted until reaching its lowest point within the window at 2816.6667 on April 28, 2020. [. . .]" *P2:* "The initial upward movement established an upper reference level at 3169.6667 by February 10, 2020, which was subsequently followed by a contained retracement. [. . .]" *P3:* "The later upward movement is structurally dependent on the preceding decline, as it unfolds entirely above the trough reached on April 28, 2020. [. . .]"	The full output describes a different chart, with a hallucinated 2020 time window and values around 2800 − −3200 rather than the 2007–2012 confidence series with values between approximately −57 and 33. This directly illustrates the aggregate drop of No-Anchor in MG and DS: without anchor metadata, the model preserves a surface P1–P2–P3 form but loses numerical and temporal grounding.
Zero-Shot WA = 1.85	*P1:* "The consumer assessment of economic climate experienced a complex pattern over the six-year period from October 2007 to October 2012. Initially, there was a sharp decline from an optimistic level of 32.9167 points in October 2007 to a low of −50.0833 points in July 2009. However, this trend reversed itself with a gradual increase that peaked at −0.6667 points in August 2011. [. . .]" *P2:* "Within the closed system defined by the variable's semantics, the consumer assessment of economic climate exhibited a series of swings and cycles. The initial phase saw a steep decline from October 2007 to July 2009, reflecting growing concerns about the economy. [. . .]"	The full output recognizes the broad three-phase shape, but it provides only two paragraphs and omits the P3 dependency paragraph required for the complex route. It also introduces graph-external domain language, as shown by the phrase "reflecting growing concerns about the economy". This illustrates the aggregate Zero-Shot pattern in Section 4.5: the base model can describe visible movement but does not reliably perform the fine-tuned dependency-resolution task.

Example 2: Household Consumption Expenditure on Services (March 2009 – February 2011)

Figure 4.11 shows a shorter structural-swing example. Unlike Example 1, this chart has a simpler two-phase shape: a sustained decline from the initial highest point to a mid-window trough, followed by a partial recovery that does not return to the initial level. VLM 4.5 and Text-Only both construct a source-bounded reversal and partial-recovery dependency, whereas No-Anchor loses the chart's numerical and temporal identity, and Zero-Shot replaces dependency reasoning with external economic explanation.

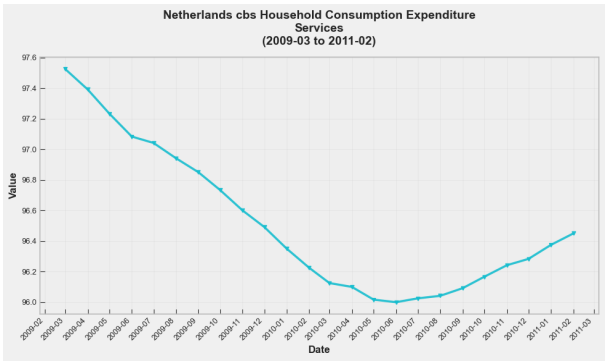


Figure 4.11: Household consumption expenditure on services (volume index, base year = 100), March 2009 – February 2011. The series is classified as *structural_swing*, complex route.

Table 4.7 follows the same format as Table 4.6.

Table 4.7: Verbatim diagnostic excerpts for Example 2 across the four conditions (structural swing, complex route).

Condition	Verbatim diagnostic excerpt from original output	Annotation based on full output
VLM 4.5 WA = 5.0	*P1:* "The series began at its peak of 97.525 in March 2009, then declined over two stages. It first decreased from 97.525 to reach a local trough of 96.025 by July 2010. Following this, the series reversed direction and increased, reaching an end value of 96.45 by February 2011. The overall movement is characterized by a contained decline followed by a partial recovery, with the trough at 96.025 serving as a turning point that initiated the upward trend." *P2:* "The initial decline establishes a lower reference level for subsequent movement, as the series moves from its starting peak of 97.525 down to a local trough of 96.025 by July 2010. This downward movement then transitions into a distinct upward phase, which unfolds from the established trough, indicating a structural reversal from the earlier declining path." *P3:* "The later increase, beginning from the local trough of 96.025, represents a partial restoration of levels observed prior to the initial decline. While the series recovers some ground, it remains below its starting value of 97.525, demonstrating that the earlier peak was not fully restored within the observation period."	The full output cleanly identifies the decline–trough–partial-recovery structure. P2 frames the trough as a structural reversal point, and P3 turns that reversal into a dependency relation by showing that the recovery remains below the initial peak. This directly supports the strong MG and DS performance of the primary VLM condition.
Text-Only WA = 4.65	*P1:* "The series began at its peak of 97.525 on 2009-03-01, then declined consistently over approximately 15 months until reaching a trough of 96.0 on 2010-06-01. Following this low point, the series reversed direction and increased steadily for about 11 months, concluding the observation period at 96.45 on 2011-02-01." *P2:* "The initial decline established a lower reference level, with the series falling from its starting peak of 97.525 to reach a trough of 96.0 by mid-2010. This downward movement was followed by a clear structural reversal, where the series shifted into an upward trajectory that persisted throughout the remainder of the observation period." *P3:* "The subsequent rise is interpreted as a recovery from the earlier trough, as it begins immediately after the lowest point reached during the first phase of decline. The later increase remains contained below the initial peak of 97.525, indicating that while the series recovered significantly, it did not fully restore its prior elevated levels."	The full output is close to VLM 4.5 in structure: it captures the two-phase swing, identifies the trough as the reversal point, and uses the initial peak as a ceiling for the partial recovery. This supports the aggregate result that Text-Only is statistically equivalent to VLM 4.5 when structured numerical text is available. Its slightly lower score reflects that P2 offers limited additional interpretation beyond restating values already introduced in P1.
No-Anchor WA = 1.0	*P1:* "The series began at 3049.675 on 2000-08-01, initiating from a local peak of 3050.25. It then declined to reach its lowest point within the observation period at 2999.0 by 2005-08-01, marking a significant structural reversal and establishing a new lower reference level." *P2:* "Following this trough, the series entered an extended upward trajectory that persisted until approximately mid-2008, where it reached a local peak around 3030. This subsequent movement represents a contained retracement rather than a full restoration of the initial starting value, as the series did not recover its earlier high." *P3:* "From this local peak around 3030 in mid-2008, the series reversed direction again, declining sharply over several months to establish a new floor near 2999.0 by early 2009. [. . .]"	The full output describes a different chart: it hallucinates a 2000–2010 time window, values around 3000, and a three-stage movement that does not match the 2009–2011 services index. This illustrates the No-Anchor aggregate pattern in Section 4.5: the model can preserve a narrative form, but without anchor metadata it loses numerical scale, temporal alignment, and source-grounded dependency.

Continued on next page

Condition	Verbatim diagnostic excerpt from original output	Annotation based on full output
Zero-Shot WA = 1.3	*P1:* "The chart depicts the household consumption expenditure on services over a two-year period from March 2009 to February 2011. Initially, there was a decline in consumption from March 2009 to July 2010, with a peak of 97.525 and a trough of 96.0. However, starting in July 2010, there was an upward trend until February 2011, reaching a new peak of 96.45. The inflection point occurred at July 2010, marking the transition between these two phases." *P2:* "Within this time frame, the overall trend is non-monotonic, characterized by two distinct phases. The first phase shows a gradual decrease in consumption expenditures from March 2009 to July 2010, reflecting a period of economic uncertainty or reduced spending capacity. This phase can be attributed to factors such as the global financial crisis that began around this time. [. . .]"	The full output identifies the broad two-phase shape, but it provides only two paragraphs and omits the P3 dependency paragraph required for the complex route. It also introduces graph-external causal claims, including "economic uncertainty", "reduced spending capacity", and "the global financial crisis". This supports the aggregate Zero-Shot finding: the base model can describe visible movement, but without fine tuning it does not reliably perform the required source-bounded dependency-resolution task.

Across both examples, the qualitative patterns are consistent with the aggregate scores reported in Section 4.5. VLM 4.5 and Text-Only both recover the main structural phases and construct source-bounded dependency relations in P3, which supports their comparable WA scores.

No-Anchor preserves the surface narrative form in both cases but loses numerical and temporal grounding, producing outputs that describe a different chart entirely. Zero-Shot identifies visible movement in both examples but omits the P3 dependency paragraph and introduces graph-external causal language, consistent with its low DS score in the aggregate evaluation. The two examples span different morphology families and window lengths suggesting that these patterns are not artefacts of a particular chart type.

Chapter 5

Discussion

This chapter discusses the findings across the four research questions and situates them within the broader literature.

5.1 Causes and Implications of the Violation Taxonomy

The four types of violations can be categorized into two causal chains based on their failure mechanisms, rather than four independent phenomena. This structural grouping is the core argument of this section.

The first chain is perceptual accuracy failure, encompassing Trend Direction Errors and Numerical Faithfulness Violations. Both arise from the model's reliance on visual approximations rather than precise extraction when reading continuous time series curves. Within this chain, subclasses (a) and (b) of Trend Direction Errors are straightforwardly perceptual: the model misreads the visual signal itself. Subclass (c), however, represents a distinct failure mode: the model correctly perceives the curve direction but cannot map it to the corresponding economic meaning, a step that requires domain-specific knowledge unavailable from the chart alone.

This subtype is a failure dimension that general chart benchmarks (such as ChartQA and CharXiv [2, 3]) cannot cover because these benchmarks aim for general visual question answering, not domain-grounded semantic interpretation.

