STATE-SPACE MODELING IN NATURAL LANGUAGE

Anonymous authors
Paper under double-blind review

ABSTRACT

In many real-world applications, the goal is to infer the latent dynamics of an underlying process from a sequence of observations. Traditionally, state-space models such as hidden Markov models (HMMs) have been used to represent dynamic systems by positing a discrete or continuous latent state space, where states evolve according to a *transition* model and observations follow an *emission* model. However, these models are typically limited to structured observations and numerical state representations that are often not interpretable or identifiable. In this paper, we propose a new class of state-space models that formulates *state inference* and *prediction* as language tasks, with both the transition and emission models implemented by a large language model (LLM). In our model, latent states are represented as concise natural-language descriptions, while observations may comprise large and unstructured text corpora. We develop a post-training procedure to fine-tune LLMs for state inference and prediction, inspired by variational inference algorithms in classical state-space modeling. We demonstrate the promise of this approach through preliminary experiments on disease progression modeling from clinical notes and on modeling geopolitical relationships using news articles.

Track: Research

1 INTRODUCTION

Large language models (LLMs) have limited context windows, and their performance can degrade sharply as prompt length increases (Hong et al., 2025). In many real-world applications, the input context is produced by an evolving process that continually accumulates text over time. For example, in healthcare settings, clinicians may wish to use LLMs to assist in analyzing a patient's expanding electronic health record (EHR) (Sellergren et al., 2025). Fundamentally, patients in an EHR dataset can be viewed as independent realizations of stochastic health trajectories governed by shared latent dynamics. In such settings, a potentially effective strategy for compressing the growing input context is to model these latent dynamics, thereby learning to extract past information that is most relevant at each time point and most useful for downstream predictions at future time points.

A variety of methods have been proposed for handling long-context inputs, yet these approaches generally do not address settings in which the context itself forms a time series. Previous work includes methods based on sparse attention (Roy et al., 2021), memory-compression architectures such as Compressive Transformers (Rae et al., 2019), and inference-time methods including context condensation and recursive language modeling (Khattab et al., 2021; Zhang et al., 2025). In this paper, however, we do not focus on long-context modeling in general, but rather on contexts generated by real-world, time-stamped stochastic processes that produce continually growing text corpora. Examples include sequences of clinical notes in a patient's EHR, which document patient visits over time, as well as evolving collections of news articles describing geopolitical events or financial markets. Our key hypothesis is that learning to characterize the shared latent dynamics underlying these processes is equivalent to learning how to compress their growing context at each point in time.

Building on the observations above, we propose a state-space modeling approach for processing long contexts generated by temporal processes. State-space models (SSMs) have long served as a foundational framework for modeling sequential data and dynamical systems in control theory (Hamilton, 1994), signal processing (Rao & Arun, 2002), and time-series analysis (Aoki, 2013; Durbin & Koopman, 2012). In an SSM, the evolution of an observed sequence is governed by an underlying latent state with structured *transition* dynamics, while observations are generated through a separate *emission* process. This separation between latent dynamics and observations yields a compact and

interpretable representation of temporal dependencies, which enables SSMs to capture long-range structure through recursive state updates rather than explicit pairwise interactions. Previous work has extended SSMs using deep neural networks to parameterize state transitions and emission distributions, as in the case of Deep Markov Models and its variants (Krishnan et al., 2015; 2017; Johnson et al., 2016; Alaa & van der Schaar, 2019). However, these approaches have largely been limited to structured or tabular data in traditional time-series modeling settings.

Our modeling approach conceptualizes context compression as a state-space modeling problem. In our framework, sequences of text corpora are assumed to be generated by an emission distribution conditioned on an underlying latent state representing the context, while state evolution satisfies a Markov property in which each state depends only on its predecessor. Crucially, these latent states are represented as natural-language summaries of the observed text at each time step. Prior work suggests that training objectives based on predicting future summaries can outperform standard next-token prediction in LLMs (Mahajan et al., 2025). From this perspective, context compression becomes equivalent to modeling the dynamics of an evolving system rather than truncating or heuristically condensing input. This state-space decomposition allows us to formulate both *state inference* (context compression) and *state prediction* as language modeling tasks.

