Forecasting CPI Inflation Components with Hierarchical Recurrent Neural Network

Advanced Analytics: New Methods and Applications for Macroeconomic Policy Conference

Bank of England, European Central Bank & King's College London

July 21, 2022

Jonathan Benchimol
Research Department
Bank of Israel

Coauthors: **Oren Barkan**, **Itamar Caspi**, **Eliya Cohen**, **Allon Hammer**, and **Noam Koenigstein**

Outline

- Consumer Price Index and Dataset Properties
- Recurrent Neural Networks (RNNs)
- Hierarchical Recurrent Neural Networks (HRNN)
- Evaluation and Results
- Policy Implications and Conclusion

CPI and Dataset Properties

Consumer Price Index

- The Consumer Price Index (CPI) measures the average change over time in the prices consumers pay for a basket of goods and services.
- The CPI quantifies the average cost of living in a given country by estimating the purchasing power of a single unit of currency.
- The CPI is the key macroeconomic indicator for measuring inflation (or deflation).

US Consumer Price Index

- In the US, the CPI is calculated and reported by the Bureau of Labor Statistics (BLS) on a monthly basis.
- The BLS has classified all expenditure items into more than 200 categories, arranged into eight major groups: (1) Housing, (2) Food and Beverages, (3) Medical Care, (4) Apparel, (5) Transportation, (6) Energy, (7) Recreation, and (8) Services.
- The consumer goods and services are grouped in a **hierarchy** of increasingly detailed categories (levels).

Hierarchical Data Structure

- **Level 0**
  Aggregated CPI across all components

- **Level 1**
  Aggregated components (e.g., Energy, Apparel)

- **Mid levels (2-5)**
  Fine grained components, expenditure classes, item strata (e.g., Insurance)

- **Lower levels (6-8)**
  Finer grained components (e.g., Bacon, Tomatoes)

Hierarchical Data Structure

Level	Component	2018-01-01	2018-02-01	2018-03-01
0	All items	249.245	249.619	249.462
1	Food	252.070	252.061	252.388
2	Food at home	239.325	238.837	239.188
3	Cereals and bakery products	271.600	271.270	272.296
4	Cereals and cereal products	226.091	226.258	226.365
5	Flour and prepared flour mixes	236.060	239.483	237.640
5	Breakfast cereal	222.500	219.211	220.063
5	Rice, pasta, cornmeal	232.821	235.754	235.062
6	Rice	160.832	160.229	159.276
5	Bakery products	298.582	297.046	298.962
6	Bread	177.242	176.251	179.796
7	White bread	319.327	317.387	324.984
7	Bread other than white	345.247	342.862	347.639
6	Fresh biscuits, rolls, muffins	176.873	176.527	177.557
6	Cakes, cupcakes, and cookies	286.711	288.557	286.966
7	Cookies(6)	271.231	273.215	270.947
7	Fresh cakes and cupcakes	306.506	306.144	308.332
6	Other bakery products	264.747	263.396	263.411
7	Fresh sweetrolls, coffeecakes, doughnuts	294.893	297.587	298.936
7	Crackers, bread, and cracker products	301.937	298.492	299.961

Example

- The White Bread entry is classified under the following eight level hierarchy:
- All Items
- Food and Beverages
- Food at Home
- Cereals and Bakery Products
- Cereals and Cereal Products
- Bakery Products
- Bread
- White Bread

Forecasting CPI

- **Central banks** conduct monetary policy to achieve price stability (low and stable inflation).

- **Investors** in fixed income assets (such as government bonds) estimate future inflation to foresee upcoming trends in discounted real returns.

- **Government and private debt management** depend on the expected path of inflation.

- **Policymakers and marketmakers** monitor CPI component levels (e.g., core inflation, oil-related products).

Related Work

- Most related work deal with predicting the **headline CPI** only.
- Forecasts based on simple averages of past inflation are more accurate than structural models [1].
- ML models based on exogenous features: online prices, house prices, exchange rates etc. [2].
- Feed-forward NN to predict inflation rate in 28 OECD countries.
  About 50% of the countries NN were superior to autoregressive models [3].

[1] Makridakis, Assimakopoulos, Spiliotis. Objectivity, reproducibility and replicability in forecasting research. *International Journal of Forecasting* (2018).
[2] Medeiros, Vasconcelos, Veiga, Zilberman. Forecasting inflation in a data-rich environment: the benefits of machine learning methods. *Journal of Business & Economic Statistics* (2021).
[3] Choudhary, Haider. Neural network models for inflation forecasting: an appraisal. *Applied Economics* (2012).

Objective

- **<u>Our goal</u>**: Forecast US monthly CPI inflation for **all components**, without exogenous features.

- Harness the **hierarchical** pattern of the data to improve prediction at low levels.

- Utilize the **sequential** pattern of the data employing **Recurrent Neural Networks**.

- Improve predictions of **volatile** and non-stationary time series at lower-level components.

Dataset

- CPI-U (Urban CPI) from 1994 to 2019 from the BLS.
- Monthly prices of 424 components, structured hierarchically.
- Each component is a time series of inflation rates belonging to a level between 0 and 8.
- The **train set** comprises 70% of early entries, and the other 30% comprise the **test set**.