The second chain is language prior dominance, encompassing External Causal Hallucinations and Latent-state Mechanism Wording. In the absence of explicit constraints, the model faces implicit task ambiguity: faithfulness to the chart or "sounding like a financial analyst". The pre-training corpus is rich in macro-event references and energy metaphor idioms; without closed-system constraints, these patterns are not actively invoked but appear in the output as an inertial extension of linguistic habits.

Identifying these two causal chains has direct significance in framework design: Anchor metadata addresses the first chain by encoding chart-extractable values as explicit numerical constraints, bypassing the visual approximation that underlies both violation types; the closed-system constraint of source-bounded mechanistic reasoning addresses the second by explicitly forbidding the language patterns that language prior dominance introduces. The two interventions are discussed in Section 5.2.

The hallucination taxonomy proposed by Ji et al. [4] and Maynez et al. [6] distinguishes two violation classes: intrinsic violations, which contradict the source material, and extrinsic violations, which introduce information absent from it. The first three types of violations can be mapped within the framework: Trend Direction Errors and Numerical Faithfulness Violations are intrinsic violations, while External Causal Hallucinations are extrinsic violations.

However, Latent-state Mechanism Wording cannot be captured by this dichotomous framework, which is the core argument of this section. It is not intrinsic violation: "exhaustion of upward momentum" does not directly contradict what is visible on the chart; the curve is indeed declining, therefore it does not meet the criterion of "contradicting the source material". It is also not a standard extrinsic violation. Extrinsic introduces independently verifiable external facts (such as a historical event), while latent-state wording introduces an unfalsifiable internal state, which is neither in the chart nor independently

verifiable external knowledge fact.

The epistemological problem with Latent-state Mechanism Wording is that phrases like "exhaustion of upward momentum" or "release from the prior elevated state" can explain any trend reversal; they lack falsifiability but give the narrative a superficial explanatory power. The high incidence of latent-state wording observed during iterative human annotation confirms that this violation type is structurally linked to the mechanistic task itself.

Ji et al.'s [4] framework uses "source relation" (whether the information comes from source material) as its criterion, a dimension that fails to capture issues at the level of "falsifiability". We therefore argue that the taxonomy we present identifies a third type of violation beyond the intrinsic/extrinsic dichotomy. Its distinguishing feature is not "making a mistake" or "introducing external information," but rather "using an unfalsifiable explanatory idiom." This type of violation is particularly prominent in the context of financial chart narratives because financial and economic language is inherently rich in such energy metaphorical idioms, and the model's pre-training exposure in this area is relatively high.

During the dataset construction phase, topics such as energy, inflation, and population were filtered out entirely due to semantic ambiguity of variables or polysemy of indicators. Combined with the findings of External Causal Hallucinations, this filtering decision has deeper theoretical implications than data cleaning.

For these indicators, the failure is definitional, not a generation failure the pipeline can correct: the semantics of the variable inherently require off-chart knowledge, so source-bounded constraints cannot fully cover the narrative needs. High-risk topic filtering therefore explicitly delineates the applicability boundaries of the ChartNarrator framework: it applies to indicators whose variable semantics are self-consistent within chart boundaries, not to indicators whose narrative needs inherently exceed the chart's evidence space.

5.2 Mechanism and Scope of Anchor Metadata Effects

Among the four evaluation dimensions, Mechanistic Grounding shows the largest anchor gain, a result that most directly reflects the core intent of the anchor design: Stage 2 requires the model to produce both numerically accurate claims and mechanistic inference, and the anchor provides the prerequisite for both.

The anchor's core function is to supply the raw materials that reasoning requires: without accurate numerical input, the model cannot deploy its existing reasoning capacity. As shown in Figure 4.6 (Section 4.3), the No-Anchor condition produces a numerical range approximately four times the true range; this is not an error in inference logic but a reading failure caused by distortion of the input signal.

Therefore, the MG contribution of the anchor can be precisely described as a "numerical glasses" effect: the model's mechanistic inference ability exists under both conditions, but this ability can only be reliably activated when the anchor provides an accurate numerical foundation.

Beyond numerical precision, the anchor fields `direction_note` and `higher_means` address a category of error that numerical grounding alone cannot resolve: Trend Direction Error Type (c), identified in Section 4.2, in which the model correctly perceives curve direction but maps it to the wrong economic meaning (e.g., a decrease in unemployment represents labour market improvement, not deterioration). This type of semantic misinterpretation is not specifically detected in general chart benchmarks (ChartQA [2] and CharXiv [3]) because general charts do not involve the counterintuitive domain semantic relationship of "decline = improvement".

The directional semantic field of an anchor therefore represents a contribution dimension that general chart grounding methods cannot cover: the anchor answers not only "what the value is" but also "what this change in value means." The necessity of this dimension in the financial domain is structural, not marginal: any time-series chart involving inverse indicators—such as unemployment rates or debt ratios—requires this type of explicit semantic constraint.

CSC scores near the ceiling (4.94) under with-anchor conditions, but this result cannot be solely attributed to anchor metadata. The pipeline explicitly lists constraints prohibiting external entity inference in both Stage 1 and Stage 2 generation hints, and the Judge further penalizes CSC violations with low scores; this systematic pipeline design, rather than any single anchor field, drives the near-ceiling result.

The 'closed_system_note' field in the anchor does participate in this mechanism, but its role is as an information transmission component of the pipeline: it transmits the chart's evidence boundaries to the generation end; it does not independently ground the output.

Mistakenly attributing the high CSC score to a single anchor field would mislead the reader into believing that simply injecting the anchor resolves boundary adherence; in practice, that property requires coordinated constraint enforcement across the full pipeline.

The drop in format compliance from 100% to 81.5% under the No-Anchor condition is the most unexpected finding of the ablation. Fine-tuning should have already incorporated the three-part structure into the model weights, but even after removing the anchor block during inference, 18.5% of samples still failed to produce the correct three-paragraph format. This result indicates that the `[CHART CONTEXT]` anchor block is learned during fine-tuning as a format conditioning trigger alongside its role as a content constraint.

This functionality is emergent: the training objective never explicitly required the model to produce a three-paragraph structure only upon seeing an anchor block, but the model establishes this conditional association during joint training. The three conditions form an interpretable learning trajectory: 0% for Zero-Shot indicates the model has no idea what a three-paragraph structure is; 81.5% for No-Anchor fine-tuned shows that fine-tuning imparts structural knowledge, but activation may fail in the absence of a trigger signal; and 100% for with-anchor indicates that the presence of the anchor block stably binds the three-segment generation behavior.

This finding has general implications for generation systems that rely on fine-tuning: structural prompt components can be learned during fine-tuning as conditional signals that activate specific generation behaviours; removing them at inference time may therefore disrupt format consistency to a degree that training alone does not prevent.

ChartNarrator's anchor metadata follows the same foundational principle as the grounding constraints in ToTTo [20]: both systems constrain generation by making the boundaries of relevant evidence explicit, delegating to the model only the task of generating within those boundaries.

However, the two constraints cover different semantic dimensions: ToTTo's highlighted cell constraints answer "which data to use"; ChartNarrator's anchors additionally answer "what is the directional semantic of this variable" and "where are the boundaries of the inference space." These two additional dimensions are unnecessary in table tasks. Table cells themselves do not carry domain semantic direction, and table tasks impose no closed-system boundary analogous to ChartNarrator's constraint against introducing uncharted causal factors.

Therefore, the incremental contribution of anchor metadata relative to general grounding methods is quantifiable: it extends the function of grounding from data selection to semantic interpretation and inference boundary propagation—three functional layers that together define the domain-specific design contribution of the anchor schema.

5.3 Interpreting Judge Reliability Across Evaluation Dimensions

Section 4.4 reports Kappa results across four dimensions, yet the numerical pattern alone does not resolve the question at the centre of RQ3: under what conditions can the Judge be trusted, and where does systematic disagreement emerge? This section address that question by examining the causes underlying each dimension's result.

Dependency Strength Among the four dimensions, Dependency Strength achieved the highest inter-rater agreement, and the significance of this result extends beyond scoring reliability alone: it constitutes evidence that the dependency_labels framework functions as a valid diagnostic structure.

When paragraph P3 does not introduce new structural facts, the `anchor_borrowing=True` subgroup yields Substantial agreement between the Judge and the human annotator, indicating that the Gate A/B/C/D logic is consistently recognizable to the rater. In the `anchor_borrowing=False` subgroup, where P3 does introduce new structural facts, agreement drops to Fair and the Judge scores systematically higher than the human annotator, with the mean score gap widening considerably.

This split reveals an internal boundary in DS evaluation: the Judge reliably identifies clear structural failures, where P3 contributes no new content, but agreement weakens when the task requires assessing degrees of success among passages that do introduce new material.

The dependency_labels framework thus provides stronger scaffolding for failure diagnosis than for success grading, an asymmetry with direct implications for how the framework should be applied in training data filtering.

Mechanistic Grounding The Fair agreement for Mechanistic Grounding does not reflect random scoring disagreement; the disagreement pattern is directional and consistent, pointing to a systematic calibration gap between the Judge and the human annotator.

As Table 4.4 reports, the Judge assigned score 5 in a large majority of MG cases, whereas the human annotator's scores were more widely distributed. The high adjacent agreement rate confirms that the rater is not in substantive disagreement about quality ordering; the Judge's threshold for acceptable mechanism descriptions is simply more permissive.

Because the Stage 2 generation pipeline filtering latent wording at the prompt level, samples entering the Kappa evaluation contain few outright Mechanistic Grounding violations. MG disagreements consequently arise in the range between acceptable and high-quality descriptions, rather than at the boundary between pass and fail.