To operationalize our proposed SSM-based framework, we develop a heuristic post-training procedure for fine-tuning LLMs to perform state inference and state prediction, drawing inspiration from variational inference methods in classical state-space modeling. We demonstrate the effectiveness and utility of this approach through preliminary experiments on disease progression modeling using clinical notes and on modeling geopolitical relationships from news articles.

2 METHODS

We consider the problem of sequential modeling over a time-indexed sequence of text corpora

$$X_{1:T} = (X_1, X_2, \ldots, X_T), \quad X_t \in \mathcal{X},$$

where each X_t denotes observed, unstructured text (e.g., news articles or clinical notes) at time step t. The collection $X_{1:T}$ constitutes the available context at time T. A common objective is to perform a downstream task by responding to a query Q conditioned on this context using a language model, i.e., $Y = \mathrm{LLM}(Q; X_{1:T})$. For instance, the query Q may be a question about a patient's likely health outcomes in the future given all the clinical notes $X_{1:T}$ collected in their previous visits. However, performance on such tasks often degrades as the context $X_{1:T}$ grows, which motivates the need for principled methods to capitalize on the temporal structure of $X_{1:T}$ to compress the context.

State-space modeling of temporal context. We assume that the sequence $X_{1:T}$ is generated via a state-space model (SSM), with a factorized joint distribution of observations and latent states:

$$p(X_{1:T}, Z_{1:T}) = p(Z_1) \prod_{t=2}^{T} p(Z_t \mid Z_{t-1}) \prod_{t=1}^{T} p(X_t \mid Z_t), \tag{1}$$

where $Z_t \in \mathcal{Z}$ is the latent state of the dynamic process generating document X_t at time t. Here, we assume that this process is Markovian: the states evolve so that each new state Z_t depends only on the previous state Z_{t-1}. Thus, the joint distribution of observed text and latent states can be fully determined by an *emission* distribution $p(X_t \mid Z_t)$ and a *state transition* distribution $p(Z_t \mid Z_{t-1})$.

We further assume that the state space $\mathcal{Z}$ consists of natural-language strings that explicitly encode the system's underlying state. This enables latent states to be both expressive and interpretable, while maintaining compatibility with standard language modeling objectives. For example, in an EHR setting, a latent state may capture a patient's evolving health across multiple comorbidities summarized in textual form. Individual observations X_t (e.g., physician notes) are then generated as conditionally independent views of this state. This perspective naturally accommodates long contexts, as information is propagated through the recurrent state dynamics rather than requiring direct interactions among all past observations. Moreover, representing states in natural language allows the model to leverage pretrained language priors to derive principled algorithms for state inference.

Learning and inference. By representing latent states as natural-language strings, we can formulate both learning and inference as standard language modeling tasks. Concretely, the transition model

$p(Z_t|Z_{t-1})$ and emission model $p(X_t|Z_t)$ can be parameterized as autoregressive language models, where state evolution and observation generation can be learned via next-token prediction objectives. Inference over latent states similarly reduces to conditional text generation. Given prior context, the posterior distribution over the latent states $p(Z_t|X_{1:t})$ can be approximated by prompting a language model to generate a state description consistent with past states and current observations.

A standard way to conduct learning in inference with deep learning models is to jointly train (i) an inference model and (ii) the emission and transition models by maximizing the marginal log-likelihood $\log p(X_{1:T})$ end-to-end (Krishnan et al., 2017). This objective is generally intractable, therefore it is common to optimize its variational lower bound (ELBO) with an amortized inference model:

$$\mathcal{L}_{\theta,\phi} = \sum_{t=1}^{T} \log p_{\text{LLM}_\theta}(X_t|Z_t) + \log p_{\text{LLM}_\theta}(Z_1) + \sum_{t=2}^{T} \log p_{\text{LLM}_\theta}(Z_t|Z_{t-1}) - \log q_{\text{LLM}_\phi}(Z_{1:T}|X_{1:T}),$$

where p_{LLM_θ} and q_{LLM_ϕ} are the token distributions induced by two LLMs with parameters θ and ϕ, or equivalently, a single LLM steered to model observations, states as well as state inferences using different prompts. We developed a simplification of the ELBO loss that first starts by labeling the latent states $\{Z_t\}$ using a high-capacity teacher model applied with backward-induction, and then training the inference model in a supervised fashion. Full pseudocode is provided in the Appendix (Algorithm 1).

