

Forecasting CPI Inflation Components with Hierarchical Recurrent Neural Networks

Oren Barkan^a, Jonathan Benchimol^b, Itamar Caspi^b,
Allon Hammer^{c,1}, Noam Koenigstein^{c,1}

^a*Ariel University*

^b*Bank of Israel*

^c*Tel-Aviv University*

Abstract

We present a hierarchical architecture based on Recurrent Neural Networks (RNNs) for predicting disaggregated inflation components of the Consumer Price Index (CPI). While the majority of existing research is focused on predicting headline inflation, many economic and financial institutions are interested in its partial disaggregated components. To this end, we developed the novel Hierarchical Recurrent Neural Network (HRNN) model, which utilizes information from higher levels in the CPI hierarchy to improve predictions at the more volatile lower levels. Based on a large dataset from the US CPI-U index, our evaluations indicate that the HRNN model significantly outperforms a vast array of well-known inflation prediction baselines. Our methodology and results provide additional forecasting measures and possibilities to policy and market makers on sectoral and component-specific price changes.

Keywords: Inflation forecasting, Disaggregated inflation, Consumer Price Index, Machine learning, Gated Recurrent Unit, Recurrent Neural Networks.

JEL Classification: C45, C53, E31, E37.

*Corresponding author. The views expressed in this paper are those of the authors and do not necessarily represent the views of the Bank of Israel.

Email addresses: orenba@ariel.ac.il (Oren Barkan), jonathan.benchimol@boi.org.il (Jonathan Benchimol), itamar.caspi@boi.org.il (Itamar Caspi), allonhammer@gmail.com (Allon Hammer), noamk@tauex.tau.ac.il (Noam Koenigstein)

1. Introduction

The Consumer Price Index (CPI) is a measure of the average change over time in the prices paid by a representative consumer for a common basket of goods and services. The CPI attempts to quantify and measure the average cost-of-living in a given country by estimating the purchasing power of a single unit of currency. Therefore, it is the key macroeconomic indicator for measuring inflation (or deflation). As such, the CPI is a major driving force in the economy influencing a plethora of market dynamics. In this work we present a novel model based on Recurrent Neural Networks (RNNs) for forecasting disaggregated CPI inflation components.

The ability to accurately estimate the upcoming disaggregated inflation rate is of high interest to policymakers and market players: Inflation forecasting is a critical tool in adjusting monetary policies around the world ([Friedman, 1961](#)). Central banks predict future inflation trends to justify interest rate decisions and to control and maintain inflation around its target. Better understanding of upcoming inflation dynamics at the component level can help inform and elucidate decision-makers for optimal monetary policy ([Ida, 2020](#)). Another example is investors in fixed-income markets who wish to estimate future sectorial inflation in order to foresee upcoming trends in discounted real returns. Some private firms need to predict specific inflation components in order to forecast price dynamics and mitigate risks accordingly. These are just a few examples that emphasize the importance of disaggregated inflation forecasting.

Fiscal authorities may also need to forecast sectoral inflation dynamics to adjust social security payments and assistance packages to specific industrial sectors. Government and private debt levels and interest payments heavily depend on the expected path of inflation. Our work suggests that hierarchical recurrent neural networks and machine learning methods deserve more attention from policy and market makers.

In the US, the Consumer Price Index (CPI) is calculated and reported by the Bureau of Labor Statistics (BLS). It represents the cost of a basket of goods and services across the country on a monthly basis. The CPI is a hierarchical composite index system that partitions all consumer goods and services into a hierarchy of increasingly detailed categories. In the US, the top CPI headline is composed of eight major sector indexes: (1) Housing, (2) Food and Beverages, (3) Medical Care, (4) Apparel, (5) Transportation, (6) Energy, (7) Recreation, and (8) Other goods and services. Each sector is composed of finer and finer sub-indexes until the entry levels or “leaves” are reached. These entry level indexes represent concrete measurable products or services whose price levels are being tracked. For 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*.

In the mid-1980s, many advanced economies began a major process of disinflation known as the “great moderation”. This period was characterized by steady low inflation and moderate yet steady economic growth ([Faust and Wright, 2013](#)). Later, the Global Financial Crisis (GFC) of 2008, and more recently the economic effects of the Covid-19 pandemic, were met with unprecedented monetary policies, potentially altering the underlying inflation dynamics worldwide ([Woodford, 2012](#); [Gilchrist et al., 2017](#); [Bernanke et al., 2018](#)). While economists still debate about the underlying forces that drive inflation, all agree on the importance and value of contemporary inflation research,

measurements and estimation. Moreover, the CPI is a composite index comprised of an elaborate hierarchy of sub-indexes each with its own dynamics and driving forces. Hence, in order to better understand inflation dynamics, it is useful to deconstruct the CPI index and look into the specific disaggregated components “underneath” the main headline.

Most existing inflation forecasting models attempt to predict the headline CPI while implicitly assuming the same approach can be effectively applied to its disaggregated components (Faust and Wright, 2013). However, as we show later, and in line with the literature, the disaggregated components are more volatile and harder to predict. Moreover, changes in the CPI components are more prevalent at the lower levels than up at the main categories. As a result, lower hierarchy levels often have less historical measurements for training modern machine learning algorithms.

In this work, we present the Hierarchical Recurrent Neural Network (HRNN) model—a novel model based on Recurrent Neural Networks (RNNs) that utilizes the CPI’s inherent hierarchy for improved predictions at its lower levels. HRNN is a hierarchical arrangement of RNNs analogous to the CPI’s hierarchy. This architecture allows for information to propagate from higher levels to lower levels in order to mitigate volatility and information sparsity that otherwise impedes advanced machine learning approaches. Hence, a key advantage of the HRNN model stems from its superiority at inflation predictions at lower levels of the CPI hierarchy. Our evaluations indicate that HRNN outperforms many existing baselines at inflation forecasting of different CPI components below the top headline and across different time horizons.

Finally, our data and code are publicly available on GitHub¹ to enable reproducibility and foster future evaluations of new methods. By doing so, we comply with the call to make data and algorithms more open and transparent to the community (Makridakis et al., 2020, 2018).

The remainder of the paper is organized as follows. Section 2 presents a literature review of baseline inflation forecasting models and machine learning models. Section 3 explains recurrent neural networks methodologies. Our novel Hierarchical Recurrent Neural Networks (HRNN) model is presented in Section 4. Section 5 describes the price data and data transformations. In Section 6, we present our results and compare them to alternative approaches. Section 7 discusses our work and its potential policy implications. Finally, we conclude in Section 8 and present a few additional tables and figures in the Appendix.

2. Related Work

While inflation forecasting is a challenging task of high importance, the literature indicates that significant improvement upon basic time-series models and heuristics is hard to achieve. Indeed, Atkeson et al. (2001) found that forecasts based on simple averages of past inflation were more accurate than all other alternatives, including the canonical Phillips curve and other forms of structural models. Similarly, Stock and Watson (2007, 2010) provide empirical evidence for the superiority of univariate models in forecasting inflation during the great moderation period (1985 to 2007) and during

¹The code and data are available at https://github.com/AllonHammer/CPI_HRNN.git

the recovery ensuing the GFC. More recently, [Faust and Wright \(2013\)](#) conducted an extensive survey of inflation forecasting methods and found that a simple “glide path” prediction from the current inflation rate performs as well as model-based forecasts for long-run inflation rates and often outperforms them.