This calibration gap has a direct consequence for training set construction because the training set was filtered using the Stage 2 Judge's MG threshold but not a human-annotated one, the admitted samples may include outputs that a human evaluator would rate as insufficient grounded. Since the Judge's bias is directional and consistent: it is systematically more permissive than a human-annotated standard.

Its effect on training set quality is therefore bounded and describable: the effective MG lower bound of the training set reflects the Judge's standard, which sits approximately 0.69 scores points above the human annotator's mean. Raising the MG admission threshold in future iterations to compensate for this offset would bring the effective human-annotated quality floor closer to the intended standard.

Closed-System Compliance The degenerate result for Closed-System Compliance, where the rater assigned score 5 to every entry, is better read as indirect evidence that the generation pipeline succeeded in filtering non-compliant outputs; it does not reflect a limitation of the evaluation design.

Both Stage 1 and Stage 2 generation constraints include explicit prohibitions on external domain inferences and entity introductions, and the Stage 2 Judge applies a penalizing low score to CSC violations. Together, these filters remove non-compliant samples before they reach the Kappa evaluation pool. When every sample in the validation set has already cleared these constraints, the absence of disagreement on CSC reflects the pipeline operating as intended.

The four-dimension reliability profile positions LLM-as-a-Judge as a dimension-aware filtering instrument within ChartNarrator with trust boundaries that differ across dimensions. When a dimension is governed by explicit structural decision rules, as in DS with its dependency_labels framework, the Judge achieves filtering precision consistent with human judgment. Where the task requires fine-grained specificity assessment, as in MG and SC, reliable use of the Judge depends on first characterizing the direction and magnitude of its systematic bias.

This asymmetry defines an operational scope for the Judge; it does not undermine the Judge's utility. Within that scope, automated filtering can reliably substitute for manual annotation at scale; judgments that fall outside it require threshold calibration or supplementary human review.

From a research design perspective, the Kappa validation contributes not just a reliability certificate but a boundary map: it identifies the specific conditions under which the Judge's outputs should be used with caution, information that is directly transferable to any subsequent study adopting the same evaluation framework.

We noted that this reliability analysis rests on a comparison between a single annotator and a Judge drawn from the same model family; the generalizability of these findings is therefore constrained by same-model-family bias and the single-annotator design, both of which are detailed in Sections 5.6.

5.4 Modality, Fine-tuning, and Training Set Effects

A complete answer to RQ4 requires reading all three comparison lines (Zero-shot, No-Anchor and Text-Only) together; no single pairwise difference is sufficient to attribute performance to its source.

Contribution of Fine-tuning and Anchor Metadata The improvement from Zero-Shot to No Anchor quantifies the contribution of fine-tuning itself: the model progressed from zero format compliance under Zero-Shot conditions to an 81.5% compliance rate, acquiring the three-part generation paradigm that serves as a prerequisite for all subsequent performance gains. However, the improvement from No-Anchor to VLM 4.5 is the largest single gain among the three lines. As Section 3.3 establishes, the gain attributable to anchor metadata injection (ΔWA = +1.802, No-Anchor to VLM 4.5) is more than twice the gain from fine-tuning alone (ΔWA = +0.698, Zero-Shot to No-Anchor), indicating that anchor metadata, rather than fine-tuning, is the dominant factor determining final output quality.

Visual Modality and Text-Only Equivalence The near-equivalence between VLM 4.5 and Text-Only, however, resists a simple explanation and is addressed in the following paragraph. Anchor metadata drives the majority of the performance gain; fine-tuning is a necessary but secondary prerequisite, and the visual modality yields no measurable advantage under the current configuration.

The near-equivalence of VLM 4.5 and Text-Only admits two complementary explanations, neither of which is sufficient on its own.

First, anchor metadata already encodes the information most critical to the narrative task — temporal boundaries, extrema values, and directional semantics — directly into the text input, leaving limited marginal value for an additional visual perception channel.

Secondly, precise numerical extraction from dense time-series charts remains a documented weakness of models at this parameter scale [3], and the clean tabular sequences available to the Text-Only condition may therefore carry higher effective information quality than the visual approximations produced by the VLM. Therefore, this result reflects the current scale of VLMs failing to fully utilize the fine-grained information provided by the image channel; it does not indicate that visual input carries no value. The two mechanisms jointly explain why the visual channel contributes no measurable gain under the current experimental configuration.

This result can be analyzed from a different angle. Anchor metadata was explicitly designed to encode the information most critical to the narrative task, including numerical extrema, temporal boundaries, and directional semantics as structured text fields. If anchor injection succeeds in this design objective, the residual informational value of the visual channel should be small by construction: the chart's most narratively relevant content has already been transferred into text input.

The near-equivalence of Text-Only and VLM 4.5 therefore constitutes indirect validation of the anchor schema's completeness as an information transfer mechanism. The practical implication is that a deployment scenario requiring high faithfulness but lacking reliable visual processing can substitute structured CSV input with anchor metadata and expect equivalent narrative quality, at lower computational cost and with substantially reduced catastrophic failure rate (1.1% vs. 6.5%)

However, this equivalence holds only under high CSV serialization quality: in out preliminary study, we observed cases in which timestamp columns were misinterpreted as numerical values, indicating that the Text-Only condition carries sensitivity to serialization format choices that the visual modality does not share. Anchor metadata constrains what the model may claim, but it does not repair upstream serialization errors in the tabular input itself.

Training Set Size versus Quality Threshold Beyond the modality question, the training set experiments reveal a further nuance regarding quality thresholds. The raw gap between Primary VLM 4.5 and VLM 5.0 (Table 4.5) appears to support the interpretation that training set diversity matters more than filtering stringency, but this comparison simultaneously varies both training set size (1,234 vs. 881 samples) and quality threshold, precluding attribution to either variable in isolation.

The controlled experiment described in Section 3.7 reduces this gap to statistical insignificance once training set size is held constant, identifying sample quantity rather than quality threshold as the primary driver of the original difference.

At the current scale, increasing the number of training samples has a greater impact on model performance than increasing the quality threshold, a finding consistent with observations in the instruction tuning literature that sample quantity can outweigh filtering stringency at limited data scales [30]. However, the extrapolation of this conclusion is limited: we note that this conclusion rests on a test set of only 88 samples, carries limited statistical power, and has been validated only at the 7B parameter scale. Whether it holds true for different model sizes or different data volume ranges is currently uncertain.

Output Robustness and Catastrophic Failure Rate Mean performance scores, however, do not fully characterize the differences between conditions; the catastrophic failure rate reveals a robustness dimension that the mean WA obscures.

Although the mean WA scores of VLM 4.5 and Text-Only are comparable, as Section 3.7 reports, VLM 4.5's catastrophic failure rate (6.5%) is nearly six times that of Text-Only (1.1%), and its output distribution is considerably wider, indicating that structured text input produces more stable outputs. This difference in stability is not visible at the mean level, but for deployment scenarios where minimizing catastrophic errors is a priority, it forms the practical basis for choosing Text-Only over VLM 4.5.

This constitutes an additional practical insight beyond the main conclusions of RQ4: mean performance equivalence does not imply deployment equivalence, and output robustness warrants explicit consideration alongside average faithfulness scores.

5.5 Framework-Level Observations

Three observations emerge from reading the four preceding sections together; each depends on evidence from multiple research questions. the first concerns the dual function of anchor metadata under the fine-tuning paradigm; the second, the correspondence between the violation taxonomy and the evaluation dimensions; the third, the conditions under which the framework's core design principles transfer to other domains.

Dual Function of Anchor Metadata Under Fine-Tuning

Anchor metadata serves at inference time as a factual constraint: by encoding numerical extrema, directional semantics, and volatility as structured prompt fields, it bounds the space of claims available during generation. Fine-tuning reveals a second, structurally distinct function. The model learns the `[CHART CONTEXT]` anchor block as a format conditioning trigger because it consistently co-occurs with the two-stage output structure across the training set; the block's presence at inference time activates that structure independently of whether the individual field values are complete. The two functions—content grounding and format activation—each contribute to output stability and operate most effectively together.

This co-occurrence learning generalizes to a design principle: in fine-tuned generation systems, structural prompt components simultaneously carry semantic content and the behavioral associations acquired during training. A component that appears consistently alongside a target output pattern becomes a conditional signal for that pattern; its removal at inference time may disrupt format compliance to a degree that training alone does not prevent. Treating structural consistency as an explicit inductive bias therefore offers a low-cost path to improving output stability without additional training data.

Table 5.1: Violation taxonomy and evaluation dimension mapping.

Violation Type	**Evaluation Dimension**
Trend Direction Errors	Mechanistic Grounding, Structural Coherence
Numerical Faithfulness Violations	Structural Coherence
External Causal Hallucinations	Closed-System Compliance
Latent-state Mechanism Wording	Mechanistic Grounding, Dependency Strength

Failure Modes as the Basis for Evaluation Dimension Design

ChartNarrator's evaluation framework was derived directly from the violation taxonomy established in RQ1; each dimension carries a traceable design motivation grounded in observed failure modes. Table 5.1 presents the resulting correspondence between violation types and evaluation dimensions.

Trend Direction Errors directly correspond to Mechanistic Grounding and Structural Coherence; Numerical Faithfulness Violations directly correspond to Structural Coherence; External Causal Hallucinations directly correspond to Closed-System Compliance; Latent-state Mechanism Wording operates across two layers, directly corresponding to Mechanistic Grounding,with Dependency Strength serving as its downstream indicator.