$$p_t = \text{CPI-U at time } t$$

$$rate_t = 100 * log(\frac{p_t}{p_{t-1}})$$

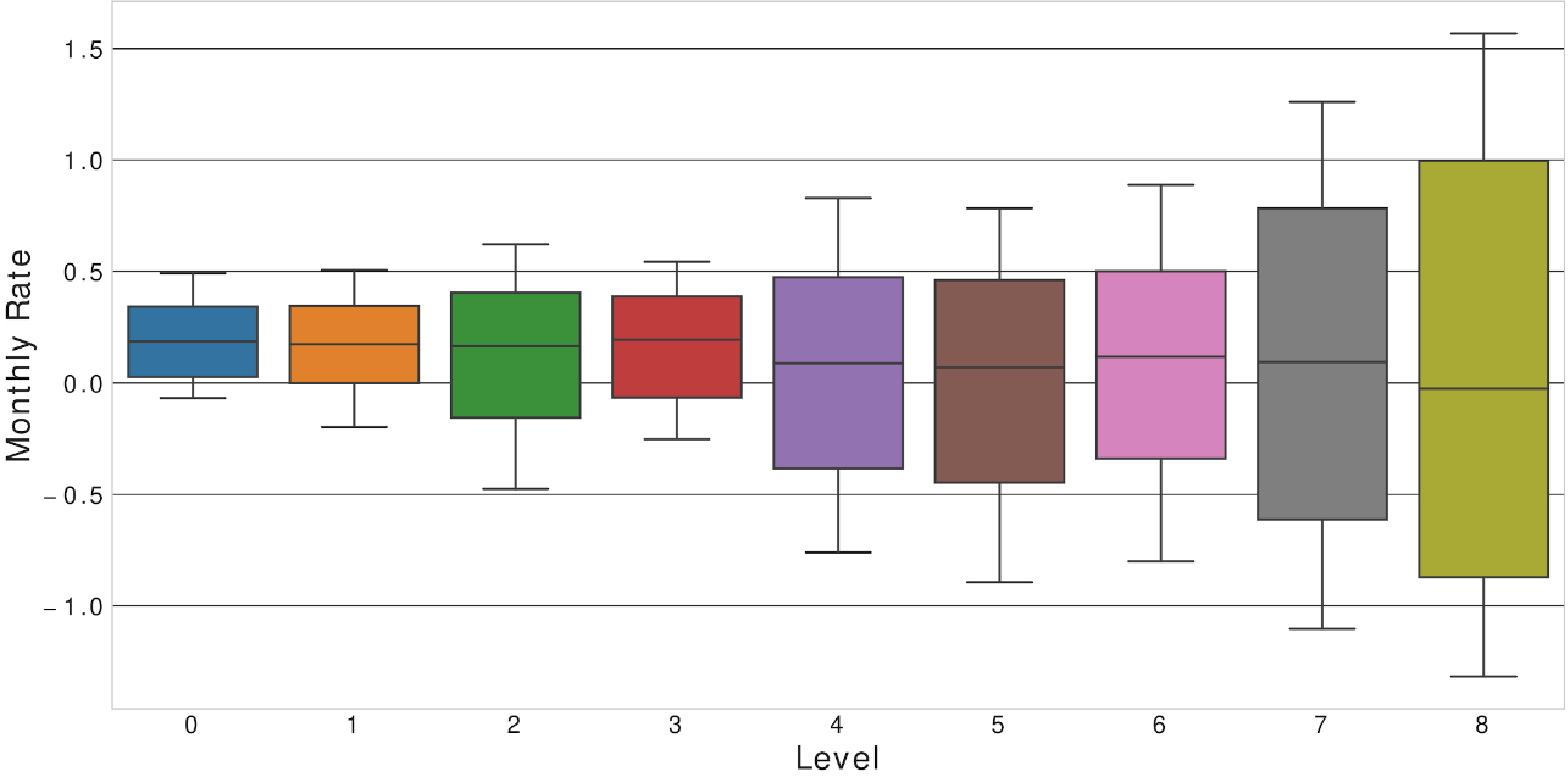
Summary Statistics

Table 1: Descriptive Statistics

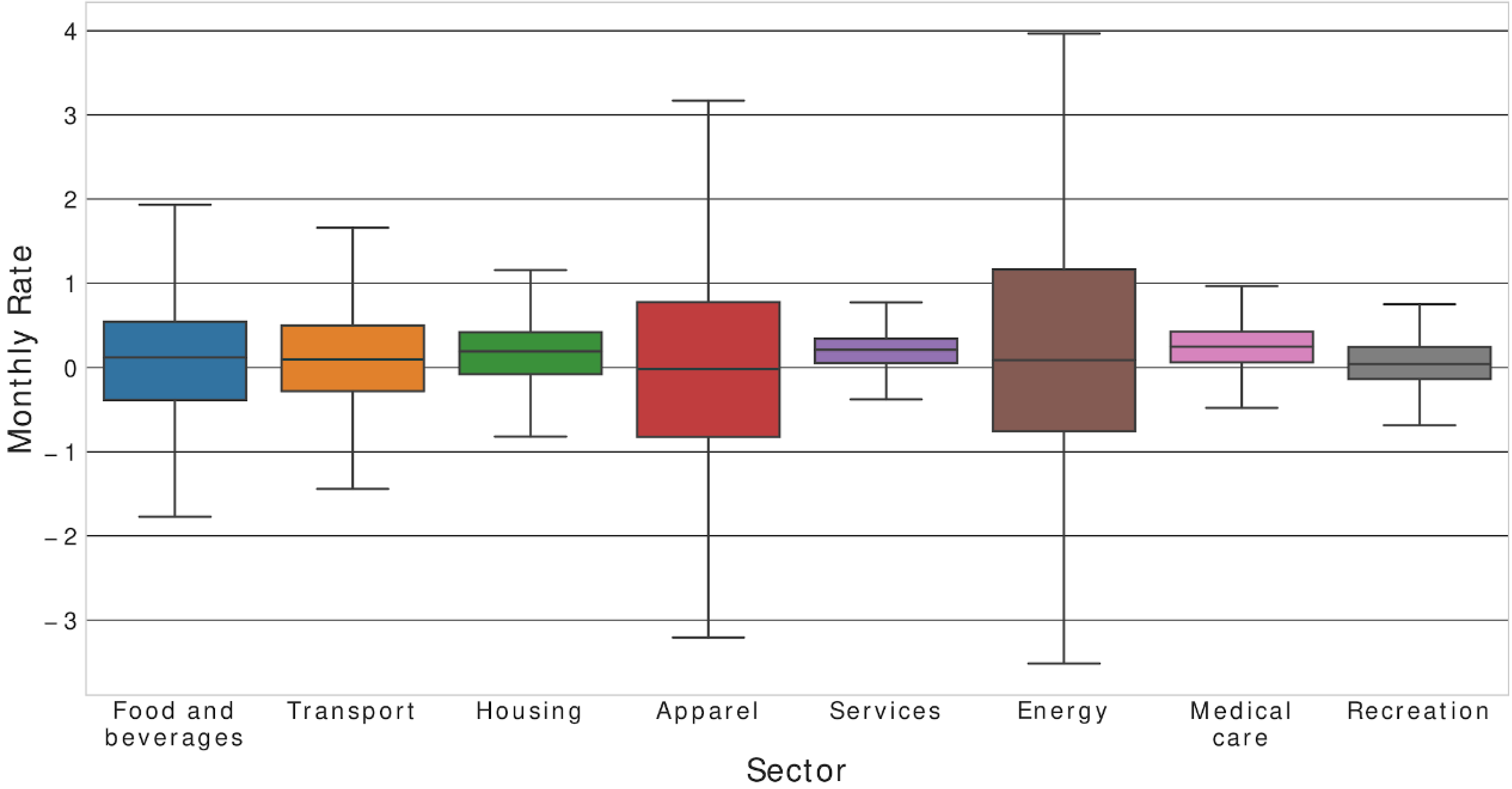
Data set	# Monthly Measurements	Mean	STD	Min	Max	# of Indexes	Avg. Measurements per Index
Headline Only	303	0.18	0.33	-1.93	1.22	1	303
Level 1	6742	0.17	0.96	-18.61	11.32	34	198.29
Level 2	6879	0.12	1.10	-19.60	16.81	46	149.54
Level 3	7885	0.17	1.31	-34.23	16.37	51	121.31
Level 4	7403	0.08	1.97	-35.00	28.17	58	107.89
Level 5	10809	0.01	1.43	-21.04	242.50	92	87.90
Level 6	7752	0.09	1.49	-11.71	16.52	85	86.13
Level 7	4037	0.11	1.53	-11.90	9.45	50	80.74
Level 8	595	0.08	1.56	-5.27	5.02	7	85.00
Full Hierarchy	52405	0.10	1.75	-35.00	242.50	424	123.31

Notes: General statistics of the headline CPI and CPI-U for each level in the hierarchy and the full hierarchy of indexes.

Volatility at Different Levels



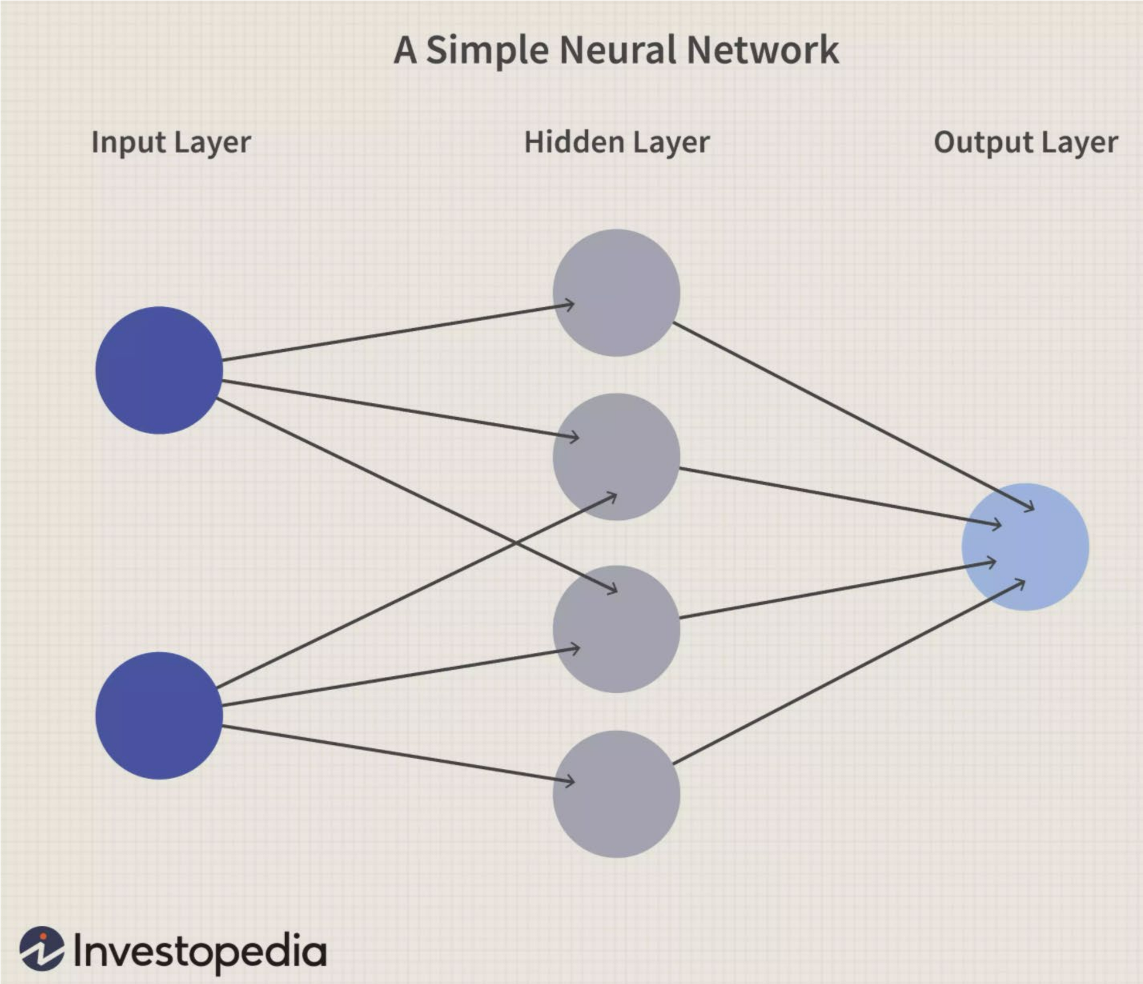
Volatility at Different Sectors



Recurrent Neural Networks

Artificial Neural Networks

A neural network is a group of algorithms that endeavors to recognize underlying relationships in a set of data through a process that mimics the way the human brain operates.