Recently, an increasing amount of effort has been directed towards the application of machine learning models for inflation forecasting. For example, [Medeiros et al. \(2021\)](#) compared inflation forecasting with several machine learning models such as lasso regression, random forests, and deep neural networks. However, [Medeiros et al. \(2021\)](#) focused mostly on using exogenous features such as cash and credit availability, online prices, housing prices, consumer data, exchange rates, and interest rates. When exogenous features are considered, the emphasis shifts from learning the endogenous time series patterns to effectively extracting the predictive information from the exogenous features. In contrast to [Medeiros et al. \(2021\)](#), we preclude the use of any exogenous features and focus on harnessing the internal patterns of the CPI series. Moreover, unlike previous works that dealt with estimating the main headline, this work is focused on predicting the disaggregated indexes that comprise the CPI.

In general, machine learning methods flourish where data is found in abundance and many *training* examples are available. Unfortunately, this is not the case with CPI inflation data. While a large amount of relevant exogenous features exist, there are only twelve monthly readings annually. Hence, the amount of available training examples is limited. Furthermore, [Stock and Watson \(2007\)](#) show that statistics such as average inflation rate, conditional volatility, and persistency levels are shifting in time. Hence, inflation is a *non-stationary* process which further limits the amount of relevant historical data points.

[Coulombe et al. \(2020\)](#), [Mullainathan and Spiess \(2017\)](#), [Athey and Susan \(2018\)](#) and [Chakraborty and Joseph \(2017\)](#) present comprehensive surveys of general machine learning applications in economics. Here, we do not attempt to cover the plethora of research employing machine learning for economic forecasting. Instead, in the next section we focus on models that apply neural networks to CPI forecasting.

This research joins several studies that apply neural network methods to the specific task of inflation forecasting: [Nakamura \(2005\)](#) employed a simple feed-forward network to predict quarterly CPI headline values. A special emphasis is placed on *early stopping* methodologies in order to prevent over-fitting. Their evaluations are based on US CPI data during 1978-2003 and predictions are compared against several autoregressive (AR) baselines. Presented in Section 6, our evaluations confirm the findings of [Nakamura \(2005\)](#), that a fully connected network is indeed effective at predicting the headline CPI. However, when the CPI components are considered, we show that the model in this work demonstrates superior accuracy across all time horizons.

[Choudhary and Haider \(2012\)](#) used several neural networks to forecast monthly inflation rates in 28 countries in the Organisation for Economic Cooperation and Development (OECD). Their findings showed that on average, neural network models were superior in 45% of the countries while simple AR models of order one (AR1) performed better in 23% of the countries. They also proposed to arithmetically combine an ensemble of multiple networks for further accuracy.

[Chen et al. \(2001\)](#) explored semi-parametric nonlinear autoregressive models with exogenous variables (NLARX) based on neural networks. Their investigation covered a

comparison of different nonlinear activation functions such as the Sigmoid activation, radial basis activation, and Ridgelet activation.

[McAdam and McNelis \(2005\)](#) explored Thick Neural Network models that represent “trimmed mean” forecasts from several models. By combining the network with a linear Phillips Curve model, they predict the CPI for the US, Japan, and Europe at different levels.

In contrast to the aforementioned works, our model predicts monthly CPI values in *all* hierarchy levels. We utilize information patterns from higher levels of the CPI hierarchy in order to assist the predictions at lower levels. Such predictions are more challenging due to the inherent noise and information sparsity at the lower levels. Moreover, the HRNN model in this work is better equipped to harness sequential patterns in the data by employing *Recurrent Neural Networks*. Finally, we exclude the use of exogenous variables and rely solely on historical CPI data to focus on internal CPI patterns modeling.

[Zahara et al. \(2020\)](#) employed long-short term memory (LSTM) units for CPI predictions based on exogenous daily online prices of 34 food staples in Indonesia. They apply Recurrent Neural Networks (RNNs) to forecast the CPI but did not compare their approach to alternative baselines nor make their data and code available for further evaluation and reproducibility purposes.

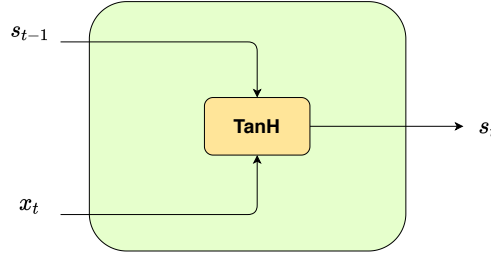
[Almosova and Andresen \(2019\)](#) employed LSTMs for inflation forecasting and compared them to multiple baselines such as autoregressive models, random walk models, seasonal autoregressive models, Markov switching models, and fully-connected neural networks. At all time horizons, the root mean squared forecast of their LSTM model was approximately one-third of the random walk model and significantly more accurate than the other baselines.

As we explain in Section 3.3, our model makes use of Gated Recurrent Networks (GRUs), which are similar to LSTMs. Unlike [Zahara et al. \(2020\)](#) and [Almosova and Andresen \(2019\)](#), a key contribution of our model stems from its ability to propagate useful information from higher levels in the hierarchy down to the nodes at lower levels. By ignoring the hierarchical relations between the different CPI components, our model is reduced to a set of simple unrelated GRUs. This set-up is similar to [Almosova and Andresen \(2019\)](#), as the difference between LSTMs and GRUs is negligible. In Section 6, we perform an ablation study in which HRNN ignores the hierarchical relations and is reduced to a collection of independent GRUs, similar to the model in [Almosova and Andresen \(2019\)](#). Our evaluations indicate that this approach is not optimal at any level of the CPI hierarchy.

3. Recurrent Neural Networks

Before describing our novel HRNN model, we discuss different Recurrent Neural Networks (RNNs) approaches. RNNs are neural networks that model sequences of data in which each value is assumed to be dependent on previous values. Specifically, RNNs are feed-forward networks augmented by implementing a feedback loop ([Mandic and Chambers, 2001](#)). As such, RNNs introduce a notion of time to the standard feed-forward neural networks and excel at modelling temporal dynamic behavior ([Chung et al., 2014](#)). Some RNN units retain an internal memory state from previous time

Figure 1. An illustration of a basic RNN unit.



Each line carries an entire vector, from the output of one node to the inputs of others. The yellow box is a learned neural network layer.

steps representing an arbitrarily long context window. Many RNN implementations were proposed and studied in the past, a comprehensive review and comparison of the different RNN architectures is available in [Lipton et al. \(2015\)](#); [Chung et al. \(2014\)](#). In this section, we will cover the three most popular units: Basic RNN, Long-Short Time Memory (LSTM), and Gated Recurrent Unit (GRU).