Dependency Strength plays a dual role in this correspondence: it provides a positive assessment of whether P3 establishes a genuine cross-stage structural dependency, and at the same time serves as a downstream symptom indicator for upstream failures such as Latent-state Mechanism Wording.

When P2 is dominated by latent-state language that is when observable chart structure features are replaced by vague internal state descriptions, P3 can only proceed from an empty foundation. The result is an anchor-borrowing pattern, where P3 restates the values in the anchor field instead of establishing new cross-stage dependencies.

A persistently low DS score therefore reflects the propagation effect of latent-state wording across the evaluation dimensions, tracing the path back to the generation quality of P2. This cascading relationship suggests that, at the level of evaluation framework usage, the diagnostic path for a persistently low DS score should begin with examining P2 for latent-state language; P3 generation is a secondary concern.

The syntactic-semantic-pragmatic framework widely used in data-to-text evaluation [26], comprising *Semantic Correctness*, *Syntactic Coverage*, and *Pragmatic Appropriateness*, lacks a systematic correspondence between its dimension definitions and domain-specific failure modes, which limits its diagnostic precision on tasks where fine-grained faithfulness judgments are required.

ChartNarrator's design path, which first establishes a failure mode taxonomy and then derives evaluation dimensions from it, gives each dimension a traceable design motivation and produces evaluation results with more direct diagnostic value. For tasks requiring fine-grained faithfulness diagnosis, this path provides a workable alternative. The cost of this path is a higher degree of domain-specific binding to the evaluation framework; the taxonomy establishment step needs to be re-executed when migrating across domains. However, for any domain specific faithfulness generation task, this cost is worthwhile compared to the improved diagnostic accuracy.

Transferability of Framework Design Principles

Anchor injection as an explicit constraint principle transfers readily to other domains; ToTTo [20] has demonstrated the effectiveness of structural constraints in table-to-text generation. However, the anchor field itself requires domain-specific redesign: medical time-series charts require normal ranges and warning thresholds, while scientific charts require error bars and confidence intervals.

The transferability of the two-stage separation depends on whether the task requires mechanistic explanation. For tasks that require only descriptive narration, the two-stage structure introduces unnecessary complexity. For tasks that require causal reasoning, such as interpreting indicator changes in clinical reports, the separation of Stage 1 and Stage 2 has design value; however, the closed-system constraint of Stage 2 would need to be redefined, since in a medical context the boundary between permissible domain inference and external causal violation is considerably less clear than in the financial domain.

The Kappa validation protocol, comprising stratified sampling, dimension-level decomposition, and manual annotation comparison, is the most directly transferable component of the framework and should be treated as a necessary step in any system using LLM-as-a-Judge [9]; the protocol as implemented in ChartNarrator [12] requires no domain-specific modification before application elsewhere.

The main practical obstacle to framework migration is the domain-specific design cost of the anchor field: in the financial domain, the anchor can be automatically extracted from statistical data; in other domains, the corresponding structural information may require the intervention of domain experts to define and extract, which is an engineering obstacle that the design principle itself does not resolve. At the level of data construction, as discussed in Section 5.4, controlled experiments at the current scale

indicate that the data volume has a larger effect on performance than quality threshold; in migration settings with limited fine-tuning data, prioritizing coverage over strict quality thresholds is therefore a more practical starting point.

ChartNarrator's three design principles transfer at the level of methodology; whether they transfer in practice depends on whether the target domain can supply structural anchor information at reasonable cost.

5.6 Limitations

This work has several limitations, organized across three dimensions: the reliability of the evaluation framework, the scale and scope of the experiments, and residual structural risks at the generation level.

The most significant limitations concern the reliability of the evaluation framework, as they directly bear on the credibility of the RQ3 conclusions. The Stage 2 Judge (Gemini 2.5 Pro) and Generator (Gemini 2.5 Flash) belong to the same model family; prior work has shown that LLM judges exhibit family bias, systematically assigning higher scores to outputs from models within the same family [31, 32], a tendency linked to lower perplexity of same-family outputs relative to the judge's own generation distribution.

Kappa validation defines the credible scope of the Judge by identifying which dimensions are reliable (DS) and which dimensions lack discriminative power due to zero variance; however, Kappa itself cannot eliminate family bias, only mark its boundaries.

A related structural limitation concerns the dual role of the LLM-as-a-Judge in the framework: the same Judge configuration governs training set construction (filtering Stage 2 outputs by WA threshold) and final model evaluation (scoring held-out test outputs). This creates a potential circularity in which the Judge rewards outputs that conform to its own scoring preferences than an independent standard of faithfulness.

Two factors partially mitigate this concern without eliminating it. First, Kappa verification shows that the Judge reaches Substantial agreement with the human annotator on Dependency Strength ($\kappa_w = 0.659$), the most heavily weighted dimension; this confirms that the Judge's DS standard is not purely self-referential. Second, Zero-Shot which uses the same base model as VLM 4.5 but without fine-tuning on Judge-filtered data scores significantly lower than VLM 4.5, indicating that the Judge discriminates on the basis of output quality. However, the possibility that both models have been simultaneously optimized toward the Judge's preferences cannot be excluded, and independent human evaluation on a held-out subset would be required to fully resolve this concern.

A second limitation concerns the structure of the Kappa verification itself: we used a single annotator (the thesis author), so the resulting Kappa measures agreement between one human annotator and the LLM Judge rather than inter-rater reliability between two independent annotators.

Stage 2 evaluations requires annotators to have full working knowledge of morphology routing logic, anchor schema design, and P1/P2/P3 role separation criteria, making an independent second annotator practically unavailable; the single-annotator design is therefore justified, through it remains a methodological limitation. To control the evaluation bias, we required the human annotator to complete all dimension scores independently before consulting the Judge's output; the internal consistency of a single annotator's criteria cannot, however, be verified through Kappa alone. The Fair agreement on Mechanistic Grounding ($\kappa_w = 0.306$) additionally implies that the MG quality standard applied during training set construction reflects the Judge's calibration, not a human consensus; the effective MG lower bound of the training set is therefore more permissive than a human-annotated equivalent would be.

Beyond evaluation reliability, the scale and coverage of the experiments present a second set of limitations that bear on the external generalizability of the findings.

The primary comparison uses a held-out test set of $n = 124$, with the VLM 4.5 versus VLM 5.0 controlled comparison further reduced to $n = 88$; within the morphology-level breakdown, the `trend_drift` bucket contained only $n = 10$ samples, limiting statistical power. Whether the non-significant result ($\Delta\text{WA} = 0.051, p > 0.05$) holds on a larger test set remains unverified.

The training set composition introduces a further coverage boundary. Simple-route samples, comprising 2.6% of the Stage 2 input pool, were excluded from both training and evaluation because they reach

the structural scoring ceiling of the current framework; their generation quality on monophasic or near-monotonic series therefore remains unassessed.

A further boundary concerns the geographical and domain coverage of the dataset: all experiments draw on macroeconomic indicators published by Statistics Netherlands (CBS), so the framework's generation quality on data from other economies, languages, or micro financial sources such as equity prices or corporate financial statements has not been established.

A third class of limitations arises from known structural boundaries within the generation framework risks that are identifiable at the design level but cannot be eliminated through prompt engineering. Four risks warrant mention.

First, morphology-aware routing carries residual uncertainty at the boundary between monotonic and non-monotonic series; even after anchor repair and selective rollback, a small number of boundary samples may be assigned to suboptimal generation paths.

Second, when the anchor schema does not record a pace-change inflection point, the model estimates the transition position directly from the image, an estimate that cannot be cross-checked against the anchor metadata and may therefore introduce numerical imprecision undetected by the Judge.

Third, when the observation window exceeds 100 months, the limited number of anchor constraints forces P1 to compress mid-series structure substantially, and features omitted from P1 cannot surface in the mechanistic reasoning of P2 and P3, placing a systematic ceiling on narrative completeness for long-window samples.

Fourth, when the morphology family is `oscillatory` but the temporal structure contains only a single reversal, the model may produce a `structural_swing` narrative; such morphological mismatches are syntactically well-formed and cannot be detected automatically.

Chapter 6

Conclusions and Future Work

This chapter summarizes the contributions of this thesis and outlines directions for future work.

6.1 Conclusions

Chart-to-text generation for financial time series is a faithfulness-critical task: the generated narrative must be verifiably anchored to the visible data, rather than optimized for fluency alone. Existing frameworks, however, do not simultaneously address the three requirements this demands — grounded data alignment, mechanistic narrative generation, and verified faithfulness evaluation, leaving each dimension managed in isolation.

ChartNarrator addresses these three interrelated challenges by integrating anchor metadata injection, a two-stage generation pipeline, and inter-rater agreement verification into a single modular framework.

Systematic analysis across the preliminary study and Stage 2 annotation identified four violation types that separate into two groups according to their underlying failure mechanisms.

Perceptual accuracy failures, namely Trend Direction Errors and Numerical Faithfulness Violations, stem from the model's misinterpretation of visual signals. Language prior-dominated failures, namely External Causal Hallucinations and Latent-state Mechanism Wording, arise from the model's tendency to introduce off-chart information or generate unfalsifiable mechanistic expressions. Latent-state Mechanism Wording in particular falls outside the intrinsic/extrinsic dichotomy established by Ji et al. [4], constituting a distinct faithfulness failure mode that existing hallucination taxonomies do not capture.