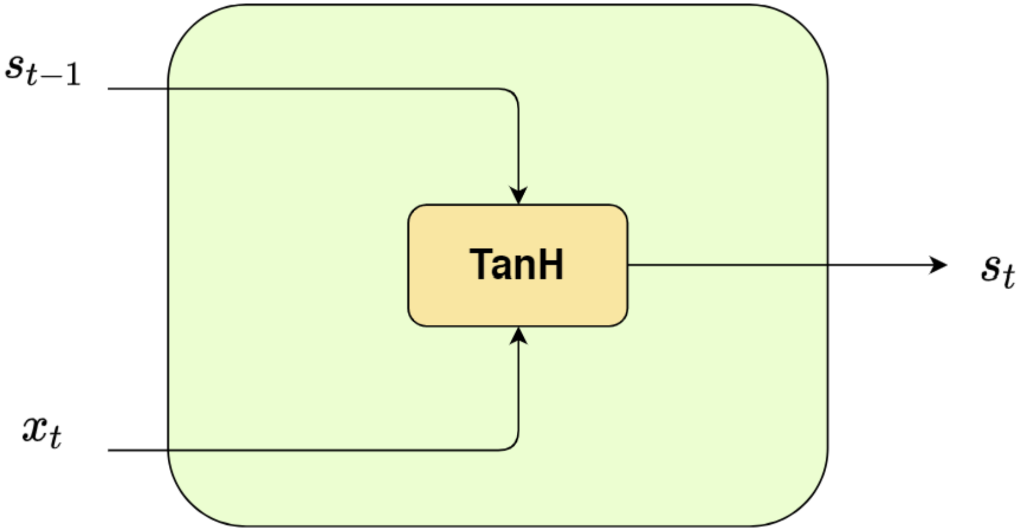


Recurrent Neural Networks

- RNNs are neural networks that model sequences of data in which each value is assumed to be dependent on previous values.
- RNNs are feed-forward networks augmented by including a feedback loop.
- RNNs introduce a notion of time to the standard feed-forward neural networks and excel at modeling temporal dynamic behavior (Chung et al., 2014).
- Some RNN units retain an internal memory state from previous time steps representing an arbitrarily long context window.
- Our paper covers the three most popular units: **Basic RNN**, Long-Short Time Memory (**LSTM**), and Gated Recurrent Unit (**GRU**).

Basic RNN

Let $\{x_t\}_{t=1}^{T}$ be the model's input time series consisting of T samples. Similarly, let $\{s_t\}_{t=1}^{T}$ be the model's out time series consisting of T samples. Namely, at timestamp t the model's input is x_t and its output (prediction) is s_t.



$$s_t = \tanh\left(x_t u + s_{t-1} w + b\right)$$

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

The linear combination is the argument of a hyperbolic tangent *activation* function allowing the unit to model nonlinear relations between inputs and outputs.

Long-Short Term Memory Networks

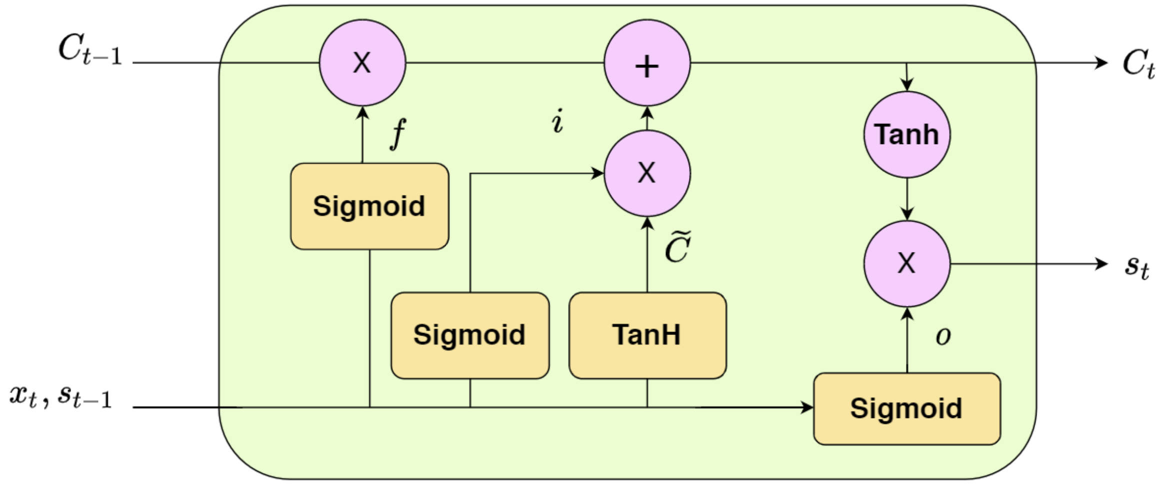
- Basic RNNs suffer from the “short-term memory” problem

Use recent history to forecast, but for long enough sequences, cannot carry relevant information from earlier to later periods, e.g., relevant patterns from the same month in previous years.

- Long-Short Term Memory networks (LSTMs) deal with this problem by introducing **gates** that enable the preservation of relevant “long-term memory” and combining it with the most recent data.
- The introduction of LSTMs paved the way for significant strides in various fields such as NLP, speech recognition, robot control and more.

Long-Short Term Memory Networks

- A LSTM unit has the ability to "memorize" or "forget" information through the use of a special memory **cell state**.
- The cell state is carefully regulated by three gates that **control** the flow of information in the LSTM unit: input gate (*i*), forget gate (*f*), and output gate (*o*).
- The cell state *C* is updated by a combination of its previous state and its current candidate $\tilde{C}$.



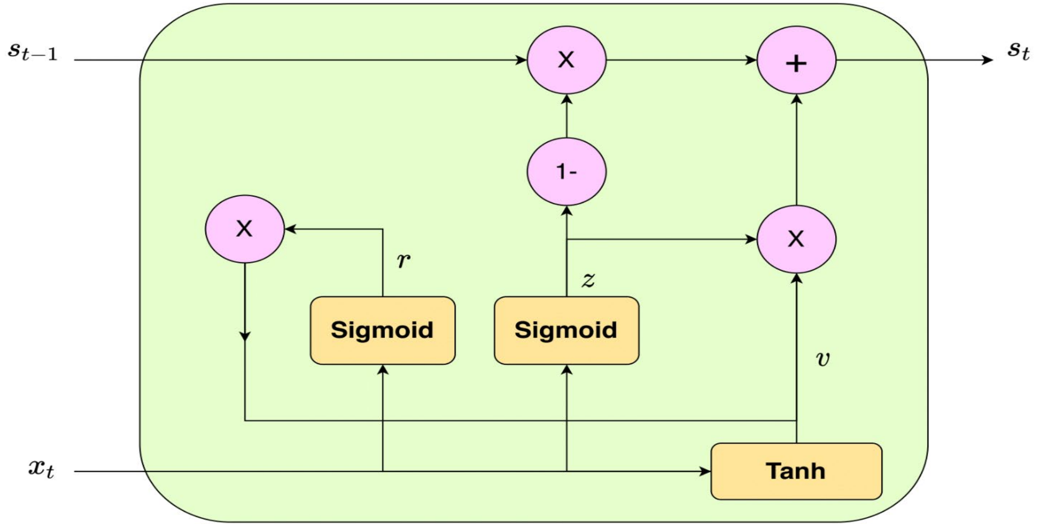
$$
\begin{aligned}
i &= \sigma(x_t u^i + s_{t-1} w^i + b^i), \\
f &= \sigma(x_t u^f + s_{t-1} w^f + b^f), \\
o &= \sigma(x_t u^o + s_{t-1} w^o + b^o), \\
\tilde{c} &= \tanh(x_t u^c + s_{t-1} w^c + b^c), \\
c_t &= f * c_{t-1} + i * \tilde{c}, \\
s_t &= o * \tanh(c_t),
\end{aligned}
$$