2.1 Baselines and Evaluation

We compare against embedding-based HMMs, TopicGPT+HMM, Neural HMMs, a forward LLM baseline, and an LSTM classifier (see Appendix A.2). Discrete baselines use latent cluster indices as states. Textual summaries from the Forward LLM and NL-SSM are mapped to binary regimes using Goldstein scores (Goldstein, 1992; Gerner et al., 2009). We report Normalized Mutual Information (NMI) for geopolitical regime alignment, and sensitivity/F1 for CKD progression detection.

3 Results

3.1 Geopolitical Regime Discovery

We evaluate on the Global Database of Events, Language, and Tone (GDELT) dataset (Leetaru & Schrodt, 2013), a global database of timestamped news events. Fixed sentence embeddings (B1) and neural HMMs (B3) failed to discover meaningful regime structure ($\text{NMI} \leq 0.01$; see Table 1), clustering by geography rather than regime. TopicGPT achieved greater model alignment (NMI: 0.197), demonstrating that LLM-induced topic abstractions provide effective state descriptions. However, overlapping conflict and cooperation signals across topics suggest static assignments conflate mixed regimes.

Table 1: Normalized Mutual Information (NMI) between inferred states and ground-truth regimes.

Method	**NMI** $\uparrow$
B1: Fixed Embed + HMM	0.006
B2: TopicGPT + HMM	0.197
B3: Neural HMM	0.002
B4: Forward LLM	0.129
NL-SSM	**0.551**

NL-SSM discovers meaningful geopolitical regimes. NL-SSM achieved substantially higher alignment (NMI: 0.551), producing temporally discriminative state summaries. In contrast, the Forward LLM relied on hedging language (e.g., would, possibly, could, might), and frequently predicted persistent cooperation.

Temporal Dynamics of Learned States

We quantified temporal variation by computing cosine similarity between consecutive states using sentence-transformer embeddings. As shown in Fig. 1, the Forward LLM shows strong persistence

on GDELT (95% above 0.5; mean=0.71, std=0.12), whereas NL-SSM states vary more (42% above 0.5; mean=0.46, std=0.32; $p < 0.0001$). CKD shows the same trend: Forward LLM summaries remain nearly identical (100%; mean=0.93, std=0.06), while NL-SSM evolves across patient visits (71%; mean=0.68, std=0.36; $p < 0.0001$), suggesting future-conditioned supervision encourages temporally adaptive states.

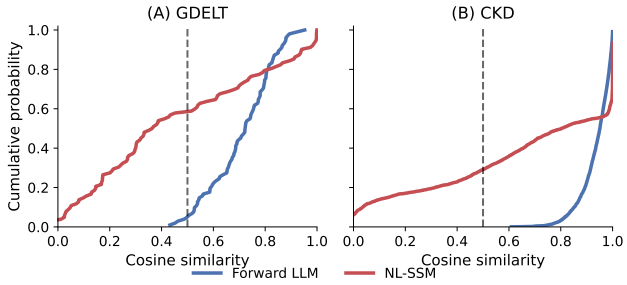

Figure 1: Cumulative distribution of cosine similarity between consecutive states on GDELT and CKD datasets.

3.2 Clinical State Inference (Nephrology)

Clinical content analysis

We evaluate NL-SSM on chronic kidney disease (CKD) prognosis (13.2% progression rate). We analyze clinical content with nephrology-informed categories; see Appendix A.8 and Table 3. NL-SSM summaries contain significantly more objective clinical evidence (lab values, stage mentions, treatments, comorbidities; $p < 0.0001$), whereas the Forward LLM relies more on speculative trajectory and prognosis language. This mirrors the persistence bias observed in GDELT, and suggests future-conditioned training yields physiologically grounded state summaries.

CKD Progression

NL-SSM achieves higher sensitivity (0.63 vs 0.38) and F1 (0.25 vs 0.21) than Forward LLM (Table 2), but both exhibit lower precision (15%), with false alarm rates of 52% (NL-SSM) and 34% (Forward LLM). This precision-sensitivity trade-off reflects the challenge of rare-event detection: improving recall for critical transitions comes at the cost of increased false alarms. A qualitative CKD progression example is shown in Appendix 2.