3.1. Basic Recurrent Neural Networks

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 results consisting of T samples of time series. Namely, the model's input at t is x_t , and its output (prediction) is s_t . The following set of equations defines a basic RNN unit:

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

where u , w and b are the model's parameters and $\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$ is the hyperbolic tangent function. Namely, the model's output from the previous period s_{t-1} is used as an additional input to the model at time t , along with the current input x_t . The linear combination $x_t u + s_{t-1} w + b$ is plug into a hyperbolic tangent *activation* function allowing the unit to model nonlinear relations between inputs and outputs. Different implementations may employ other activation functions, e.g., the Sigmoid function, some logistic functions, or a Rectified Linear Unit (ReLU) function ([Ramachandran et al., 2017](#)). Figure 1 depicts an illustration of a basic RNN unit.

3.2. Long Short Term Memory Networks

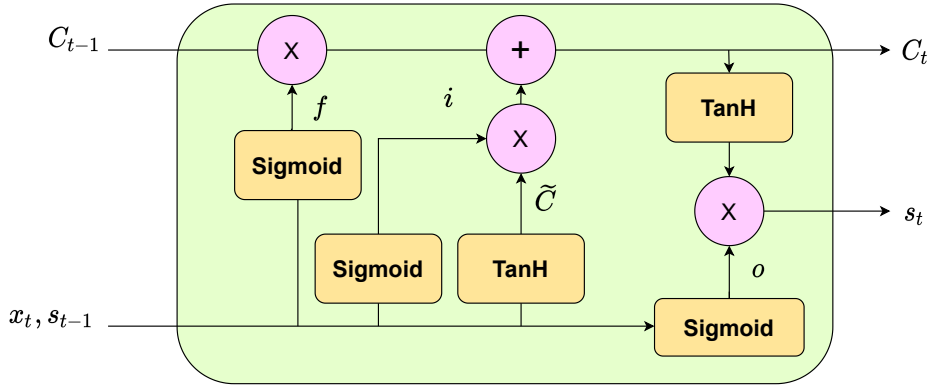
Basic RNNs suffer from the “short-term memory” problem: they utilize data from recent history to forecast, but if a sequence is long enough, it cannot carry relevant information from earlier periods to later ones, e.g., relevant patterns from the same month in previous years. Long Short Term Memory networks (LSTMs) deal with this “short-term memory” problem by introducing gates that enable the preservation of relevant “long-term memory” and combining it with the most recent data ([Hochreiter and Schmidhuber, 1997](#)). The introduction of LSTMs paved the way for significant strides forward in various fields such as natural language processing, speech recognition, robot control, and more ([Yu et al., 2019](#)).

An LSTM unit has the ability to “memorize” or “forget” information through the use of a special memory *cell state*, carefully regulated by three gates: an *input gate*, a *forget gate*, and an *output gate*. The gates regulate the flow of information into and out of the memory cell state. An LSTM unit is defined by the following set of equations:

$$\begin{aligned}
i &= \sigma(x_t u^i + s_{t-1} w^i + b^i), \\
f &= \sigma(x_t u^f + s_{t-1} w^f + b^f), \\
o &= \sigma(x_t u^o + s_{t-1} w^o + b^o), \\
\tilde{c} &= \tanh(x_t u^c + s_{t-1} w^c + b^c), \\
c_t &= f \times c_{t-1} + i \times \tilde{c}, \\
s_t &= o \times \tanh(c_t),
\end{aligned} \tag{2}$$

where $\sigma(x) = \frac{1}{1+e^{-x}}$ is the sigmoid or logistic activation function. u^i, w^i and b^i are the learned parameters that control the *input gate* i , u^f, w^f and b^f are the learned parameters that control the *forget gate* f , and u^o, w^o and b^o are the learned parameters that control the *output gate* o . \tilde{c} is the new candidate activation for the cell state determined by the parameters u^c, w^c and b^c . The *cell state* itself c_t is updated by the linear combination $c_t = f \times c_{t-1} + i \times \tilde{c}$, where c_{t-1} is its previous value of the cell state. The input gate i determines which parts of the candidate \tilde{c} should be used to modify the memory cell state, and the forget gate f determines which parts of the previous memory c_{t-1} should be discarded. Finally, the recently updated *cell state* c_t is “squashed” through a nonlinear hyperbolic tangent and the *output gate* o determines which parts of it should be presented in the output s_t . Figure 2 depicts an illustration of an LSTM unit.

Figure 2. An illustration of an LSTM Unit.



Each line carries an entire vector, from the output of one node to the inputs of others. The pink circles represent point-wise operations, while the yellow boxes are learned neural network layers. Lines merging denote concatenation, while a line forking denotes its content being copied and the copies going to different locations.

3.3. Gated Recurrent Unit

A Gated Recurrent Unit (GRU) improves the LSTM unit by dropping the *cell state* in favor of a more simplified unit that requires less learnable parameters (Dey and Salemt, 2017). GRU employs only two gates instead of three: an *update gate* and a *reset gate*. Using fewer parameters, GRUs are faster and more efficient, especially when training data is limited, such as in the case of inflation predictions and particularly disaggregated inflation components.

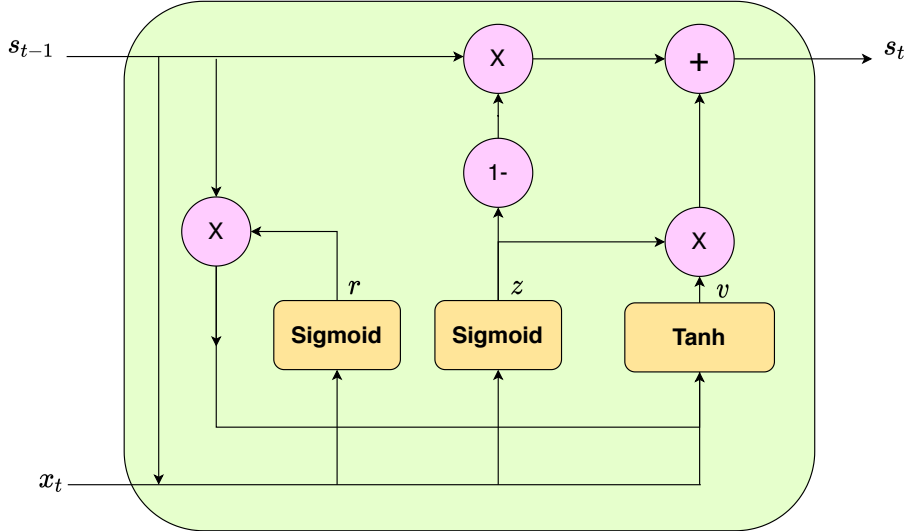
A GRU unit is defined by the following set of equations:

$$\begin{aligned} z &= \sigma(x_t u^z + s_{t-1} w^z + b^z), \\ r &= \sigma(x_t u^r + s_{t-1} w^r + b^r), \\ v &= \tanh(x_t u^v + (s_{t-1} \times r) w^v + b^v), \\ s_t &= z \times v + (1 - z) s_{t-1}, \end{aligned} \tag{3}$$

where u^z , w^z and b^z are the learned parameters that control the *update gate* z , and u^r , w^r and b^r are the learned parameters that control the *reset gate* r . The candidate activation v is a function of the input x_t and the previous output s_{t-1} , and is controlled by the learned parameters: u^v , w^v and b^v . Finally, the output s_t is a combination of the candidate activation v and the previous state s_{t-1} controlled by the *update gate* z . Figure 2 depicts an illustration of a GRU unit.