Learned params

$$\sigma(x) = \frac{1}{1+e^{-x}}$$

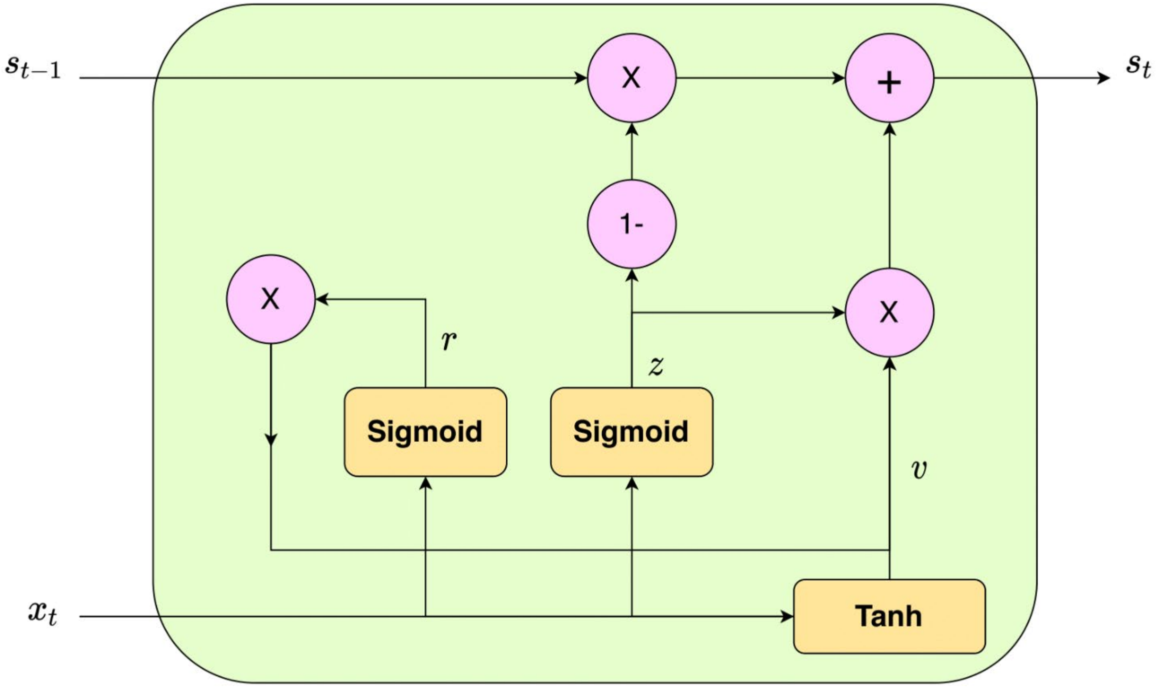
Gated Recurrent Unit

- A Gated Recurrent Unit (GRU) is a newer improvement of the LSTM unit that dropped the cell state in favor of a more simplified unit that requires **less learnable parameters**.
- GRUs are **faster** and **more efficient** especially when training data is limited, such as in the case of inflation predictions (and especially disaggregated inflation components).



Gated Recurrent Unit

- The candidate activation v is a function of the input and the previous output.
- The output s is a combination of the candidate activation v and the previous output controlled by z.



$$
\begin{aligned}
z &= \sigma(x_t u^z + s_{t-1} w^z + b^z), \\
r &= \sigma(x_t u^r + s_{t-1} w^r + b^r), \\
v &= \tanh(x_t u^v + (s_{t-1} * r) w^v + b^v), \\
s_t &= z * v + (1 - z) * s_{t-1},
\end{aligned}
$$

Hierarchical Recurrent Neural Networks

Hierarchical Recurrent Neural Networks

- HRNN exhibits a network graph that follows the CPI hierarchy.
- Each node is a RNN that models the inflation rate of a specific component in the full CPI hierarchy.
- HRNNs propagate information from RNN models in **higher** levels to **lower** levels via hierarchical priors over the RNNs' learned weights.
- Expected result: Better predictions for lower-level components.

HRNN Formulation

- Define a parametric function *g* representing a RNN node in the hierarchy.

- *g* predicts a scalar value for the next input value of a series.

- Assuming a normal likelihood relation between *g* and the observed time series.

x_n^t = Inflation rate at time *t* of node *n*

T_n = Last time step of node *n*

τ_n = Precision variable of node *n*

θ_n = RNN learned params of node *n*

$$X_n^t \triangleq (x_n^1, ..., x_n^t)$$

$$p(X_n^{T_n} | \theta_n, \tau_n) = \prod_{t=1}^{T_n} p(x_n^t | X_n^{t-1}, \theta_n) = \prod_{t=1}^{T_n} \mathcal{N}(x_n^t; g(\theta_n, X_n^{t-1}), \tau_n^{-1})$$

HRNN Formulation

- Define hierarchical network of normal priors over the nodes' parameters:

$$p(\theta_n|\theta_{\pi_n}, \tau_{\theta_n}) = \mathcal{N}(\theta_n; \theta_{\pi_n}, \tau_{\theta_n}^{-1}\mathbf{I})$$

- This models the relationship between a node's parameters and its **parent** in the hierarchy.
- This relationship grows stronger according to the **correlation** between the two series.
- It ensures that each **node** is kept close to its parent, in terms of squared Euclidean distance in the parameter space.

θ_n = RNN learnt params of node *n*

θ_{π_n} = RNN learnt params of node *n's* parent

$C_n = \rho(X_n^{T_n}, X_{\pi_n}^{T_{\pi n}})$ = Pearson corr. coefficient between the parent and the child's time series

$\tau_{\theta_n} = e^{\alpha + C_n}$

= Precision parameter induced by the Pearson corr. and an additional hyperparameter α.

HRNN Formulation

- According to the Bayes Rule, the posterior probability is:

$$p(\theta|\mathrm{X},\mathrm{T}) = \frac{p(\mathrm{X}|\theta,\mathrm{T})p(\theta)}{P(X)} \propto$$

$$\prod_{n\in\mathcal{I}}\prod_{t=1}^{T_n}\mathcal{N}(x_n^t; g(\theta_n, X_n^{t-1}), \tau_n^{-1})\prod_{n\in\mathcal{I}}\mathcal{N}(\theta_n;\theta_{\pi_n}, \tau_{\theta_n}^{-1}\mathbf{I})$$

- Maximum A-Posteriori (MAP) approach:

$$\theta^* = \underset{\theta}{\operatorname{argmax}} \log p(\theta|\mathrm{X},\mathrm{T})$$

$\mathcal{I} = \{n\}_{n=1}^{N}$ = Enumeration of all nodes from all levels

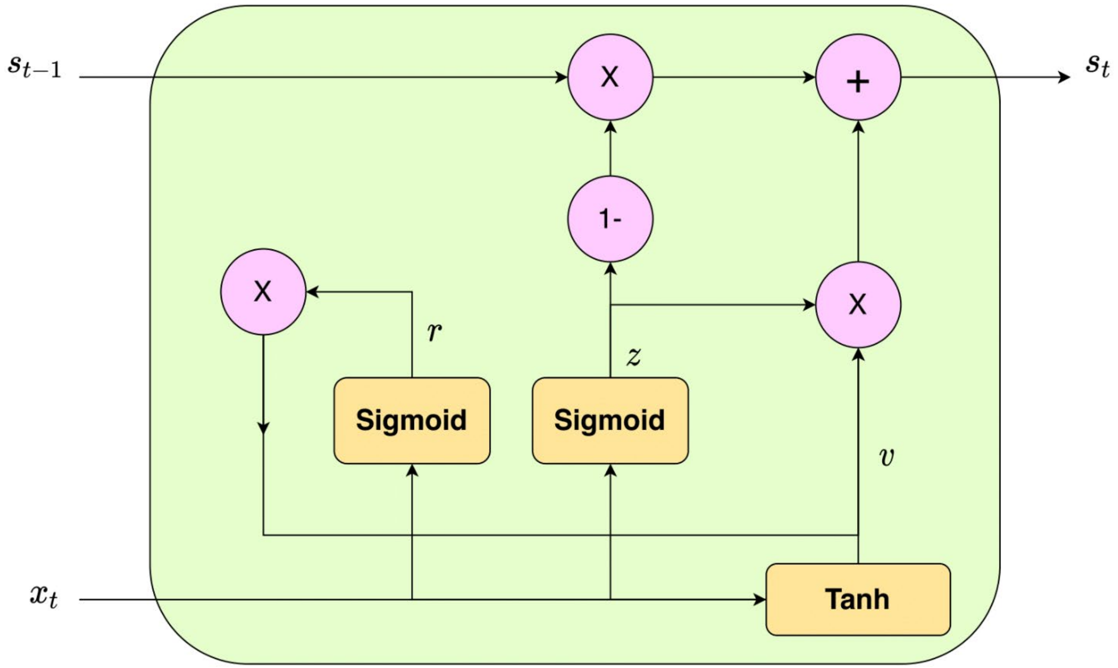
$X = \{X_n^{T_n}\}_{n\in\mathcal{I}}$ = Aggregation of all series from all levels

$\theta = \{\theta_n\}_{n\in\mathcal{I}}$ = Aggregation of all learnt params from all levels

$\mathrm{T} = \{\tau_n\}_{n\in\mathcal{I}}$ = Aggregation of all precision params from all levels