Table 2: CKD progression detection performance on all test samples ($N = 2567$). Missing predictions count as incorrect. 95% confidence intervals via patient-clustered bootstrap.

Method	F1	Sensitivity	Coverage
Forward LLM	0.21 [0.18, 0.25]	0.38 [0.33, 0.43]	81.6%
NL-SSM	**0.25 [0.22, 0.28]**	**0.63 [0.56, 0.69]**	**96.0%**

4 Discussion

Our results demonstrate that training with future context can reshape the learned state space. While embedding-based and topic-based probabilistic state-space model baselines cluster by static semantic features and the Forward LLM exhibits persistence bias, NL-SSM produces time-specific state descriptions that better capture temporal evolution. This highlights a fundamental tradeoff in persistent regimes: stability optimization achieves high accuracy but systematically misses critical transitions.

Limitations. Evaluating natural language state summaries is challenging. While projecting to categorical labels enables quantitative evaluation, it may not capture nuanced information. Second, NL-SSM requires a high-capacity teacher during training, increasing computational cost.

REFERENCES

Ahmed M Alaa and Mihaela van der Schaar. Attentive state-space modeling of disease progression. *Advances in neural information processing systems*, 32, 2019.

Masanao Aoki. *State space modeling of time series*. Springer Science & Business Media, 2013.

James Durbin and Siem Jan Koopman. *Time series analysis by state space methods*. Oxford university press, 2012.

Deborah J. Gerner, Philip A. Schrodt, Omur Yilmaz, and Erin Weddle. *Conflict and Mediation Event Observations (CAMEO) Manual*, 2009. URL `http://data.gdeltproject.org/documentation/CAMEO.Manual.1.1b3.pdf`. Version 1.1b3.

Joshua S Goldstein. A conflict-cooperation scale for weis events data. *Journal of conflict resolution*, 36(2):369–385, 1992.

James D Hamilton. State-space models. *Handbook of econometrics*, 4:3039–3080, 1994.

Kelly Hong, Anton Troynikov, and Jeff Huber. Context rot: How increasing input tokens impacts llm performance. *URL https://research. trychroma. com/context-rot, retrieved October*, 20:2025, 2025.

Matthew Johnson, David K Duvenaud, Alexander Wiltschko, Ryan P Adams, and Sandeep R Datta. Composing graphical models with neural networks for structured representations and fast inference. In *NeurIPS*, 2016.

Omar Khattab, Christopher Potts, and Matei Zaharia. Baleen: Robust multi-hop reasoning at scale via condensed retrieval. *Advances in Neural Information Processing Systems*, 34:27670–27682, 2021.

Rahul G Krishnan, Uri Shalit, and David Sontag. Deep kalman filters. *arXiv preprint arXiv:1511.05121*, 2015.

Rahul G Krishnan, Uri Shalit, and David Sontag. Structured inference networks for nonlinear state space models. In *AAAI*, 2017.

Kalev Leetaru and Philip A. Schrodt. Gdelt event database codebook v2.0, 2013. URL `https://data.gdeltproject.org/documentation/GDELT-Event_Codebook-V2.0.pdf`.

Divyat Mahajan, Sachin Goyal, Badr Youbi Idrissi, Mohammad Pezeshki, Ioannis Mitliagkas, David Lopez-Paz, and Kartik Ahuja. Beyond multi-token prediction: Pretraining llms with future summaries. *arXiv preprint arXiv:2510.14751*, 2025.

Chau Minh Pham, Alexander Hoyle, Simeng Sun, Philip Resnik, and Mohit Iyyer. Topicgpt: A prompt-based topic modeling framework. doi: 10.48550. *arXiv preprint arXiv.2311.01449*, 2024.

Jack W Rae, Anna Potapenko, Siddhant M Jayakumar, and Timothy P Lillicrap. Compressive transformers for long-range sequence modelling. *arXiv preprint arXiv:1911.05507*, 2019.