Anchor metadata injection targets perceptual accuracy failures directly. The ablation results confirm a consistent quality gain (ΔWA = +1.802), concentrated in Mechanistic Grounding. Fine-tuning also reveals a secondary function: the anchor block acts as a format conditioning trigger, with its structural presence in the training data sufficient to activate two-stage output compliance at inference time.

Evaluating these quality gains reliably requires verifying that the automated judge aligns with human judgment. Kappa verification shows that LLM-as-a-Judge reaches substantial agreement on Dependency Strength ($\kappa = 0.659$), fair agreement on Mechanistic Grounding and Structural Coherence, and moderate agreement overall ($\kappa = 0.463$). This pattern confirms that the judge functions reliably as a dimension-aware filtering instrument rather than a general-purpose quality metric.

Finally, comparative experiments under a unified evaluation framework reveal that the fine-tuned VLM (WA = 4.306) performs comparably to the Text-Only model (WA = 4.422), both significantly outperforming the No-Anchor and Zero-Shot conditions. At the current experimental scale, training set size has a more prominent effect on performance than quality threshold stringency.

Reliable faithful chart narration depends on the systematic integration of constraint design, task separation, and evaluation verification; no single component is sufficient in isolation.

Contributions

This thesis makes three contributions that extend beyond the specific experimental setup. The violation taxonomy identifies four faithfulness failure modes specific to financial time series chart narration; the Latent-state Mechanism Wording type in particular falls outside the intrinsic/extrinsic dichotomy of existing hallucination frameworks [4, 6], identifying a class of unfalsifiable mechanistic expression that may recur in other chart-grounded generation settings. The anchor metadata and two-stage pipeline together demonstrate that verifiable faithful generation is achievable through structured constraints alone, without reference narratives, which broadens the applicability of faithfulness-oriented generation to settings where gold-standard references are unavailable. The Kappa verification protocol provides a transferable methodology for empirically establishing the reliability boundaries of LLM-as-a-Judge on new faithful generation tasks, rather than treating reliability as a given.

6.2 Future work

The current framework has been validated under controlled, small-scale conditions (CBS Dutch macroeconomic indicators, 1,234 training samples, 7B parameter model). This scale defines the scope of the conclusions and points to two directions for future work.

The first direction concerns scale. The controlled experiment in RQ4 shows that training set size has a more prominent effect on performance than quality threshold stringency at the current scale. Whether this relationship holds at larger sample sizes or parameter scales is unclear; the quality-diversity tradeoff may behave differently as scale increases, and this constitutes an open question that warrants systematic investigation.

The second direction concerns domain transfer. The structural principles of the Two-Stage generation pipeline generalize to other time-series narrative settings, but the anchor schema's field definitions must be redesigned for the target domain. The `higher_means` field, for instance, requires a context-conditional definition in medical time series, where the directionality of a physiological change depends on clinical context; the `volatility` field carries different semantics in scientific results charts, where high variance may indicate measurement noise rather than a substantive signal.

Two methodological limitations of the Kappa verification procedure each have a concrete improvement path whose results would be directly comparable to the values reported in Section 4.4.

The first concerns same-model-family bias. The judge (Gemini 2.5 Pro) and generator (Gemini 2.5 Flash) belong to the same model family. Prior work has indicated that LLM judges tend to assign higher scores to outputs from within the same family [31, 32]. Kappa verification marks the boundaries of judge reliability but cannot isolate the contribution of family bias to the observed agreement values. Rerunning the verification with a cross-family judge, such as GPT-4o evaluating Gemini outputs, would allow the magnitude of this bias to be quantified directly.

The second limitation is the single annotator design: current Kappa measures consistency between a single annotator and the LLM judge. Introducing an independent second annotator can distinguish between "judge vs. human disagreement" and "human vs. human disagreement," more accurately determining whether the judge's reliability boundary stems from limitations of the judge itself or the inherent subjectivity of the annotation task.

Section 5.6 identifies four types of structural generation risks that mark the known boundaries of the current framework. Each boundary reflects a specific design choice in the anchor schema and points to a corresponding improvement direction.

Morphology routing uncertainty could be reduced by enriching the anchor extractor with manually annotated morphology ground truth, giving routing decisions a supervised signal beyond the current threshold judgments on numerical features. The long-window coverage ceiling identified in Section 5.6 points to a complementary direction: dynamically scaling anchor constraint density with window length, so that mid-series structure is not systematically compressed when the observation window exceeds 100 months.

The Text-Only condition's sensitivity to serialization format remains unquantified. The preliminary study observed cases in which timestamp columns were misidentified as numerical values, suggesting that format choice has a substantive effect on generation quality. A systematic comparison of tabular

serialization such as Markdown tables, JSON, and natural language templates would quantify this effect and inform more robust input design.

The current framework defines faithfulness as source-bounded, a design choice that ensures verifiability but constrains how deeply the narrative can explain what the chart describes. A chart may display a sharp rise in the unemployment rate over a given quarter, yet the source-bounded constraint prevents the model from connecting this movement to concurrent macroeconomic events.

This constraint points to a natural extension of the framework: a Stage 3 module built on the Stage 1+2 chart-grounded baseline that explicitly injects time-aligned, verified external event metadata, enabling the model to produce event-contextualized narratives that move from describing chart-visible movements to situating those movements within a validated factual context.

Such an extension would preserve the source-bounded principle as its foundation: Stage 1 and 2 constraints remain active as the chart-level guarantee, and Stage 3 adds a controlled world-knowledge grounding layer on top. Under this design, External Causal Hallucination would shift from a violation type within the current framework to a regulated input in an extended one.

Realizing this extension requires addressing several prerequisite questions: how event metadata is sourced and verified for temporal alignment with the chart window; how faithfulness evaluation criteria should be redefined once external events are admitted as inputs; and how the architecture can ensure that Stage 3 inference remains subordinate to the chart-level constraints established by Stage 1 and 2.

Bibliography

[1] Shankar Kantharaj, Rixie Tiffany Ko Leong, Xiang Lin, Ahmed Masry, Megh Thakkar, Enamul Hoque, and Shafiq Joty. Chart-to-text: A large-scale benchmark for chart summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4010–4023, 2022.

[2] Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2263–2279, 2022.

[3] Zirui Wang, Mengzhou Xia, Luxi He, Howard Chen, Yitao Liu, Richard Zhu, Kaiqu Liang, Xindi Wu, Haotian Liu, Sadhika Malladi, Alexis Chevalier, Sanjeev Arora, and Danqi Chen. CharXiv: Charting territories in large language model evaluation. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

[4] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38, 2023.

[5] T. Loughran and B. McDonald. Textual analysis in accounting and finance: A survey. *Journal of Accounting Research*, 54:1187–1230, 2016.

[6] Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, 2020.

[7] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

[8] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–81, 2004.

[9] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Yao, Rong Xu, Daoyuan Chen, Eric P. Xing, Joseph E. Gonzalez, Ion Stoica, and Hao Zhang. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

[10] Yucheng Han, Chi Zhang, Xin Chen, Xu Yang, Zhibin Wang, Gang Yu, Bin Fu, and Hanwang Zhang. ChartLlama: A multimodal LLM for chart understanding and generation, 2023.

[11] Fangyu Liu, Francesco Piccinno, Syrine Krichene, Chenxi Pang, Kenton Lee, Mandar Joshi, Yasemin Altun, Nigel Collier, and Julian Martin Eisenschlos. MatCha: Enhancing visual language pretraining with math reasoning and chart derendering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12756–12770, 2023.

[12] Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46, 1960.

[13] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. QLoRA: Efficient finetuning of quantized LLMs. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

[14] Alan Lundgard and Arvind Satyanarayan. Accessible visualization via natural language descriptions: A four-level model of semantic content. *IEEE Transactions on Visualization and Computer Graphics*, 28(1):1073–1083, 2022. doi: 10.1109/TVCG.2021.3114770.

[15] Ahmed Masry, Parsa Kavehzadeh, Xuan Long Do, Enamul Hoque, and Shafiq Joty. UniChart: A universal vision-language pretrained model for chart comprehension and reasoning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14662–14684, 2023.

[16] Jinlan Fu, See-Kiong Ng, and Zhiwei Liu. Gptscore: Evaluate as you desire. *arXiv preprint arXiv:2302.04166*, 2023.

[17] Fangyu Liu, Julian Eisenschlos, Francesco Piccinno, Syrine Krichene, Chenxi Pang, Kenton Lee, Mandar Joshi, Wenhu Chen, Nigel Collier, and Yasemin Altun. DePlot: One-shot visual language reasoning by plot-to-table translation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 10381–10399, 2023.

[18] Kung-Hsiang Huang, Mingyang Zhou, Hou Pong Chan, Yi Fung, Zhenhailong Wang, Lingyu Zhang, Shih-Fu Chang, and Heng Ji. Do LVLMs understand charts? analyzing and correcting factual errors in chart captioning. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 730–749, 2024.

[19] Benny J. Tang, Angie Boggust, and Arvind Satyanarayan. VisText: A benchmark for semantically rich chart captioning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7268–7298, 2023.

[20] Ankur P. Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. ToTTo: A controlled table-to-text generation dataset. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 1173–1186, 2020.

[21] Bhuwan Dhingra, Manaal Faruqui, Ankur Parikh, Ming-Wei Chang, Dipanjan Das, and William Cohen. PARENT: Addressing the circular evaluation problem in generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5364–5376, 2019.

[22] Tim Loughran and Bill McDonald. When is a liability not a liability? Textual analysis, dictionaries, and 10-Ks. *The Journal of Finance*, 66(1):35–65, 2011.

[23] Dogu Araci. FinBERT: Financial sentiment analysis with pre-trained language models. *arXiv preprint arXiv:1908.10063*, 2019.