GRUs enable the “memorization” of relevant information patterns with significantly fewer parameters when compared to LSTMs. Hence, GRUs constitute the basic unit for our novel HRNN model described in Section 4.

Figure 3. An illustration of a GRU unit.



Each line carries an entire vector, from the output of one node to the inputs of others. The pink circles represent point-wise operations, while the yellow boxes are learned neural network layers. Lines merging denote concatenation, while a line forking denotes its content being copied and the copies going to different locations.

4. Hierarchical Recurrent Neural Networks

The disaggregated components at lower levels of the CPI hierarchy (e.g., newspapers, medical care products, etc.) suffer from missing data as well as higher volatility in change rates. HRNN exhibits a network graph in which each node is associated with a Recurrent Neural Network (RNN) unit that models the inflation rate of a specific (sub)-index (node) in the “full” CPI hierarchy. HRNN’s unique architecture allows it to propagate information from RNN nodes in higher levels to lower levels in the CPI hierarchy, coarse to fine-grained, via a chain of hierarchical informative priors over the RNNs’ parameters. As presented in Section 6, this unique property of HRNN is materialized in better predictions for nodes at lower levels of the hierarchy.

4.1. Model Formulation

Let $\mathcal{I} = \{n\}_{n=1}^N$ be an enumeration of the nodes in the CPI hierarchy graph. In addition, we define $\pi_n \in \mathcal{I}$ as the parent node of the node n . For example, if the nodes $n = 5$ and $n = 19$ represent the indexes of *tomatoes* and *vegetables* respectively, then $\pi_5 = 19$ i.e. the parent node of *tomatoes* is *vegetables*.

For each node $n \in \mathcal{I}$, we denote by $x_t^n \in \mathbb{R}$ the observed random variable that represents the CPI value of the node n at timestamp $t \in \mathbb{N}$. We further denote $X_t^n \triangleq (x_1^n, \dots, x_t^n)$, where $1 \leq t \leq T_n$ and T_n is the last timestamp for node n . Let $g : \mathbb{R}^m \times \Omega \rightarrow \mathbb{R}$ be a parametric function representing an RNN node in the hierarchy. Specifically, \mathbb{R}^m is the space of parameters that control the RNN unit, Ω is the input time series space, and the function g predicts a scalar value for the next value of the input series. Hence, our goal is to learn the parameters $\theta_n \in \mathbb{R}^m$ s.t. for $X_t^n \in \Omega$, $g(\theta_n, X_t^n) = x_{t+1}^n, \forall n \in \mathcal{I}$, and $1 \leq t < T_n$.

We proceed by assuming a Gaussian error on g ’s predictions and receive the following expression for the likelihood of the observed time series:

$$p(X_{T_n}^n | \theta_n, \tau_n) = \prod_{t=1}^{T_n} p(x_t^n | X_{t-1}^n, \theta_n, \tau_n) = \prod_{t=1}^{T_n} \mathcal{N}(x_t^n; g(\theta_n, X_{t-1}^n), \tau_n^{-1}), \quad (4)$$

where $\tau_n^{-1} \in \mathbb{R}$ is the variance of g ’s errors.

Next, we define a hierarchical network of normal priors over the nodes’ parameters that attach each node’s parameters with those of its parent node. The hierarchical priors follow:

$$p(\theta_n | \theta_{\pi_n}, \tau_{\theta_n}) = \mathcal{N}(\theta_n; \theta_{\pi_n}, \tau_{\theta_n}^{-1} \mathbf{I}), \quad (5)$$

where τ_{θ_n} is a configurable precision parameter that determines the “strength” of the relation between node n ’s parameters and the parameters of its parent π_n . Higher values of τ_{θ_n} strengthen the attachment between θ_n and its prior θ_{π_n} .

The precision parameter τ_{θ_n} can be seen as a global hyper-parameter of the model to be optimized via cross-validation. However, different nodes in CPI the hierarchy have varying degrees of correlation with their parent nodes. Hence, the value of τ_{θ_n} in HRNN is given by:

$$\tau_{\theta_n} = e^{\alpha + C_n}, \quad (6)$$

where α is a hyper-parameter and $C_n = \rho(X_{T_n}^n, X_{T_{\pi_n}}^{\pi_n})$ is the Pearson correlation coefficient between the time series of n and its parent π_n .

Importantly, Equation (5) describes a novel prior relationship between the parameters of a node and its parent in the hierarchy that “grows” increasingly stronger according to the historical correlation between the two series. This ensures that a child node n is kept close to its parent node π_n in terms of squared Euclidean distance in the parameters space, especially if they are highly correlated. Note that in the case of the root node (the headline CPI), π_n does not exist and hence we set a normal non-informative regularization prior with zero mean and unit variance.

Let us now denote the aggregation of all series from all levels by $X = \{X_{T_n}^n\}_{n \in \mathcal{I}}$. Similarly, we denote by $\theta = \{\theta_n\}_{n \in \mathcal{I}}$ and $T = \{T_n\}_{n \in \mathcal{I}}$ the aggregation of all the RNN parameters and precision parameters from all levels, respectively. Note that X (the data) is observed, θ are unobserved *learned* variables, and T are determined by Equation 6. The hyper-parameter α from Equation 6 is set by a cross-validation procedure.

With these definitions at hand, we now proceed with the Bayes rule. From Equation 4 and Equation 5, we extract the posterior probability:

$$p(\theta|X, T) = \frac{p(X|\theta, T)p(\theta)}{P(X)} \propto \prod_{n \in \mathcal{I}} \prod_{t=1}^{T_n} \mathcal{N}(x_t^n; g(\theta_n, X_{t-1}^n), \tau_n^{-1}) \prod_{n \in \mathcal{I}} \mathcal{N}(\theta_n; \theta_{\pi_n}, \tau_{\theta_n}^{-1} \mathbf{I}). \quad (7)$$

HRNN optimization follows a *Maximum A-Posteriori* (MAP) approach. Namely, we wish to find optimal parameter values θ^* such that:

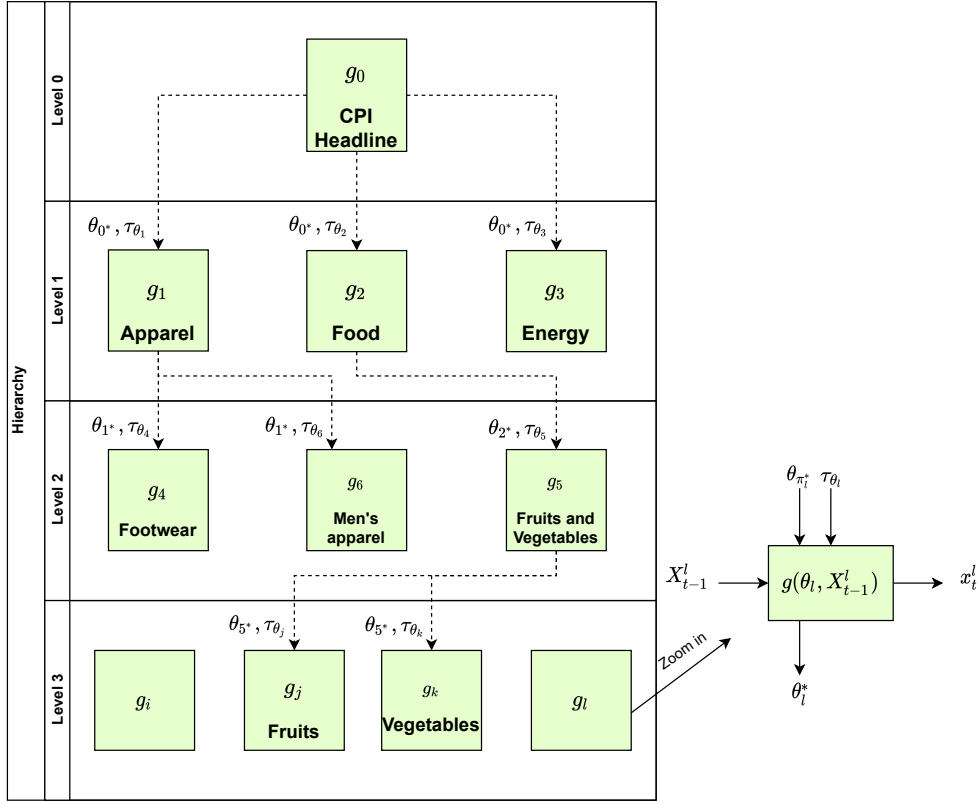
$$\theta^* = \underset{\theta}{\operatorname{argmax}} \log p(\theta|X, T). \quad (8)$$

Note that the objective in Equation (8) depends on the parametric function g . HRNN is a general framework that can use any RNN, e.g., Simple RNN, LSTM, GRU, etc. In this work, we chose g to be a scalar GRU because GRUs are capable of long-term memory but with fewer parameters than LSTMs. Hence, each node n is associated with a GRU with its own parameters: $\theta_n = [u_n^z, u_n^r, u_n^v, w_n^z, w_n^r, w_n^v, b_n^z, b_n^r, b_n^v]$. Then, $g(\theta_n, X_t^n)$ is computed by t successive applications of the GRU to x_i^n with $1 \leq i \leq t$ according to Equation (3). Finally, the HRNN optimization proceeds with stochastic gradient ascent over the objective in Equation (8). Figure 4 depicts an illustration of the entire HRNN architecture.

4.2. HRNN Inference

In machine learning, after the model’s parameters have been estimated in the *training* process, it can be applied to make predictions in a process known as *inference*. In our case, equipped with the MAP estimate θ^* , inference with the HRNN model is achieved as follows: Given a sequence of historical CPI values X_t^n for node n , we predict the next CPI value $y_{t+1}^n = g(\theta_n, X_t^n)$, as explained in Section 4.1. This type of prediction is for next month’s CPI, namely, horizon $h = 0$. In this work, we also test the ability of the model to perform predictions for further horizons $h \in \{0, \dots, 8\}$. The h -horizon

Figure 4. An illustration of the full HRNN model.



predictions are obtained in a recursive manner, whereby each predicted value y_t^n is fed back as an input for the prediction of y_{t+1}^n . As expected, Section 6 shows that forecasting accuracy gradually degrades as horizon h increases.

5. Dataset

This work is based on monthly CPI data released by the US Bureau of Labor and Statistics (BLS). In what follows, we discuss the dataset's characteristics and our pre-processing procedures. For the sake of reproducibility, the final version of the processed data is available in our HRNN code.

5.1. The US Consumer Price Index

The official CPI of each month is released by the BLS several days into the following month. The price tags are collected in 75 urban areas throughout the US from about 24,000 retail and service establishments. The housing and rent rates are collected from about 50,000 landlords and tenants across the country. The BLS releases two different measurements according to urban demographics:

1. The **CPI-U** represents the CPI for urban consumers and covers approximately 93% of the total population. According to the Consumer Expenditure Survey, the CPI items and their relative weights are derived from their estimated expenditure. These items and their weights are updated each year in January.
2. The **CPI-W** represents the CPI for urban wage earners and clerical workers and covers about 29% of the population. This index is focused on households with

at least 50 percent of income coming from clerical or wage-paying jobs, and at least one of the household’s earners must have been employed for at least 70% of the year. CPI-W reflects changes in the cost of benefits, as well as future contract obligations.

In this work, we focus on CPI-U, as it is generally considered the best measure for the average cost of living in the US. Monthly CPI-U data per product is generally available from January 1994. Our samples thus span from January 1994 to March 2019. Note that throughout the years, new indexes were added, and some indexes have been omitted. Consequently, hierarchies can change, which contributes to the challenge of our exercise.

5.2. The CPI Hierarchy

The CPI-U is an eight-level deep hierarchy comprising 424 different nodes (indexes). Level 0 represents the headline CPI, or the aggregated index of all components. An index at any level is associated with a weight between 0-100, which represents its contribution to the headline CPI at level 0. Level 1 consists of the 8 main aggregated categories or sectors: (1) “Food and Beverages”, (2) “Housing”, (3) “Apparel”, (4) “Transportation”, (5) “Medical Care”, (6) “Recreation”, (7) “Education and Communication”, and (8) “Other Goods and Services”. Mid-levels (2-5) consist of more specific aggregations e.g., “Energy Commodities”, “Household Insurance”, etc. The lower levels (6-8) consists of fine-grained indexes, e.g., “Apples”, “Bacon and Related Products”, “Eyeglasses and Eye Care”, “Tires”, “Airline fares”, etc. Tables 7 and 8 (in Appendix A) depict the first three hierarchies of the CPI (levels 0-2).

5.3. Data Preparation

We used publicly available data from the BLS website². However, the BLS releases hierarchical data on a monthly basis in separate files. Hence, separate monthly files from January 1994 until March 2019 were processed and aggregated to create a single repository. Moreover, the format of these files has changed over the years (e.g., txt, pdf, and csv formats were all in use) and a significant effort was made in order to parse the changing formats from different time periods.

The hierarchical CPI data is released in terms of monthly index values. We transformed the CPI values to monthly logarithmic change rates as follows: We denote by x_t the CPI value (of any node) at month t . The logarithmic change rate at month t is denoted by $rate(t)$ and given by:

$$rate(t) = 100 \times \log \left(\frac{x_t}{x_{t-1}} \right). \quad (9)$$

Unless otherwise mentioned, the remainder of the paper relates to monthly logarithmic change rates as in Equation (9).

We split the data into a *training* dataset and a *test* dataset as follows: For each time series, we kept the first (early in time) 70% of the measurements for the *training* dataset.

²Taken from: www.bls.gov/cpi

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.

The remaining 30% of the measurements were removed from the *training* dataset and used to form the *test* dataset. The *training* dataset was used to train the HRNN model as well as the other baselines. The *test* dataset was used for evaluations. The results in Section 6 are based on this split.