Bhaskar D Rao and KS Arun. Model based processing of signals: A state space approach. *Proceedings of the IEEE*, 80(2):283–309, 2002.

Aurko Roy, Mohammad Saffar, Ashish Vaswani, and David Grangier. Efficient content-based sparse attention with routing transformers. *Transactions of the Association for Computational Linguistics*, 9:53–68, 2021.

Andrew Sellergren, Sahar Kazemzadeh, Tiam Jaroensri, Atilla Kiraly, Madeleine Traverse, Timo Kohlberger, Shawn Xu, Fayaz Jamil, Cían Hughes, Charles Lau, et al. Medgemma technical report. *arXiv preprint arXiv:2507.05201*, 2025.

Alex L Zhang, Tim Kraska, and Omar Khattab. Recursive language models. *arXiv preprint arXiv:2512.24601*, 2025.

A Appendix

A.1 Full NL-SSM Algorithm

Algorithm 1 Natural Language State-Space Model (NL-SSM)

Require: Document sequence $\{x_t\}_{t=1}^{T}$
Ensure: Latent state sequence $\{s_t\}_{t=1}^{T}$

Backward label induction (teacher, offline)
$s_T \leftarrow \mathrm{LM}_{\text{teach}}(x_T)$
for $t = T-1$ **to** 1 **do**
    $s_t \leftarrow \mathrm{LM}_{\text{teach}}(x_t, s_{t+1})$
end for

Train state summarization model
for $t = 1$ **to** T **do**
    Train $\mathrm{LM}_{\text{summ}}$ on $(s_{t-1}, x_t) \rightarrow s_t$
end for

Train state transition model
for $t = 1$ **to** $T-1$ **do**
    Train $\mathrm{LM}_{\text{trans}}$ on $s_t \rightarrow s_{t+1}$
end for

Inference (causal filtering)
$\hat{s}_1 \leftarrow \mathrm{LM}_{\text{summ}}(x_1)$
for $t = 2$ **to** T **do**
    $\hat{s}_t \leftarrow \mathrm{LM}_{\text{summ}}(\hat{s}_{t-1}, x_t)$
end for

Forecasting
for $t = 1$ **to** $T-1$ **do**
    $\tilde{s}_{t+1} \leftarrow \mathrm{LM}_{\text{trans}}(\hat{s}_t)$
end for

A.2 Baselines

Baseline 1: Fixed Embeddings + HMM. We embed each document using a frozen `all-MiniLM-L6-v2` encoder, and form monthly observations averaging these embeddings. A Gaussian-emission HMM is then trained using Expectation–Maximization to maximize marginal likelihood. HMMs are trained with the expectation-maximization algorithm until convergence of the log-likelihood.

Baseline 2: TopicGPT + HMM. We adopt TopicGPT Pham et al. (2024), which uses LLMs to induce topics from text corpora. We use GPT-4o to produce a global topic list (see Appendix A.3) from 100 sampled training articles, assign topics to three randomly sampled articles per month per country pair, and train a categorical HMM. Topics are fixed across country-pair splits.

Baseline 3: Neural HMM. This baseline jointly trains a neural encoder and an HMM using Generalized EM. The encoder f_ϕ maps each document to a continuous embedding and is instantiated f_ϕ using `all-MiniLM-L6-v2` with only the final transformer layer fine-tuned.

Baseline 4: Forward LLM. We use the same high-capacity language model (GPT-4o) as a purely causal baseline. At each timestep, the model produces a state summary conditioned only on the previous state summary (if available) and the current observations (three randomly sampled articles). The model is prompted with the same task description as NL-SSM (see Appendix A.4), but is never trained or conditioned on future states. This baseline isolates the effect of future-conditioned training supervision while controlling for model capacity.

Baseline 5: Long Short-Term Memory (LSTM). We train an LSTM classifier to predict regime labels from `all-MiniLM-L6-v2` embeddings using a 6-month sliding window. Parameters are shared across country pairs.

Our Contribution: NL-SSM. We use GPT-4o as a teacher model during training to generate future-conditioned state summaries, and Phi-3-mini-128k as student models for state summarization, and state transition. During inference, NL-SSM operates strictly causally, and observes the same information as the Forward LLM baseline (i.e., only the previous timestep's inferred state and current observation), isolating the effect of future-conditioned training supervision (see Appendix A.5).