[24] Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. BloombergGPT: A large language model for finance. *arXiv preprint arXiv:2303.17564*, 2023.

[25] Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. G-Eval: NLG evaluation using GPT-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, 2023.

[26] Ehud Reiter and Robert Dale. *Building Natural Language Generation Systems*. Cambridge University Press, Cambridge, UK, 2000.

[27] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfan Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-VL: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

[28] Google DeepMind. Gemini 2.0 flash, flash-lite and pro. Technical report, Google DeepMind, 2025. URL `https://developers.googleblog.com/en/gemini-2-family-expands/`.

[29] Google DeepMind. Gemini 2.5: Our newest Gemini model with thinking. Technical report, Google DeepMind, 2025. URL `https://blog.google/innovation-and-ai/models-and-research/google-deepmind/gemini-model-thinking-updates-march-2025/`.

[30] Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. LIMA: Less is more for alignment. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

[31] Evangelia Spiliopoulou, Riccardo Fogliato, Hanna Burnsky, Tamer Soliman, Jie Ma, Graham Horwood, and Miguel Ballesteros. Play favorites: A statistical method to measure self-bias in LLM-as-a-judge, 2025.

[32] Koki Wataoka, Tsubasa Takahashi, and Ryokan Ri. Self-preference bias in LLM-as-a-judge, 2024.

Appendix A

Appendix

A.1 Data Construction Tables

Table A.1: Selected Financial and Macroeconomic Indicators from CBS Open Data

Indicator Name	CBS ID	Domain	Brief Description
Consumer Confidence	83693ENG	Confidence	Monthly survey index of household economic sentiment.
Producer Confidence	81234ENG	Confidence	Monthly survey index of sentiment in the manufacturing sector.
Inflation (CPI)	83131ENG	Prices	Consumer price index for general price-level change.
Energy Prices (Consumers)	85592ENG	Prices	Consumer energy price index.
Producer Price Index (Manufacturing)	85771ENG	Prices	Output price index for the manufacturing sector.
Terms of Trade (Goods)	85962ENG	Trade	Ratio of export to import price indices for goods.
Existing Homes: Average Price	85773ENG	Housing	Average transaction price of sold residential dwellings.
Labour Participation & Unemployment	80590ENG	Labour	Monthly labour-force participation and unemployment rates.
Wages (CAO Total)	82838ENG	Labour	Collective labour agreement wages, including special payments.
Manufacturing Turnover	81817ENG	Industry	Turnover index for manufacturing branches.
Retail Turnover	81810ENG	Industry	Turnover index for retail branches.
Industry Production & Sales	85806ENG	Industry	Production and turnover indices across industrial sectors.
Investment in Tangible Assets	85939ENG	Investment	Volume index of investment in tangible fixed assets.
Construction Production Index	85809ENG	Construction	Turnover index for construction branches.
Civil Engineering Input Price Index	86049ENG	Construction	Input price index for civil engineering works.
Government Finance Key Figures	85968ENG	Government	Quarterly figures on revenue, expenditure, balance, and debt.
Household Consumption Expenditure	85937ENG	Consumption	Volume index of household consumption expenditure.
Bankruptcies Key Figures	82242ENG	Bankruptcies	Monthly number of pronounced bankruptcies by legal form.

Continued on next page

Indicator Name	CBS ID	Domain	Brief Description
Population Dynamics (Monthly)	83474ENG	Demographics	Monthly counts of births, deaths, and net migration.

Table A.2: Preprocessing Rules Applied to CBS Indicators

Rule	Applied to	Indicators
12-month rolling sum	Flow variables (monthly counts)	Existing Homes: Average Price; Bankruptcies Key Figures; Population Dynamics (Monthly)
12-month rolling mean	Level, index, and rate variables	All remaining indicators (see Table A.1)
No smoothing	Windows < 24 months	Applied adaptively across all indicators when the observation window is shorter than 24 months

Table A.3: Observation Window Sampling Strategy

Window Type	Length (months)	Sampling Probability	Rationale
Short-term (Micro)	12–24	20%	Captures localised structural episodes and short-cycle movements
Medium-term (Cycle)	24–96	60%	Covers full business cycles and multi-phase macroeconomic patterns; forms the majority of training instances
Long-term (Trend)	96–240	20%	Captures extended secular trends and long-run structural shifts

Note. Window boundaries are sampled randomly within each range. If the total series length is shorter than the upper bound of a given tier, the effective maximum is capped at the series length. Windows shorter than 24 months use raw monthly values without smoothing (see Table A.2).

Table A.4: Chart Rendering Parameters

Parameter	Options	Selection Rule
Plot style	`seaborn-whitegrid`, `seaborn-darkgrid`, `ggplot`, `bmh`, `fivethirtyeight`, `seaborn-ticks`	Sampled uniformly at random per chart
Line colour	8-colour palette (`#1f77b4`, `#ff7f0e`, `#2ca02c`, `#d62728`, `#9467bd`, `#8c564b`, `#e377c2`, `#17becf`)	Sampled uniformly at random per chart
Marker style	`o`, `s`, `^`, `v`, `D`, `p`	Applied only when series length < 40 points; sampled uniformly at random
Grid style	3 configurations varying linestyle (`-`, `:`, `-`), opacity (0.2–0.4), and linewidth (0.5–0.8)	Sampled uniformly at random per chart
Chart type	Line chart; diverging bar chart	Line chart used by default; diverging bar chart applied when the series contains negative values

Continued on next page

Parameter	Options	Selection Rule
X-axis tick interval	1 month (≤24 points); 3 months (≤60 points); 6 months (≤120 points); 1 year (>120 points)	Determined by series length
Output resolution	100 DPI	Fixed across all charts
Figure size	10×6 inches	Fixed across all charts

Note. Randomisation across plot style, colour, marker, and grid is applied independently per chart instance. Fixed parameters (resolution, figure size) are held constant to ensure that visual scale differences across charts reflect only data variation rather than rendering artefacts.

A.2 Two-Stage Anchor Comparison Tables

Table A.5: Stage 1 Anchor Fields Exposed to the Generation Prompt

Field	Group	Description
`start_date` / `end_date`	Time window	Start and end dates of the effective observation window.
`duration_months`	Time window	Length of the observation window in calendar months.
`start_value` / `end_value`	Values	First and last observed values within the window.
`min_value` / `max_value`	Values	Minimum and maximum values within the window.
`mean_value` / `median_value`	Values	Arithmetic mean and median of the series.
`std_value`	Values	Standard deviation of the series.
`absolute_change`	Changes	Difference between end value and start value.
`percentage_change`	Changes	Percentage change from start to end.
`annualized_change`	Changes	Compound annual growth rate over the window.
`peak.date` / `peak.value`	Extremes	Date and value of the global maximum.
`trough.date` / `trough.value`	Extremes	Date and value of the global minimum.
`coefficient_of_variation`	Volatility	Ratio of standard deviation to mean.
`range`	Volatility	Difference between maximum and minimum values.
`value_type`	Meta	Semantic type of the series (`count` or `index`).

Note. This table lists only the anchor fields explicitly exposed to the Stage 1 generation prompt. Extractor-side metadata, preprocessing details, data quality tracking, and boundary-position flags are omitted.

Table A.6: Stage 2 Anchor Fields Exposed to the Generation Prompt

Field	Group	Description
`unit_description`	Variable semantics	Description of the unit and measurement basis.
`higher_means`	Variable semantics	Directional interpretation of increasing values (`better`, `worse`, or `neutral`).

Continued on next page

Field	Group	Description
direction_note	Variable semantics	Instruction clarifying how directional changes should be interpreted.
closed_system_note	Variable semantics	Constraint on allowable external reasoning.
start / end	Time window	Start and end dates of the effective window.
duration_months	Time window	Length of the observation window in months.
start_value / end_value	Values	First and last observed values.
absolute_change / percentage_change	Changes	Absolute and percentage change over the window.
peak.date / peak.value	Extremes	Date and value of the global peak.
trough.date / trough.value	Extremes	Date and value of the global trough.
trend_structure.phases	Trend structure	Ordered phases with direction, date range, start/end values, and magnitude.
trend_structure.inflections	Trend structure	Internal turning points with date, value, and type.
morphology_signal.family	Morphology signal	Assigned morphology family.
morphology_signal.family_confidence	Morphology signal	Confidence level of the family assignment.
morphology_signal.modifier_flags	Morphology signal	Boolean modifiers affecting interpretation.
binary_phase_signal.eligible_for_phase_routing	Binary phase	Whether the series qualifies for binary phase routing.
binary_phase_signal.binary_pc	Binary phase	Percentage change used for binary decision.
binary_phase_signal.phase_count_raw	Binary phase	Raw count of detected phases.
binary_phase_signal.reversal_evidence	Binary phase	Indicator of reversal behaviour.
binary_phase_signal.prominent_internal_turn	Binary phase	Indicator of strong internal turning points.
complexity_signal.route_label	Complexity signal	Assigned generation route (simple or complex).
complexity_signal.is_simple	Complexity signal	Boolean flag for simplified generation.
complexity_signal.routing_reason	Complexity signal	Rule triggering the routing decision.
complexity_signal.trend_type	Complexity signal	Derived trend classification.
complexity_signal.phase_count	Complexity signal	Number of structural phases.

Continued on next page

Field	Group	Description
complexity_signal.generation_guidance	Complexity signal	Instruction guiding narrative generation.