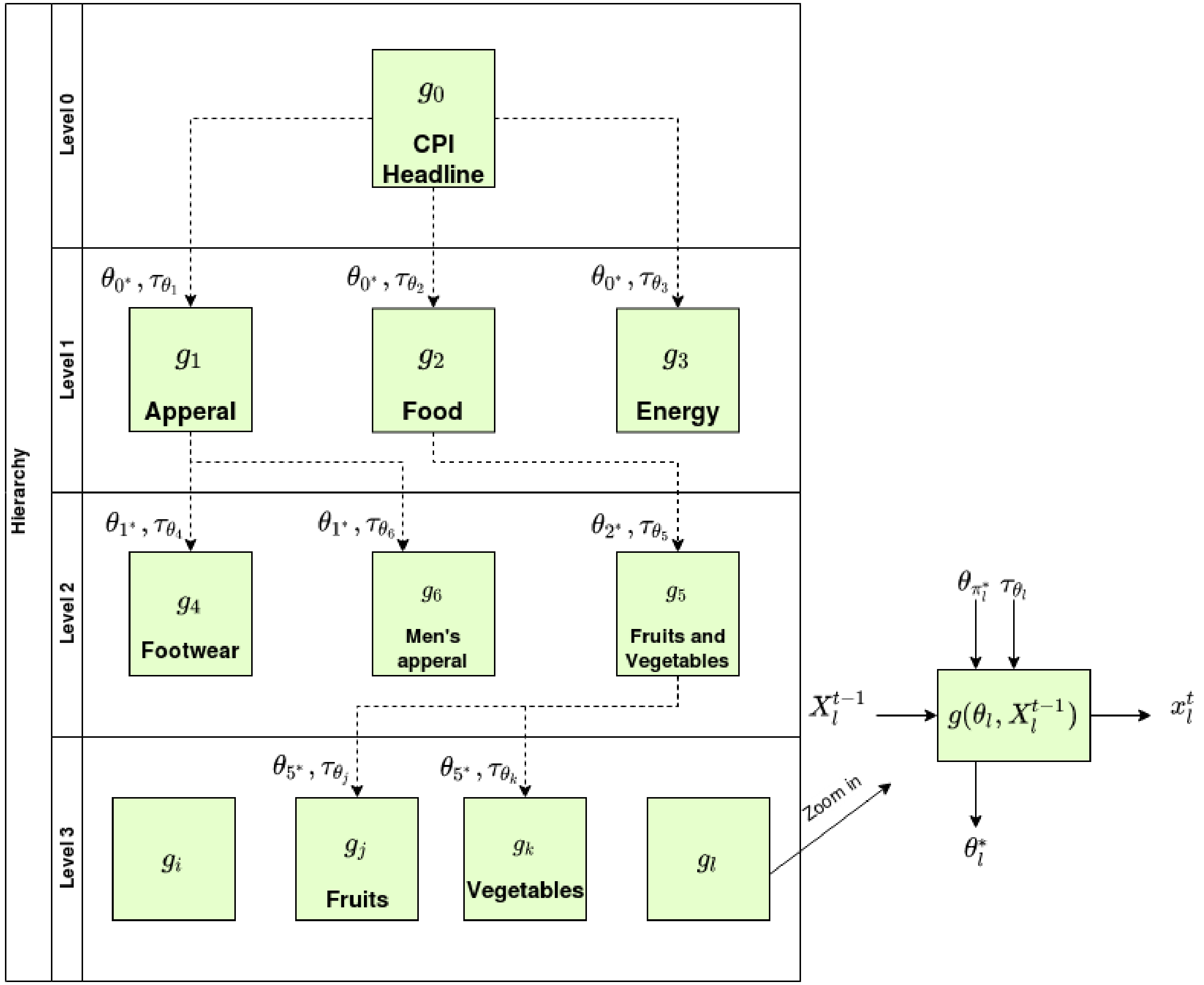
HRNN Based on GRUs

- HRNNs implement g as a scalar GRU.
- Specifically, each node n, is associated with a GRU of its own.
- HRNNs optimization proceeds with stochastic gradient ascent over the objective in MAP.



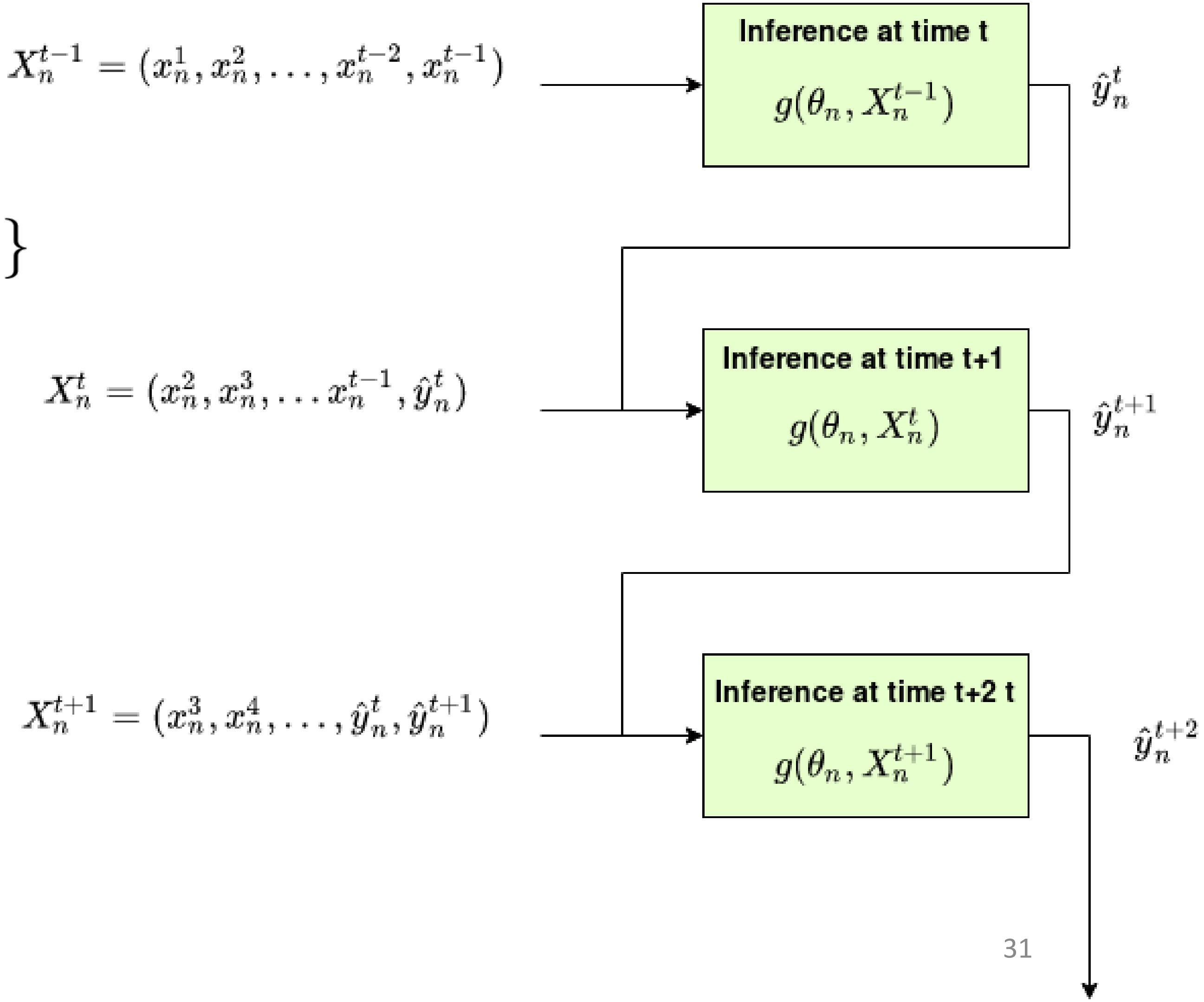
HRNN Architecture

- Each node is a **scalar GRU** predicting the inflation in the next time step for the given component.
- **Constraints** from the parent node are propagated down to the child node.
- GRUs are **trained** from top to bottom.



HRNN Inference

- Equipped with trained weights θ_n for node n.
- Predict for future time step $h \in \{0, .., 8\}$
- The prediction for horizon h is obtained by the predictions of previous horizons $h' < h$ iteratively.
- Each time using the previous predicted value $\hat{y}_n^{t+h'}$ as input to the GRU.