Table 1 summarizes the number of data points and general statistics of the CPI time series after applying Equation (9). When comparing the headline CPI with the full hierarchy, we see that at lower levels the standard deviation (STD) is significantly higher and the dynamic range is larger, implying much more volatility. The average number of measurements per index decreases at the lower levels of the hierarchy as not all indexes are available for the entire period.

Figure 5 depicts box plots of the CPI change rate distributions at different levels. The boxes depict the median value and the upper 75th and lower 25th percentiles. The whiskers indicate the overall minimum and maximum rates. Figure 5 further emphasizes that as we go down the CPI hierarchy, the change rates are more volatile.

High dynamic range, high standard deviation, and less training data are all indicators of the difficulty of making predictions inside the hierarchy. Based on this information, we can expect that the disaggregated component predictions inside the hierarchy will be more difficult than the headline.

Finally, Figure 6 depicts a box plot of the CPI change rate distribution for different sectors. We notice that some sectors (e.g., apparel and energy) suffer from higher volatility than others. As expected, predictions for these sectors will be more difficult.

6. Evaluation and Results

We evaluate HRNN and compare it with well-known baselines for inflation prediction as well as some alternative machine learning approaches. In what follows, we use the following notation: Let x_t be the CPI log-change rate at month t . We consider models for \hat{x}_t - an estimate for x_t based on historical values. Additionally, we denote by ε_t the random estimation error at time t . In all cases, the h -horizon forecasts were generated by recursively iterating the one-step forecasts forward. Hyper-parameters were set through a 10-fold cross-validation procedure.

Figure 5. Box plots of monthly inflation rate per hierarchy level.

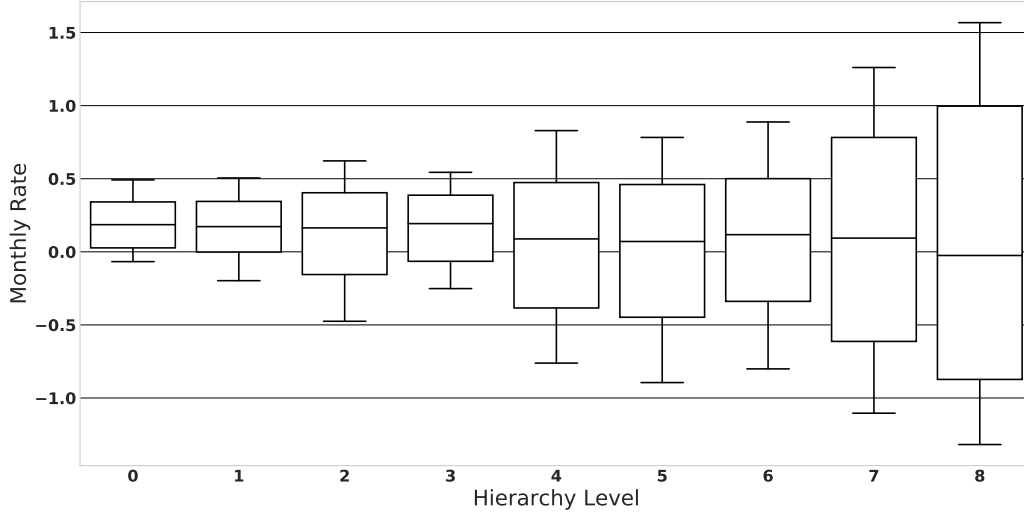
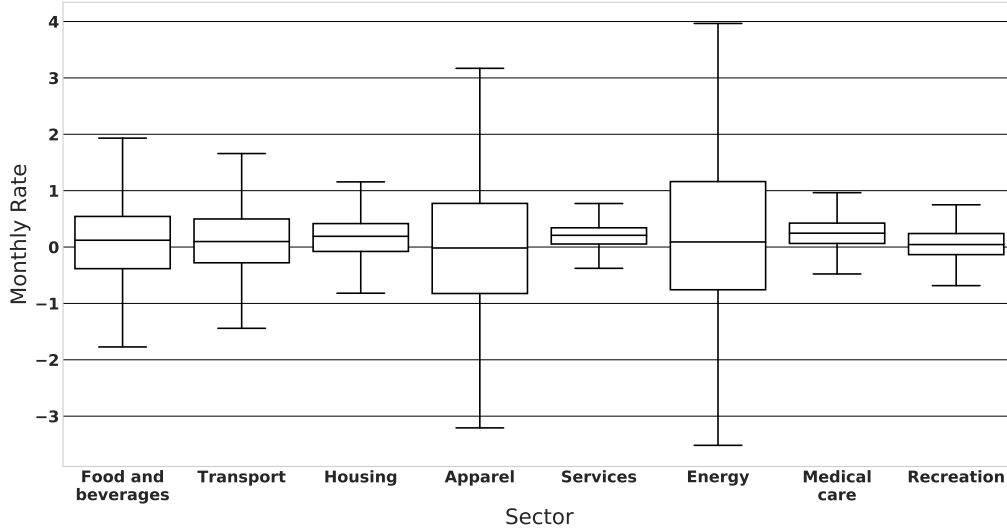


Figure 6. Box plots of monthly inflation rate per sector.



6.1. Baseline Models

We compare HRNN with the following CPI prediction baselines:

1. **Autoregression (AR)**- The $AR(\rho)$ estimates \hat{x}_t based on the previous ρ months as follows: $\hat{x}_t = \alpha_0 + (\sum_{i=1}^{\rho} \alpha_i x_{t-i}) + \varepsilon_t$, where $\{\alpha_i\}_{i=0}^{\rho}$ are the model's parameters.
2. **Phillips Curve (PC)** - A $PC(\rho)$ is an extension of $AR(\rho)$ that considers the unemployment rate u_t at month t in CPI forecasting model such as: $\hat{x}_t = \alpha_0 + (\sum_{i=1}^{\rho} \alpha_i x_{t-i}) + \beta u_{t-1} + \varepsilon_t$, where $\{\alpha_i\}_{i=0}^{\rho}$ and β are the model's parameters.