A.3 BASELINE 2: TOPICGPT

TopicGPT injects task-specific guidance by explicitly constraining the language model, GPT 4o, to produce relation-level abstractions (e.g., diplomatic, military, or economic interactions) rather than generic semantic topics. However, this guidance is static and corpus-level: topics are induced once from the training data, do not condition on temporal context, and do not incorporate future information. As a result, TopicGPT captures coarse relational semantics, but is not designed to model temporal regime transitions. We produce a fixed set of reusable, relation-level topics that characterize bilateral country relations across all country pairs and time periods.

Topics are induced using only documents from the training split to avoid information leakage. We first restrict the corpus to training examples and randomly sample 100 documents for topic induction using a fixed random seed to ensure reproducibility.

Prompt: The language model is prompted as follows:

> You are analyzing how relations between two countries evolve over time.
>
> **Context:**
> You are given a set of real-world news articles.
>
> **Task:**
> Produce 20 topics describing the bilateral relations between the countries during this time period.
>
> **Rules:**
>
> - Each topic must be a short label (no description)
> - Labels must be 1–3 words
> - Topics must be reusable across all country pairs
>
> **Return format:**
>
> ```
> topic_list:
> - topic1
> - topic2
> ...
> ```

The final global topic list consists of the following 20 relation-level categories:

- Military Cooperation
- Diplomatic Tensions
- Trade Relations
- Humanitarian Aid
- Political Protests
- Security Concerns
- Economic Sanctions
- Cultural Exchange
- Leadership Changes
- Peace Negotiations

- Territorial Disputes
- Election Influence
- Energy Partnerships
- Technological Collaboration
- Immigration Policies
- Defense Alliances
- Crisis Management
- Environmental Agreements
- Human Rights Issues
- Strategic Partnerships

A.4 Baseline 4: Forward Reasoning LLM Prompts (GDELT)

The Forward Reasoning LLM baseline is a strictly causal approach that mirrors the inference-time structure of NL-SSM, but without future-conditioned supervision. The model operates in two stages at each timestep: (1) inferring the current latent state using observed text, and (2) forecasting the next latent state without access to future observations.

Prompt 1: State Inference with Observations. At each timestep t, the language model infers the current state summary conditioned on the previous state summary (if available) and the current month's news articles.

> You are analyzing how relations between two countries evolve over time.
>
> **Task:**
> Based only on the prior month's summary (if available) and the current articles, write a concise 2–3 sentence summary describing the current bilateral relations between the countries.
>
> **Country pair:** *[A–B]*
> **Time period:** *[Month, Year]*
>
> **Prior state summary (from the previous month):**
> *[Previous state summary]*
> If no prior state is available, the model is explicitly informed that no prior summary exists.
>
> **Current articles:**
> *Article 1: [Text]*
> *Article 2: [Text]*
> *Article 3: [Text]*
>
> **State summary:**

Prompt 2: State Transition. To evaluate temporal prediction, the model is then prompted to forecast the next state summary using only the inferred state at time t, without access to any articles from time $t+1$.

> You are analyzing how relations between two countries evolve over time.
>
> **Task:**
> Based only on the current state summary, predict the bilateral relations at the next time period. Write a concise 2–3 sentence summary describing the expected state and trajectory.
>
> **Country pair:** *[A–B]*
> **Current time period:** *[Month t]*
> **Next time period:** *[Month $t+1$]*
>
> **Current state summary:**
> *[State summary at time t]*
>
> **Predicted next state summary:**

Decoding Details. All forward baseline prompts use greedy decoding with temperature set to 0 and a maximum response length of 1500 tokens. The model is queried strictly causally and never conditions on future observations.

A.5 NL-SSM Training and Inference Prompts

Train 1 (State Inference). Trains $(\texttt{teacher\_state}_{t-1}, x_t) \rightarrow \texttt{teacher\_state}_t$.