Evaluation and Results

Evaluation Metrics

- Evaluation metrics:
  - **RMSE:** $\sqrt{\frac{1}{T}\sum_{t=1}^{T}(x_t - \hat{x}_t)^2}$
  - **Pearson Correlation:** $\frac{COV(X_T, \hat{X}_T)}{\sigma_X \times \sigma_{\hat{X}}}$
  - **Distance Correlation:** $\frac{\mathrm{dCov}(X_T, \hat{X}_T)}{\sqrt{\mathrm{dVar}(X_T) \times \mathrm{dVar}(\hat{X}_T)}}$

$y_t, \hat{y}_t$ = Actual and predicted inflation rate at month *t*, respectively

$Y_T, \hat{Y}_T$ = Actual and predicted inflation rate series, respectively.

Baselines

- **Autoregressive (AR)**: Estimated next month value based on previous *d* months.

$$\hat{y}_t = \left(\sum_{i=1}^{d} x_{t-i} * \alpha_i\right) + \varepsilon_t$$

- **Phillips Curve (PC):** Add unemployment rate *u* which should have an inverse relation with inflation.

$$\hat{y}_t = \left(\sum_{i=1}^{d} x_{t-i} * \alpha_i\right) + \beta * u_{t-1} + \varepsilon_t$$

- **Random Walk (RW):** Simple average of last *d* months.

$$\hat{y}_t = \frac{1}{d} \sum_{i=1}^{d} x_{t-i} + \varepsilon_t$$

- **Auto-Regression in Gap Form (AR-GAP):** Detrend time series using RW, then use AR to predict the gap form and finally add the trend to final prediction.

$$\tau_t = \frac{1}{d} \sum_{i=1}^{d} x_{t-i}$$
$$g_t = x_t - \tau_t$$
$$g_{t+h} = \alpha_0 + \left(\sum_{i=1}^{d} g_{t+h-i} * \alpha_i\right) + \varepsilon_{t,h}$$
$$\hat{y}_{t+h} = g_{t+h} + \tau_t$$

- **Vector Autoregression (VAR):** Learn *K* most similar time series together.
- **Logistic Smooth Transition Auto Regressive Model (LSTAR):** extension of AR that allows for changes in the model parameters according to a transition variable *F* (Van Dijk et al. 2000).

Ablation Study

- **Single Scalar GRU**: One scalar GRU for all components. Assumes that the different components of CPI hierarchy behave similarly.
- **HRNN without hierarchy:** Set $\tau_{\theta_n} = 0 \ \forall n \in \mathcal{I}$, removing the hierarchical priors. Equivalent to *N* independent GRU units.
- **Fully Connected Neural Network (FC):** Similar to Auto-Regression but with non-linearities.
- **Vectorial GRU based on K Nearest Neighbors:** Different GRU for each node *n,* but each entry is a vector that includes the time series of its *k* nearest (most correlated) series.

Table 2: Average Results on Disaggregated CPI Components

Model Name	RMSE per horizon AR(1)=1.00						Correlation (at horizon=0)	
	0	1	2	3	4	8	Pearson	Distance
AR(1)	1.00	1.00	1.00	1.00	1.00	1.00	0.06	0.05
AR(2)	1.00	1.00	1.00	1.00	1.00	1.00	0.08	0.06
AR(3)	1.00	1.00	1.00	1.00	1.00	1.00	0.08	0.06
AR(4)	1.00	1.00	1.00	1.00	1.00	1.00	0.09	0.07
AR-GAP(3)	1.00	1.00	1.00	1.00	1.00	1.00	0.08	0.06
AR-GAP(4)	1.00	1.00	1.00	1.00	1.00	1.00	0.09	0.07
RW(4)	1.00	1.00	1.00	1.00	1.00	1.00	-0.05	0.04
Phillips(4)	1.00	1.00	1.00	1.00	0.98	1.00	-0.06	0.04
VAR(1)	1.03	1.03	1.04	1.03	1.04	1.05	0.04	0.03
VAR(2)	1.03	1.03	1.04	1.03	1.04	1.05	0.06	0.03
VAR(3)	1.03	1.03	1.03	1.03	1.04	1.05	0.06	0.03
VAR(4)	1.02	1.03	1.03	1.03	1.03	1.04	0.07	0.04
LSTAR($\rho = 4, c = 2, \gamma = 0.3$)	1.04	1.07	1.07	1.07	1.08	1.1	0.09	0.07
GBT(4)	0.83	0.83	0.83	0.84	0.84	0.86	0.18	0.27
RF(4)	0.84	0.85	0.86	0.86	0.86	0.87	0.19	0.29
FC(4)	1.03	1.03	1.04	1.04	1.04	1.05	0.12	0.09
Deep-NN(4)	0.90	0.90	0.90	0.90	0.91	0.91	0.13	0.22
Deep-NN(4) + Unemployment	0.85	0.85	0.85	0.85	0.85	0.86	0.12	0.22
S-GRU(4)	1.02	1.06	1.06	1.07	1.04	1.12	0.10	0.08
I-GRU(4)	0.83	0.84	0.85	0.85	0.86	0.89	0.17	0.13
KNN-GRU(1)	0.91	0.93	0.96	0.97	0.96	0.96	0.19	0.15
KNN-GRU(2)	0.90	0.93	0.95	0.97	0.96	0.96	0.20	0.15
KNN-GRU(3)	0.89	0.92	0.95	0.96	0.96	0.95	0.20	0.15
KNN-GRU(4)	0.89	0.91	0.95	0.95	0.95	0.95	0.20	0.15
HRNN(1)	0.79	0.79	0.81	0.81	0.81	0.83	0.23	0.28
HRNN(2)	0.78	0.79	0.81	0.81	0.80	0.82	0.22	0.29
HRNN(3)	0.79	0.78	0.80	0.81	0.81	0.81	0.23	0.30
HRNN(4)	0.78	0.78	0.79	0.79	0.79	0.80	0.24	0.29

Notes: Average results across all 424 inflation indexes that make up the headline CPI. The RMSE results are relative to the $AR(1)$ model and normalized according to its results, i.e., $\frac{RMSE_{Model}}{RMSE_{AR(1)}}$. Results are statistically significant according to Diebold-Mariano test with $p < 0.02$.

Table 3: CPI Headline Only

Model Name*	RMSE per horizon AR(1)=1.00						Correlation (at horizon=0)	
	0	1	2	3	4	8	Pearson	Distance
AR(1)	1.00	1.00	1.00	1.00	1.00	1.00	0.29	0.22
AR(2)	1.00	0.97	0.99	1.01	1.00	0.98	0.32	0.24
AR(3)	1.00	0.98	0.98	1.00	0.96	0.97	0.33	0.25
AR(4)	1.00	0.95	0.95	0.96	0.93	0.96	0.33	0.25
AR-GAP(3)	1.00	0.98	0.98	1.00	0.96	0.97	0.33	0.25
AR-GAP(4)	0.99	0.95	0.95	0.96	0.93	0.96	0.33	0.25
RW(4)	1.05	0.98	0.99	1.01	0.97	0.96	0.23	0.2
Phillips(4)	0.93	0.94	0.95	0.95	0.93	0.95	0.33	0.25
LSTAR($\rho = 4, c = 2, \gamma = 0.3$)	0.98	0.95	0.95	0.97	0.95	0.95	0.32	0.24
RF(4)	1.05	1.06	1.03	1.07	1.04	1.03	0.27	0.28
GBT(4)	0.97	0.99	0.93	0.95	0.93	0.93	0.25	0.35
FC(4)	**0.92**	**0.94**	0.94	0.96	0.93	0.94	0.33	0.25
Deep-NN(4)	0.94	0.97	0.96	0.98	0.94	0.92	0.31	0.32
Deep-NN(4) + Unemployment	1.00	0.97	**0.92**	**0.94**	**0.92**	**0.91**	**0.37**	0.32
HRNN(4) / GRU(4)	1.00	0.97	0.99	0.99	0.96	0.99	0.35	**0.37**

Notes: Prediction results for the CPI headline index alone. The RMSE results are relative to the $AR(1)$ model and normalized according to its results, i.e., $\frac{RMSE_{Model}}{RMSE_{AR(1)}}$.

No hierarchy prediction.
No advantage for GRU compared to simple AR model.

Results

Average Results of Best HRNN Model on Disaggregated CPI Components by **Hierarchy**

Table 4: HRNN(4) vs. I-GRU(4) at different levels of the CPI hierarchy with respect to AR(1)

Hierarchy Level	HRNN(4)						I-GRU(4)					
	RMSE per horizon AR(1)=1.00				Correlation (at horizon=0)		RMSE per horizon AR(1)=1.00				Correlation (at horizon=0)	
	0	2	4	8	Pearson	Distance	0	2	4	8	Pearson	Distance
Level 1	0.95	0.97	0.99	1.00	0.33	0.37	0.98	0.98	0.99	0.97	0.25	0.38
Level 2	0.91	0.90	0.91	0.91	0.30	0.35	0.90	092	0.94	0.93	0.26	0.34
Level 3	0.79	0.79	0.80	0.81	0.21	0.31	0.82	0.89	0.94	0.94	0.23	0.37
Level 4	0.77	0.77	0.76	0.77	0.26	0.32	0.84	0.87	0.90	0.92	0.20	0.33
Level 5	0.79	0.77	0.77	0.80	0.21	0.31	0.85	0.89	0.89	0.93	0.22	0.29
Level 6	0.75	0.76	0.81	0.81	0.19	0.23	0.85	0.89	0.90	0.92	0.21	0.21
Level 7	0.75	0.78	0.77	0.80	0.17	0.17	0.87	0.89	0.92	0.94	0.18	0.15
Level 8	0.72	0.78	0.77	0.78	0.10	0.23	0.89	0.90	0.92	0.94	0.10	0.12

Notes: The RMSE results are relative to the $AR(1)$ model and normalized according to its results, i.e., $\frac{RMSE_{Model}}{RMSE_{AR(1)}}$.