3. **Vector Autoregression (VAR)** - The VAR(ρ) model is a multivariate generalization of AR(ρ). It is frequently used when two or more time series may comove. VAR(ρ) estimates next month's values of k time series based on their historical values from the previous ρ months as follows: $\hat{X}_t = A_0 + (\sum_{i=1}^{\rho} A_i X_{t-i}) + \epsilon_t$, where X_t are the last ρ values from k different time series at month t , and \hat{X}_t are the model's estimates of these values, $\{A_i\}_{i=0}^{\rho}$ are a $(k \times k)$ matrices of parameters, and ϵ_t is a vector of error terms.
4. **Random Walk (RW)** - We consider the RW(ρ) model of [Atkeson et al. \(2001\)](#). RW(ρ) is a simple, yet effective, model that predicts next month's CPI as an average of the last ρ months by: $\hat{y}_t = \frac{1}{\rho} \sum_{i=1}^{\rho} x_{t-i} + \epsilon_t$.
5. **Auto Regression in Gap (AR-GAP)** - The AR-GAP model subtracts a fixed inflation trend before predicting the inflation in gap ([Faust and Wright, 2013](#)). Inflation gap is defined as $g_t = x_t - \tau_t$, where τ_t is the inflation trend at time t which represents a slowly-varying local mean. This trend value is estimated using RW(ρ) as follows: $\tau_t = \frac{1}{\rho} \sum_{i=1}^{\rho} x_{t-i}$. By accounting for the local inflation trend τ_t , the model attempts to increase stationarity in g_t and estimate it by $\hat{g}_t = \alpha_0 + \sum_{i=1}^{\rho} \alpha_i g_{t-i} + \epsilon_t$, where $\{\alpha_i\}_{i=0}^{\rho}$ are the model's parameters. Finally, τ_t is added back to \hat{g}_t to achieve the forecast for the final inflation prediction: $\hat{x}_t = \hat{g}_t + \tau_t$.
6. **Logistic Smooth Transition Auto Regressive Model (LSTAR)**- The LSTAR is an extension of AR that allows for changes in the model parameters according to a transition variable $F(t; c, \gamma)$. LSTAR(ρ, c, γ) consists of two AR(ρ) components that describe two trends in the data (high and low), and a nonlinear transition function that links them as follows:

$$\hat{x}_t = \left(\alpha_0 + \sum_{i=1}^{\rho} \alpha_i x_{t-i} \right) (1 - F(t; \gamma, c)) + \left(\beta_0 + \sum_{i=1}^{\rho} \beta_i x_{t-i} \right) F(t; \gamma, c) + \epsilon_t, \quad (10)$$

where $F(t; \gamma, c) = \frac{1}{1 + e^{-\gamma(t-c)}}$ is a first-order logistic transition function that depends on the location parameter c , and a smoothing parameter γ . The location parameter c can be interpreted as the threshold between the two AR(ρ) regimes, in the sense that the logistic function changes monotonically from 0 to 1 as t increases and balances symmetrically at $t = c$ ([van Dijk et al., 2002](#)). The model's parameters are $\{\alpha_i\}_{i=0}^{\rho}$ and $\{\beta_i\}_{i=0}^{\rho}$, while γ and c are hyper-parameters.

6.2. Ablation Models

In order to demonstrate the specific contributions of each component of the HRNN model, we conducted an ablation study that considered "simpler" machine learning alternatives to HRNN as follows:

1. **Single (S-GRU)**- The S-GRU(ρ) is a single GRU unit that receives the last ρ values as inputs in order to predict the next value. In GRU(ρ), a single GRU is used for all the time series that comprise the CPI hierarchy. This baseline utilizes all the benefits of a GRU but assumes that the different components of the CPI behave similarly and a single unit is sufficient to model all the nodes.

2. **Independent GRUs (I-GRUs)**- In I-GRUs(ρ), we trained a different GRU(ρ) unit for each CPI node. The S-GRU and I-GRU approaches represent two extremes: The first attempts to model all the CPI nodes with a single model, while the second treats each node separately. I-GRUs(ρ) is equivalent to a variant of HRNN that ignores the hierarchy by setting the precision parameter $\tau_{\theta_n} = 0; \forall n \in \mathcal{I}$. Namely, this is a simple variant of HRNN that trains independent GRUs, one for each index in the hierarchy.
3. **K-Nearest Neighbors GRU (KNN-GRU)**- In order to demonstrate the contribution of the hierarchical structure of HRNN, we devised the KNN-GRU(ρ) baseline. KNN-GRU attempts to utilize information from multiple Pearson-correlated CPI nodes without employing the hierarchical informative priors. Hence, KNN-GRU presents a “simpler” alternative to HRNN that replaces the hierarchical structure with elementary vector GRUs as follows: First, the k nearest neighbors of each CPI node were found using the Pearson correlation measure. Then, separate vector GRU(ρ) units were trained for each CPI aggregate along its k most similar nodes using the last ρ values of node n and its k -nearest nodes. By doing so, the KNN-GRU(ρ) baseline was able to utilize both the benefits of GRU units together with relevant information that comes from correlated nodes.
4. **Fully Connected Neural Network (FC)**- The FC(ρ) is based on a fully connected neural network with one hidden layer and a ReLU activation. The output layer employs no activation in order to formulate as a regression problem with a squared loss optimization. The inputs to the FC(ρ) model are the last ρ samples and the output is the predicted value for the next month.

Note that we have also experimented with “deeper” networks of more than one layer, but those did not yield a noticeable improvement. We attribute this to the relatively low number of training examples and features inherent to the CPI prediction problem.

6.3. Evaluation Metrics

We report results in terms of three evaluation metrics:

1. **Root Mean Squared Error (RMSE)**- The RMSE is given by:

$$RMSE = \sqrt{\frac{1}{T} \sum_{t=1}^T (x_t - \hat{x}_t)^2}, \quad (11)$$

where x_t are the monthly change rate for month t , and \hat{x}_t are the corresponding predictions.

2. **Pearson Correlation Coefficient**- The Pearson correlation coefficient ϕ is given by:

$$\phi = \frac{COV(X_T, \hat{X}_T)}{\sigma_X, \sigma_{\hat{X}}}, \quad (12)$$

where $COV(X_T, \hat{X}_T)$ is the covariance between the series of actual values and the predictions, and $\sigma_{X_T}, \sigma_{\hat{X}_T}$ are the standard deviations of the actual values and the predictions, respectively.

3. **Distance Correlation Coefficient-** In contrast to Pearson correlation, which detects linear association between two random variables, the distance correlation measure can also detect nonlinear correlations. The distance correlation is derived from the distance covariance and distance variance measures. Given X_T and \hat{X}_T , an actual and predicted CPI values across T months respectively, we define two corresponding distance matrices D^X and $D^{\hat{X}}$ as follows:

$$\begin{aligned} D_{j,k}^X &= \|x_j - x_k\|_2, \quad j, k = 1, 2, \dots, T, \\ D_{j,k}^{\hat{X}} &= \|\hat{x}_j - \hat{x}_k\|_2, \quad j, k = 1, 2, \dots, T, \end{aligned} \quad (13)$$

where $j, k = 1, 2, \dots, T$ represent time steps, and $\|\cdot\|_2$ is the Euclidean norm. The matrices D^X and $D^{\hat{X}}$ are centralized into \bar{D}^X and $\bar{D}^{\hat{X}}$ according to:

$$\begin{aligned} \bar{D}_{j,k}^X &= D_{j,k}^X - \bar{D}_{j\cdot}^X - \bar{D}_{\cdot k}^X + \bar{D}^X_{\cdot\cdot}, \\ \bar{D}_{j,k}^{\hat{X}} &= D_{j,k}^{\hat{X}} - \bar{D}_{j\cdot}^{\hat{X}} - \bar{D}_{\cdot k}^{\hat{X}} + \bar{D}^{\hat{X}}_{\cdot\cdot}, \end{aligned} \quad (14)$$

where $\bar{D}_{j\cdot}^X$ and $\bar{D}_{j\cdot}^{\hat{X}}$ are the mean value of the the j 'th row of D^X and $D^{\hat{X}}$ respectively. $\bar{D}_{\cdot k}^X$ and $\bar{D}_{\cdot k}^{\hat{X}}$ are the mean value of the the k 'th column of D^X and $D^{\hat{X}}$ respectively, and $\bar{D}^X_{\cdot\cdot}$ and $\bar{D}^{\hat{X}}_{\cdot\cdot}$ are the means of all cells in D^X and $D^{\hat{X}}$ respectively.