Note. This table lists only the anchor fields explicitly exposed to the Stage 2 generation prompt. Extractor-side metadata, internal classification metrics, and non-exposed schema components are omitted.

A.3 Two-Stage Generation Prompts

Stage 1 Generation Prompt

The following prompt template was used for Stage 1 constrained factual narration. Dynamic fields enclosed in [brackets] are populated at runtime from the chart filename and anchor metadata.

```
You are a Financial Columnist. You will analyze a chart about
[topic] covering [time_context].

You are given TWO sources of truth:
(A) the chart image
(B) the aligned tabular data excerpt (same time window as the chart)

OBJECTIVE
Write an analysis that is faithful to the chart and grounded in
the provided data. Your narrative may include economic mechanisms,
but must remain strictly supported by the provided inputs.
Any real-world event references must follow the event constraint below.

INPUTS
- Time window: [time_context]
- Value type of the chart: [value_type]
- Aligned data excerpt (authoritative numeric anchors):
  [data_excerpt]
- Optional event list within the time window (may be empty):
  [event_list]

CRITICAL CONSTRAINTS

1) Temporal scope
Every statement must refer strictly to the period [time_context].
Do not mention periods, trends, or comparisons outside this window.

2) Grounding requirement
Every trend or claim must explicitly cite at least one numeric
anchor from (B), such as exact values and dates, changes between
two dates, or peaks, troughs, and breakpoints.
Claims without numeric anchors are not allowed.

2.5) Value type constraint
Interpret all numeric values strictly according to the provided
value type. If value type is "index", treat values as index points
representing relative levels. If value type is "count", treat
values as absolute counts or aggregated totals.
Do not reinterpret, rescale, or relabel the values.

3) Mechanism constraint
Economic mechanisms must be explicitly tied to the cited numeric
anchors. Do not use generic macroeconomic narratives not directly
linked to the observed data changes. Abstract economic mechanisms
are allowed only when they describe states implied by the observed
numeric patterns, not external causal drivers.

4) Event constraint
If an event list is provided, you may only reference events from
that list. If no event list is provided, do not name specific
historical events. Explain movements using abstract economic
mechanisms only.

5) Faithfulness to the chart
```

```
Do not describe visual elements that are not present. Do not infer
additional series, annotations, smoothing, or extrapolated values.
Do not change the scale, unit, or interpretation of the provided data.

6) Uncertainty handling
If a claim cannot be directly supported by (A) or (B), mark it as
UNSUPPORTED in Section 1 and do not include it in Section 2.

7) Actual data window precedence
All claims must be grounded strictly in the actual data window.
Do not speculate about periods with missing or dropped data.

8) Derived metric validity
Only reference derived metrics explicitly marked as valid in the
anchor metadata. Do not restate invalid metrics using alternative
wording.

9) Extremes vs terminal values
Do not describe a peak or trough as the ending or final level
unless it coincides with the end of the actual data window.

RESPONSE FORMAT
Return exactly two sections. No additional text.

[SECTION 1: EVIDENCE TABLE]
Provide 3-6 evidence rows, each corresponding to a distinct phase,
turning point, or stable segment visible in the chart.
Each row must include:
- Visual pattern: what changes, where, and when
- Numeric anchor(s): exact date/value/change cited from (B)
- Mechanism: economic linkage tied to the cited numeric anchors
- Event mapping: only if allowed by the Event constraint; else N/A
- Support tag: SUPPORTED or UNSUPPORTED

[SECTION 2: THE FINANCIAL COLUMN]
Write 180-250 words. Prioritize faithfulness over completeness.
Structure:
- The Visual Trend: 1-2 sentences grounded in numeric anchors
- The Story Behind the Data: connect supported mechanisms to patterns
- Bottom Line: one sentence restating key movement and mechanism,
  introducing no new facts
Section 2 must not introduce any concept not in Section 1.
```

Stage 2 Generation Prompt

The following prompt template was used for Stage 2 source-bounded mechanistic reasoning. The placeholder `[anchor_text]` is populated at runtime with the formatted Stage 2 anchor metadata (see Table A.6).

```
You are a Stage-2 financial narrative generation model. Stage-1 has
already verified all numeric facts. Stage-2 is a CLOSED-SYSTEM
mechanistic reasoning task.

OUTPUT FORMAT
- Begin directly with "-- Paragraph 1"
- No bullet points
- No reasoning steps in the output
- Number of paragraphs depends on route_label (see ROUTING LOGIC)

TASK AND EVIDENCE
Generate a chart-grounded narrative that moves from observed curve
structure to mechanistic interpretation, stays inside the
chart-bounded evidence space, and explains through structural
relations visible in the chart itself.

Use only: (A) the chart image  (B) the anchor metadata below
(C) the variable semantics in the metadata.
Do not use outside finance knowledge, real-world events,
institutions, policies, or unplotted variables.

Evidence hierarchy:
- Chart image: primary source for overall shape, range shifts,
  plateaus, rebounds, repeated swings, and range re-entry.
- Anchor metadata: binding layer for exact values, dates, extrema,
  encoded phases, morphology signal, and variable semantics.
- Do not treat metadata as world knowledge or as a substitute for
  chart reading.

CONSTRAINTS

1. NUMERIC FAITHFULNESS: every value, date, duration, and phase
   boundary must come from anchor metadata.
2. CLOSED-SYSTEM: respect closed_system_note; no external crises,
   institutions, policies, or actors.
3. DIRECTION INTEGRITY: respect direction_note and higher_means.
4. BOUNDARY DISCIPLINE: respect boundary_flags; window-edge extrema
   are not definitive long-run bounds.
5. ANCHOR BINDING: every interpretation must tie to visible
   structure, anchor values, explicit phases, morphology family,
   or binary phase signal.
6. LANGUAGE: no hidden actors, hidden conditions, or unplotted
   causes; prefer specific structural relations over generic wording.

FORBIDDEN (never use):
- firms exited / producers expanded / investors became cautious
- demand weakened / credit conditions tightened / policy faded
- exhaustion / release / build-up / accumulated pressure
- fading momentum / pent-up / overhang
- carry-over / conditioned / regime (unless grounded in a visible
  level, range, plateau, or turning-point relation)

FORBIDDEN DOMAIN INFERENCES:
- Do not translate structural movement into domain-level conclusions.
- "indicating an improvement in labour market conditions"
- "suggesting a restoration of trade competitiveness"
- "higher negotiated wages across all sectors"

ROUTING LOGIC
- complexity_signal.route_label = simple  -> SIMPLE ROUTE (2 paras)
- complexity_signal.route_label = complex -> COMPLEX ROUTE (3 paras)
Do NOT infer route from phase_count.

NARRATIVE STRUCTURE

-- Paragraph 1: OBSERVED STRUCTURE (all routes)
Describe only. No explanation, no mechanism, no causation.
```

```
Cover trajectory, turning points, peaks, troughs, plateaus.

SIMPLE ROUTE (exactly 2 paragraphs):
-- Paragraph 2: Mechanism + Structural Basis
Begin with mechanism statement, then ground in visible structure.
Acceptable bases: endpoint-at-extreme, uninterrupted extension,
contained retracement, threshold crossing.

COMPLEX ROUTE (exactly 3 paragraphs):
-- Paragraph 2: MECHANISTIC INTERPRETATION
Write exactly two linked mechanisms spanning the full structure.
Family-specific:
- trend_drift: stage changes, pace shifts, plateau interruptions
- structural_swing: reversals, step changes, partial recovery
- oscillatory_or_seasonal_regime: repeated swings, band recurrence

-- Paragraph 3: DEPENDENCY RESOLUTION
Explain why later structure follows from earlier structure.
Name specific levels, ranges, dates, or turning locations.
No restatement of Paragraph 2. No new external material.

Now generate the final narrative only.

[anchor_text]
```

A.4 Two-Stage Judge Prompts

Stage 1 LLM-as-a-Judge Prompt

The following prompt was used for Stage 1 faithfulness filtering. The judge receives the chart image, anchor metadata, and narrative text as input, and returns dimension scores in JSON format. Dimension weights are: *Semantic Correctness* 0.50, *Syntactic Coverage* 0.30, *Pragmatic Appropriateness* 0.20.

```
You are an evaluation model for a constrained data-to-text
generation task.

TASK CONTEXT
The narrative was generated under strict constraints from structured
anchor metadata derived from a chart and aligned tabular data.
This is constrained factual narration, not open-ended explanation.

AVAILABLE SOURCES OF TRUTH
(A) the chart image (visual structure and trends)
(B) the anchor metadata (time window, values, changes, extremes,
    validity flags)
(C) the narrative text itself
Do NOT use any external world knowledge, economic theory, or
background facts.

DIMENSION 1 - SEMANTIC CORRECTNESS (weight 0.50)
Evaluate whether every factual claim is semantically accurate and
directly supported by the anchor metadata.
Scope: numeric values, trend directions, temporal references,
extrema identification, derived metrics, extrema vs terminal values.

Error severity:
CRITICAL (-2 to -3): value off >10%, wrong trend direction,
  invalid derived metrics, wrong temporal ordering.
MAJOR (-1 to -2): value off 1-10%, date errors >1 month,
  misidentified extrema location.
MINOR (-0.5 to -1): rounding <1%, day vs month precision.

Score guide:
5 = perfect accuracy, no errors
4 = at most one minor error
3 = one major error OR 2-3 minor errors
2 = one critical error OR 2+ major errors
1 = multiple critical errors or systematic misalignment

DIMENSION 2 - SYNTACTIC COVERAGE (weight 0.30)
```

```
Evaluate whether the narrative covers the salient patterns in the
chart with appropriate structure and completeness.
Scope: major trend phases, critical data points, level of detail,
logical sequencing, balance between summary and detail.

Penalise: over-coverage (excessive minor fluctuations),
under-coverage (omitting major shifts, missing extrema), structural
issues (illogical sequencing, fragmented or imbalanced coverage).

Score guide:
5 = optimal coverage, all salient patterns, well-structured
4 = minor gaps or imbalances, all major patterns covered
3 = some under-coverage or slight over-coverage
2 = significant gaps or excessive focus on minor details
1 = major patterns omitted or severely imbalanced

DIMENSION 3 - PRAGMATIC APPROPRIATENESS (weight 0.20)
Evaluate whether the narrative respects the pragmatic constraints
of the generation task.
Scope: actual vs requested data window, no speculation about
missing periods, correct value type interpretation, no external
events unless permitted, appropriate hedging.

Critical violations (-2 to -3): using requested window,
speculation about dropped data, introducing external events,
external causal explanations.
Major violations (-1 to -2): wrong value type interpretation,
interpretive claims without hedging.
Minor violations (-0.5 to -1): slight temporal ambiguity,
borderline interpretive phrasing.

Score guide:
5 = all constraints fully satisfied
4 = at most one minor violation
3 = one major violation OR 2-3 minor violations
2 = one critical violation OR 2+ major violations
1 = multiple critical violations or systematic non-compliance
```