HRNN shows best performance in the lower levels, where CPI components are more volatile.

Results

Average Results of Best HRNN Model on Disaggregated CPI Components by **Sector**

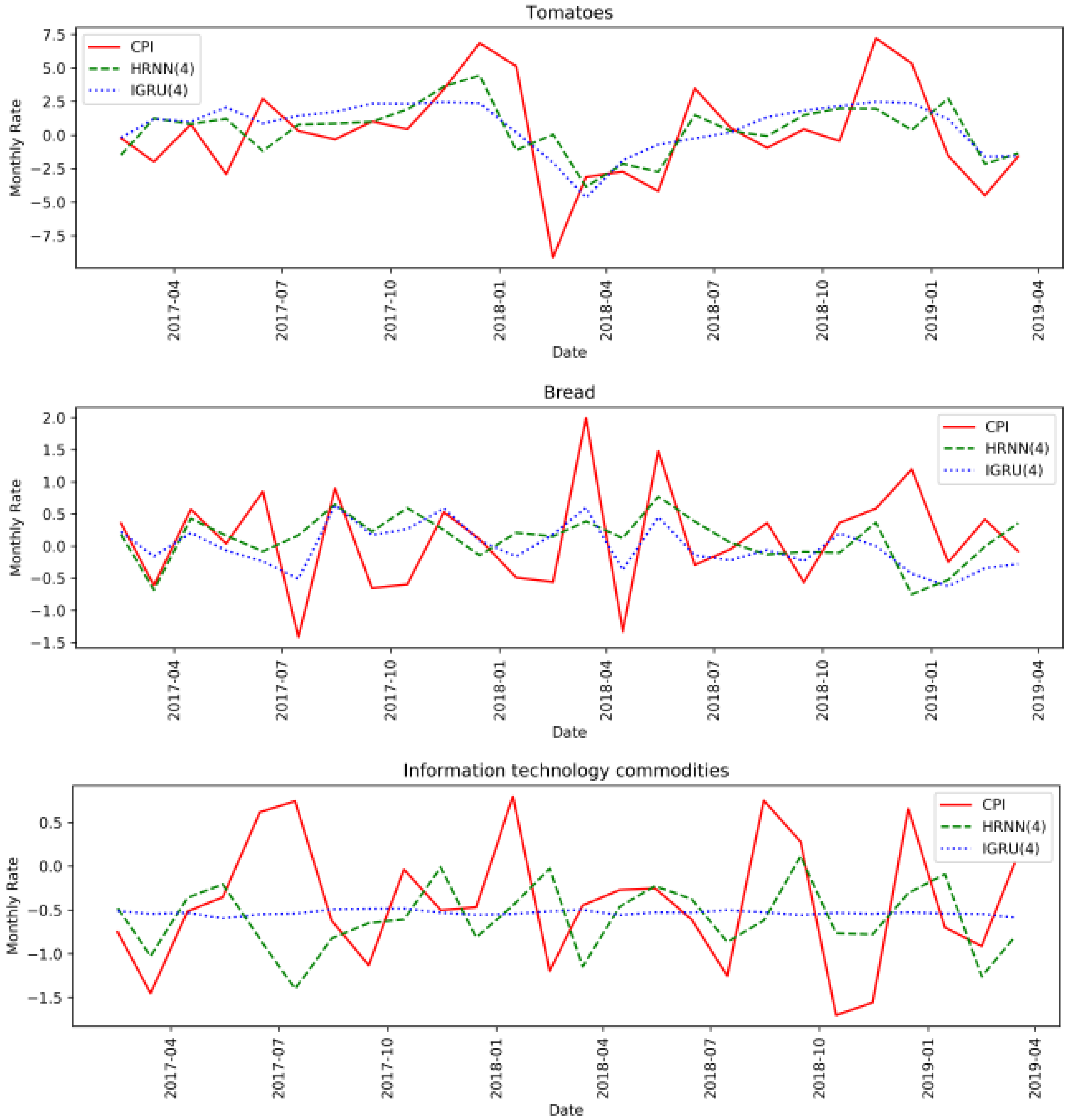
Table 5: HRNN(4) vs. I-GRU(4) results for different CPI sectors with respect to AR(1)

Industry Sector	HRNN(4)						I-GRU(4)					
	RMSE per horizon AR(1)=1.00				Correlation (at horizon=0)		RMSE per horizon AR(1)=1.00				Correlation (at horizon=0)	
	0	2	4	8	Pearson	Distance	0	2	4	8	Pearson	Distance
Apparel	0.83	0.87	0.84	0.88	0.04	0.19	0.88	0.88	0.85	0.92	0.05	0.23
Energy	0.94	0.96	0.99	0.98	0.34	0.32	0.94	0.98	1.02	0.99	0.18	0.28
Food & beverages	0.72	0.73	0.75	0.76	0.22	0.13	0.80	0.80	0.81	0.82	0.18	0.22
Housing	0.79	0.80	0.82	0.82	0.17	0.24	0.77	0.79	0.82	0.82	0.18	0.27
Medical care	0.79	0.82	0.81	0.82	0.03	0.17	0.79	0.83	0.83	0.84	0.08	0.15
Recreation	0.99	0.99	1.00	1.00	0.05	0.17	1.00	0.99	1.00	1.00	-0.07	0.17
Services	0.90	0.92	0.95	0.94	0.04	0.15	0.89	0.94	0.95	0.96	0.02	0.21
Transportation	0.83	0.84	0.85	0.85	0.27	0.28	0.82	0.85	0.86	0.88	0.26	0.36

Notes: The RMSE results are relative to the $AR(1)$ model and normalized according to its results, i.e., $\frac{RMSE_{Model}}{RMSE_{AR(1)}}$.

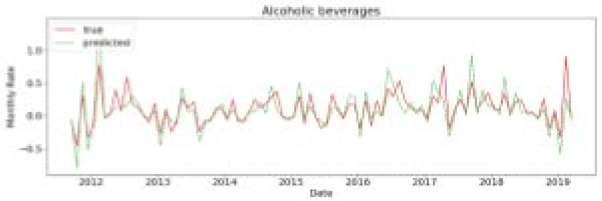
HRNN shows best performance in Food and Beverages sector which contains the most low-level CPI components.