> **Task:** Using the prior month's summary (if available) and current articles, write a 2–3 sentence summary of current bilateral relations.
> **Country pair:** [A–B], Time: [Month t]
> There is no prior state summary *or*
> Prior state summary (from the previous month): [teacher_state$_{t-1}$]
> Articles: Article 1, Article 2, Article 3
> **State summary:**

Train 2 (State Transition). Trains $\texttt{teacher\_state}_t \rightarrow \texttt{teacher\_state}_{t+1}$.

> **Task:** Using only the current state summary, predict relations in the next time period.
> **Country pair:** [A–B], Time: [Month t]
> **Current state summary:** [teacher_state$_t$]
> **Next state summary:**

Inference. Two-pass decoding: (i) infer states using Train 1 prompt and observed articles; (ii) forecast next states using Train 2 prompt without observations.

A.6 Implementation Details.

Teacher Model (Backward Label Induction): GPT-4o via OpenAI API. Temperature=0, max tokens=1500.

Student Models: Phi-3-mini-128k fine-tuned with LoRA (r=16, alpha=32, dropout=0.05). Training: 3 epochs, learning rate=2e-4, effective batch size=16 (per-device=1, gradient accumulation=16).

Evaluation: Sentence-BERT (all-MiniLM-L6-v2) for computing state similarity. Goldstein score mapping for regime classification.

A.7 CKD Clinical Marker Lexicon

We analyze clinical content in predicted CKD states using the following marker categories. Terms with asterisks (*) use prefix matching (e.g., "declin" matches "decline," "declining," "declined").

- **Lab values:** egfr, creatinine, gfr, bun, potassium, phosphorus, hemoglobin
- **Stage mentions:** stage 1, stage 2, stage 3, stage 4, stage 5, esrd
- **Trajectory:** progress, declin*, stable, improv*, worsen*, slow, rapid
- **Treatment:** dialysis, transplant, medication, ace, arb, diuretic, epo
- **Comorbidities:** diabetes, hypertension, anemia, proteinuria, edema
- **Prognosis:** risk, prognosis, outlook, future, expect*, likely

Note: All matching is case-insensitive. Medical abbreviations: ace = ACE inhibitors, arb = angiotensin receptor blockers, epo = erythropoietin.

A.8 Clinical Content Analysis

Table 3: Clinical content in CKD state predictions (mean mentions per summary). $^{***}p < 0.001$.

Category	NL-SSM	Forward LLM
Lab values***	**2.08**	1.81
Stage mention***	**1.69**	0.95
Comorbidities***	**0.91**	0.76
Treatment ***	0.77	0.46
Trajectory***	1.81	**2.31**
Prognosis***	1.53	**2.42**

A.9 CKD Qualitative Example

Qualitative Example: CKD Progression (Stage 3b → Stage 4)

NL-SSM

State at time t (Input)

Stage 3 chronic kidney disease (CKD) with an estimated glomerular filtration rate (eGFR) of 37 mL/min/1.73 m^2...**expected to progress to stage 4 CKD**, with a **high risk of end-stage renal disease (ESRD)** in the future, **necessitating dialysis or kidney transplantation**.

Forecasted state at time $t+1$ (Output)

Stage 4 CKD with an estimated glomerular filtration rate (eGFR) of 25 mL/min/1.73 m^2, **primarily due to hypertensive nephrosclerosis**. The disease trajectory is progressive, with a high risk of advancing to end-stage renal disease (ESRD) in the future, **necessitating dialysis or kidney transplantation.**

Forward LLM

State at time t (Input)

Stage 3 chronic kidney disease (CKD), with an eGFR of 37 ml/min/1.73 m^2 and a creatinine level of 1.3...**indicates a slight decline in kidney function**, with worsening edema and shortness of breath potentially related to fluid overload, necessitating the initiation of diuretic therapy.

Forecasted state at time $t+1$ (Output)

Stage 3 CKD, with a potential slight decline in eGFR to around 35 ml/min/1.73 m^2, given the current trajectory of kidney function. The **prognosis will focus on continued management of hypertension and fluid overload**, with adjustments to diuretic therapy as needed.

Figure 2: Qualitative comparison of forecasts before a true CKD stage transition. Both models receive comparable patient summaries at time t (gray). NL-SSM correctly anticipates progression to Stage 4 at $t+1$ (green), whereas the Forward LLM predicts persistence at Stage 3 (red).