Stage 2 LLM-as-a-Judge Prompt

The following prompt was used for Stage 2 faithfulness evaluation (version 4.6, finalized after six rounds of iterative refinement against Kappa results; see Section 4.4). Dynamic fields `[route_label]`, `[morphology_family]`, `[anchor_text]`, and `[narrative_text]` are populated at runtime. Dimension weights are: *Mechanistic Grounding* 0.35, *Dependency Strength* 0.35, *Closed-System Compliance* 0.20, *Structural Coherence* 0.10.

```
You are evaluating a Stage-2 financial chart narrative. Stage-1
has already guaranteed numeric faithfulness. Assess EXPLANATORY
REASONING quality only, using (A) chart image, (B) anchor metadata,
(C) narrative text. Do not use external world knowledge.
Target standard: Ver2 -- source-bounded mechanistic reasoning only.

ROUTING CONTEXT (injected from generation metadata)
  route_label:       [route_label]
  morphology_family: [morphology_family]
Read route_label directly. Do NOT infer route from paragraph count.
  simple  -> expect exactly 2 paragraphs (P1 + P2)
  complex -> expect exactly 3 paragraphs (P1 + P2 + P3)

GLOBAL TEST - SOURCE-BOUNDEDNESS
A claim PASSES if justified from: chart shape, anchor values/dates,
plotted variable semantics.
A claim FAILS if it requires: external events/institutions, hidden
actors not in the chart, unplotted causal variables, real-world
knowledge outside chart and metadata.

DIMENSION 1 - MECHANISTIC GROUNDING (weight 0.35)
This dimension owns ALL latent-wording and domain-inference
problems.

Latent state wording (penalise here):
- "financial resilience weakened", "exhaustion of upward momentum",
  "build-up of structural pressure", "fading momentum"

Domain inference wording (also penalise here):
- "indicating an improvement in labour market conditions"
- "suggesting a restoration of trade competitiveness"
- "higher negotiated wages across all sectors"

Specificity test: replace P2 mechanism with any other series of
the same direction -- if description still holds, it is a generic
label, not a chart-specific mechanism.
Minimum requirement: at least one chart-specific feature must be
present in P2: rate change, structural inflection, or stage-level
rhythm with >=2 distinct stages.

Forced deduction: P2 describes only direction and endpoint with
none of the above features -> MG must be <= 3.

Score guide:
5 = mechanisms are chart-specific and falsifiable
4 = mostly specific; minor generic bridge wording
3 = partially specific; some direction labels or endpoint summary
2 = mostly direction labels or latent state language
1 = no mechanism; pure description restatement

DIMENSION 2 - DEPENDENCY STRENGTH (weight 0.35)

Simple route: DS upper limit = 3. DS is evaluated on P2 only.
DS = 3 for honest single-phase persistence is NOT a penalty.

Complex route: DS up to 5, family-differentiated.

Dependency relation types:
reference_ceiling, bounded_retracement, failed_recovery,
shifted_range, band_recurrence, base_shift, generic_continuation.

P3 anchor-borrowing check: if P3 reuses the same level already
named in P2 without adding new structural information,
anchor_borrowing = true -> DS = 3 or lower.
```

```
Morphology mismatch caps:
- morphology_mismatch = true -> SC cannot exceed 4
- oscillation_evidence_type = single_reversal (oscillatory family)
  -> DS cannot exceed 4

Score guide (complex route):
5 = non-borrowing, named anchor, relational dependency, P3 distinct
4 = dependency present but generic or unnamed anchor
3 = anchor-borrowing OR simple-route ceiling
2 = mostly sequential; P3 paraphrases P2
1 = no dependency; pure restatement

DIMENSION 3 - CLOSED-SYSTEM COMPLIANCE (weight 0.20)
Covers external entity/actor violations only. Latent wording
and domain inference are handled by MG -- do not double-penalise.

Hard violations:
- external event or institution (named or implied)
- hidden economic actor not visible in chart
- unplotted causal variable introduced as fact
- forecast, recommendation, or policy commentary

If anchor metadata contains a non-empty closed_system_note, any
phrase matching its forbidden patterns -> CSC must be <= 3.

Score guide:
5 = no external violation
4 = very minor borderline phrasing
3 = one ambiguous phrase adds mild external colouring
2 = one clear external violation
1 = multiple or severe external violations

DIMENSION 4 - STRUCTURAL COHERENCE (weight 0.10)
Evaluate paragraph-function coherence given the route.

Simple route (P1 + P2):
P1 = observed structure only; P2 = mechanism + structural basis.
P3 absence is correct -- never deduct for missing P3.
If P2 reproduces specific values from P1 with no structural
feature beyond direction + endpoint -> SC must be <= 3.

Complex route (P1 + P2 + P3):
P2 mechanism count by family:
  trend_drift:     2 stages
  structural_swing: 2 stages (pre/post reversal)
  oscillatory:     recurrence pattern (strict 2-stage not required)
P2 must cover at least N_stages - 1 mechanisms, where N_stages is
the count of structurally distinct stages visible in P1.
If p3_role_distinct_from_p2 = false -> SC cannot exceed 4.

Score guide:
5 = clean progression, each paragraph performs its role clearly
4 = coherent; minor blur between roles
3 = structure present but uneven
2 = roles confused; wrong mechanism count
1 = incoherent or major role failure

[anchor_text]
[narrative_text]
```

A.5 Fine-Tuning Parameters

Table A.7: QLoRA fine-tuning hyperparameters for *VLM 4.5*. Unless noted, the same configuration applies to all fine-tuned experiments (NoAnchor 4.5, VLM 5.0, Text-Only 4.5).

Parameter	**Value**	**Notes**
Framework		
Training framework	LLaMA-Factory v0.9.1	
Fine-tuning method	SFT with LoRA	
Chat template	`qwen2_vl`	`qwen` for Text-Only
Hardware	A100 40 GB	Google Colab Pro
Quantization and adapter		
Quantization	4-bit NF4 (bitsandbytes)	
LoRA rank	64	
LoRA alpha	32	$\alpha/r = 0.5$
LoRA dropout	0.05	
LoRA target modules	all	All linear layers
Training		
Epochs	8	
Learning rate	1×10^{-5}	
Batch size per device	2	1 for Text-Only (OOM)
Gradient accumulation	8	16 for Text-Only; effective batch = 16
Gradient checkpointing	enabled	Required for Text-Only
Optimizer	`paged_adamw_32bit`	
Cutoff length	4,096 tokens	
Precision	`bf16`	`fp16` for Text-Only
Evaluation and checkpointing		
Evaluation strategy	every 100 steps	
Save strategy	every 200 steps	3 checkpoints retained
Training time (VLM 4.5)	≈90 min	352 steps total

A.6 Kappa Stratum

Table A.8: Per-stratum linear-weighted Cohen's Kappa (κ_w) for Mechanistic Grounding (MG), Dependency Strength (DS), and Structural Coherence (SC). Strata are defined by WA tier, `anchor_borrowing`, and `morphology_mismatch` flags; see Section 4.4 for stratum definitions. N/A denotes strata where κ_w is undefined owing to insufficient score variance.

Stratum	n	**MG** κ_w	**DS** κ_w	**SC** κ_w
A ceiling	20	0.000[a]	0.000[a]	0.000[a]
B mid-high (trend)	10	0.000	0.643	0.118
C mid-high (swing)	10	0.130	0.590	0.091
D mid-high (osc)	10	0.000	0.412	0.419
E mid-low	8	−0.200	0.273	0.200
F low	3	N/A	1.000[b]	1.000[b]
G anchor_borrowing	12	0.163	0.000	0.167
H morph_mismatch	7	0.023	0.000	0.000

[a]Stratum A yields $\kappa_w = 0.000$ across all dimensions owing to the Kappa paradox: when both the Judge's and the human annotator's score distributions are highly concentrated, linear-weighted Kappa approaches zero regardless of the observed agreement rate. This is a known statistical artefact and does not indicate substantive disagreement.

[b]Stratum F ($n = 3$) yields $\kappa_w = 1.000$ for DS and SC; this reflects the small sample size rather than a meaningful reliability estimate and should not be interpreted as substantive agreement.