Figure 7. *Examples of HRNN(4) predictions for disaggregated indexes.*

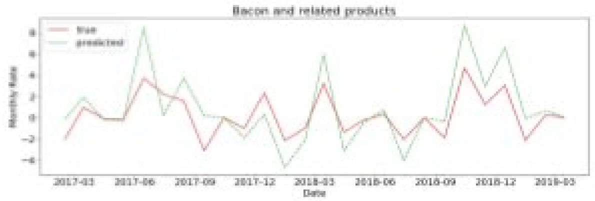


(a) *Admission to movies, theaters, and concerts*

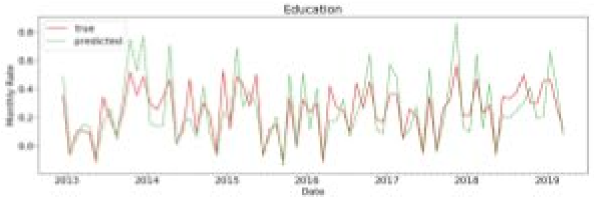
(b) *Alcoholic beverages*



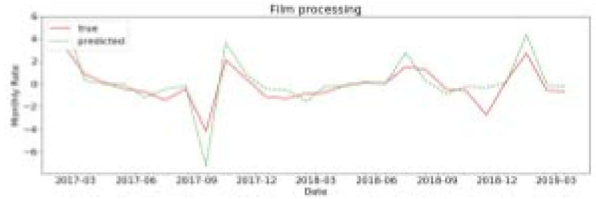
(c) *Bacon and related products*



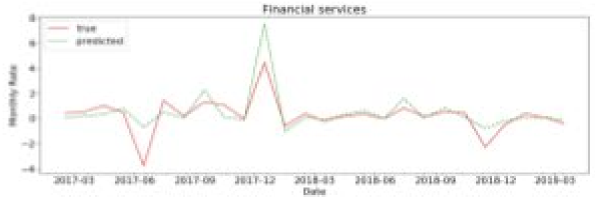
(d) *Education*



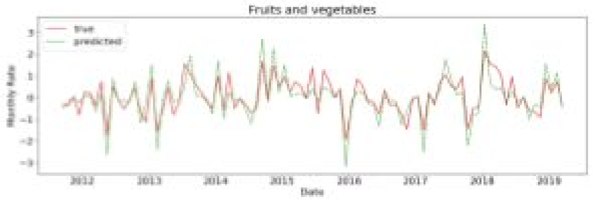
(e) *Film processing*



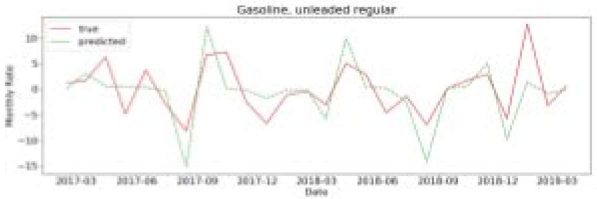
(f) *Financial services*



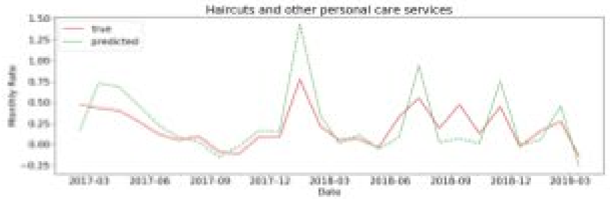
(g) *Fruits and vegetables*



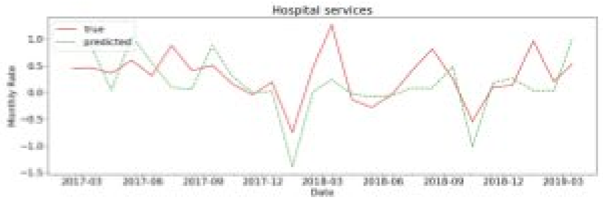
(h) *Gasoline, unleaded regular*



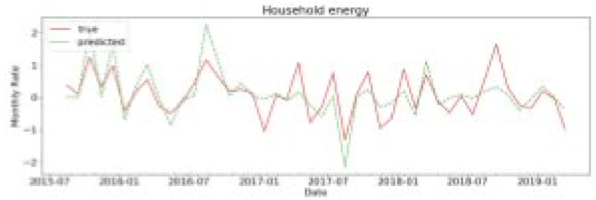
(i) *Haircuts and other personal care services*



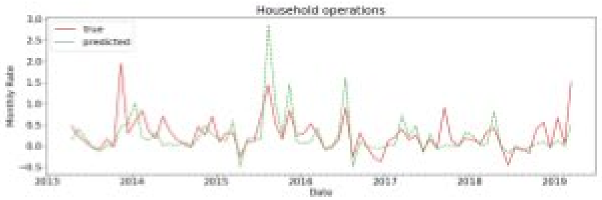
(j) *Hospital services*



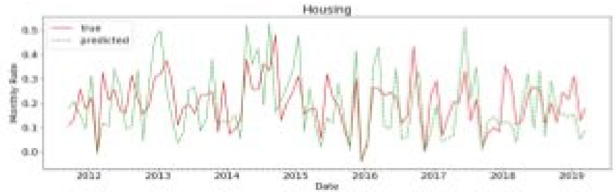
(k) *Household energy*



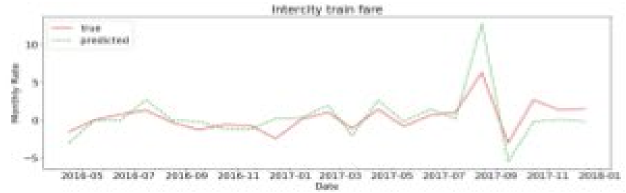
(l) *Household operations*



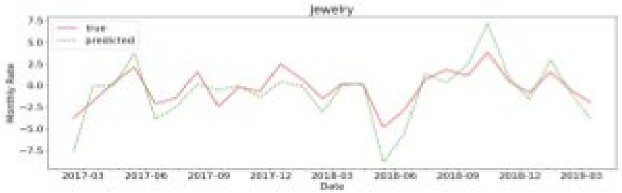
(a) *Housing*



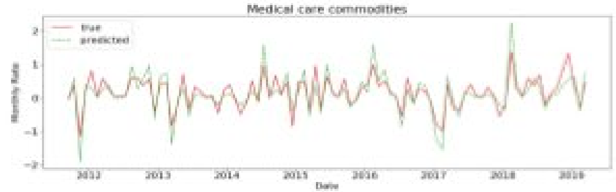
(b) *Intercity train fare*



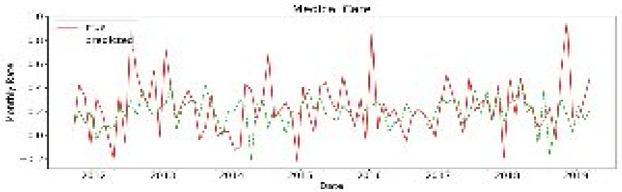
(c) *Jewelry*



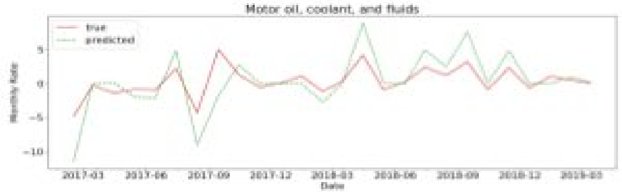
(d) *Medical care commodities*



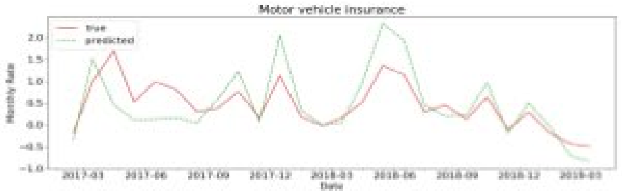
(e) *Medical care*



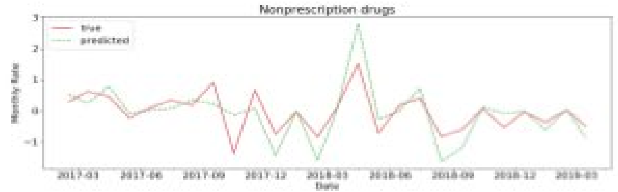
(f) *Motor oil, coolant, and fluids*



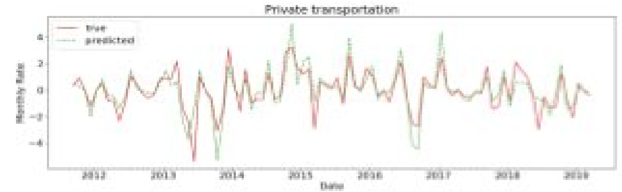
(g) *Motor vehicle insurance*



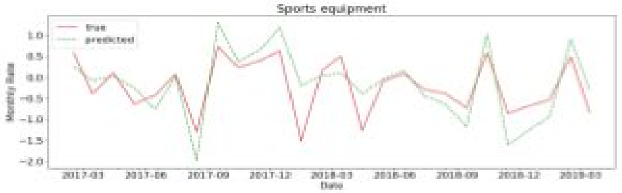
(h) *Nonprescription drugs*



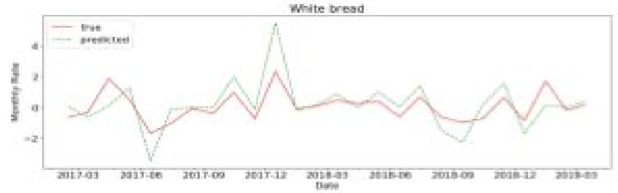
(i) *Private transportation*



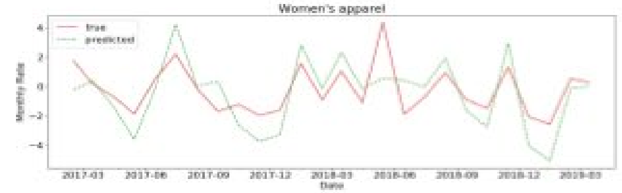
(j) *Sports equipment*



(k) *White bread*



(l) *Women's apparel*



Conclusion

Conclusion

- The hierarchical nature of the model enables information propagation from higher levels.

- HRNNs are superior at predicting low-level inflation components.

- Policy implications.

Thanks

Thank you for your attention.

Paper: Open Access @ **International Journal of Forecasting**

Replication files: **GitHub.com/AllonHammer/CPI_HRNN**

Comments: **jonathan@benchimol.name**

Other papers: **JonathanBenchimol.com/Research**