The distance co-variance is the arithmetic mean of the element-wise product of \bar{D}^X and $\bar{D}^{\hat{X}}$:

$$dCov(\bar{D}^X, \bar{D}^{\hat{X}}) = \frac{1}{T^2} \sum_{j=1}^T \sum_{k=1}^T \bar{D}_{j,k}^X \bar{D}_{j,k}^{\hat{X}}. \quad (15)$$

Similarly, the distance variance is defined by:

$$\begin{aligned} dVar(\bar{D}^X) &= dCov(\bar{D}^X, \bar{D}^X) = \frac{1}{T^2} \sum_{j,k} \left(\bar{D}_{j,k}^X\right)^2, \\ dVar(\bar{D}^{\hat{X}}) &= dCov(\bar{D}^{\hat{X}}, \bar{D}^{\hat{X}}) = \frac{1}{T^2} \sum_{j,k} \left(\bar{D}_{j,k}^{\hat{X}}\right)^2. \end{aligned} \quad (16)$$

Finally, the distance correlation coefficient r_d is given by:

$$r_d = \frac{dCov(X_T, \hat{X}_T)}{\sqrt{dVar(X_T) \times dVar(\hat{X}_T)}} \quad (17)$$

6.4. Results

The HRNN model is unique in its ability to utilize information from higher levels in the CPI hierarchy in order to make predictions at lower levels. Therefore, we provide

results for each level of the CPI hierarchy - overall 424 disaggregated indexes belonging to 8 different hierarchies. For the sake of completion, we also provide results for the headline CPI index by itself. It is important to note that in this case, the HRNN model cannot utilize its hierarchical mechanism and has no advantage over the alternatives.

The results are reported for horizons 0,1,2,3,4 and 8 months in Table 6. Following Faust and Wright (2013) and Aparicio and Bertolotto (2020), our results present the Diebold and Mariano (1995) tests, RMSEs, and Pearson correlation measures to assess one model's forecasting ability and quality over another. We also present in our results the Distance correlation to assess nonlinear comovements. The HRNN(4) model is significantly better than the other models according to Diebold and Mariano (1995) statistical significance tests, which all yielded to p-value<0.05 for all results.³

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	-0.05	-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
FC(4)	1.03	1.03	1.04	1.04	1.04	1.05	0.12	0.09
HRNN(1)	0.81	0.82	0.85	0.85	0.84	0.86	0.18	0.14
HRNN(2)	0.81	0.82	0.84	0.84	0.84	0.86	0.18	0.14
HRNN(3)	0.81	0.82	0.84	0.84	0.84	0.86	0.19	0.15
HRNN(4)	0.80	0.81	0.83	0.83	0.83	0.85	0.20	0.16
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

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., $\frac{RMSE_{Model}}{RMSE_{AR(1)}}$.

Table 6 depicts the average results from all the disaggregated indexes in the CPI hierarchy. The results are relative to the AR(1) models and normalized according to: $\frac{RMSE_{Model}}{RMSE_{AR(1)}}$. In HRNN we set $\alpha = 1.5$, and the V-GRU(ρ) models were based on $k = 5$ nearest neighbors. Table 6 shows that different versions of the HRNN model repeatedly

³For clarity purposes, we did not report the Diebold and Mariano (1995) results, which are available upon request.

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	0.92	0.96	0.33	0.25
RW(4)	1.05	0.98	0.99	1.01	0.97	0.96	-0.08	-0.06
Phillips(4)	0.93	0.94	0.95	0.95	0.93	0.95	0.33	0.25
FC(4)	0.92	0.94	0.94	0.96	0.93	0.94	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
HRNN(4) / GRU(4)	1.02	1.01	1.00	1.03	1.00	1.00	0.16	0.12

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., $\frac{RMSE_{Model}}{RMSE_{AR(1)}}$.

outperform the alternatives at any horizon. Notably, HRNN is superior to I-GRU, which emphasizes the importance of using hierarchical information and the superiority of HRNN over regular GRUs. Additionally, the HRNN is also superior to the different KNN-GRU models, which emphasizes the specific way HRNN employs informative priors based on the CPI hierarchy.

For the sake of completion, we also provide results for predictions at the head of the CPI index. Table 3 summarizes these results. When considering only the headline, the hierarchical mechanism of HRNN is redundant and the model is similar to a single GRU(4) unit. It appears we do not observe any advantage for the use of GRUs in this case when compared to a simple AR model. However, we see an advantage of simple deep learning models such as fully connected networks (FC) that outperform the more “traditional” approaches including LSTAR.

Table 4: HRNN(4) results at different levels of the CPI hierarchy with respect to AR(1)

Hierarchy Level	RMSE per horizon AR(1)=1.00						Correlation (at horizon=0)	
	0	1	2	3	4	8	Pearson	Distance
Level 1	0.98	0.99	1.00	1.00	1.00	1.00	0.23	0.21
Level 2	0.92	0.92	0.94	0.94	0.94	0.96	0.22	0.20
Level 3	0.81	0.82	0.83	0.84	0.84	0.85	0.22	0.19
Level 4	0.8	0.81	0.82	0.82	0.82	0.83	0.11	0.09
Level 5	0.78	0.8	0.8	0.79	0.8	0.82	0.21	0.17
Level 6	0.79	0.8	0.79	0.81	0.83	0.83	0.17	0.13
Level 7	0.75	0.76	0.8	0.81	0.8	0.83	0.15	0.12
Level 8	0.73	0.74	0.79	0.79	0.79	0.81	0.08	0.07

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

Table 4 depicts the results of HRNN(4), the best model, across all hierarchies (1-8, excluding the headline). Results are averaged over all disaggregated components and

Table 5: HRNN(4) results for different CPI sectors with respect to AR(1)

Industry Sector	RMSE per horizon AR(1)=1.00						Correlation (at horizon=0)	
	0	1	2	3	4	8	Pearson	Distance
Apparel	0.86	0.87	0.87	0.88	0.87	0.9	0.17	0.12
Energy	0.97	0.98	0.98	0.98	0.98	0.98	0.11	0.08
Food and beverages	0.77	0.79	0.78	0.78	0.79	0.81	0.1	0.08
Housing	0.82	0.82	0.83	0.84	0.83	0.85	0.23	0.17
Medical care	0.80	0.83	0.82	0.82	0.82	0.84	0.22	0.17
Recreation	0.99	0.99	1.00	1.00	1.00	1.00	0.08	0.06
Services	0.92	0.92	0.93	0.93	0.93	0.94	0.21	0.16
Transportation	0.88	0.9	0.9	0.91	0.92	0.93	0.23	0.17

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

normalized by the AR(1) model RMSE as before. As evident from Table 4, the HRNN model shows the best relative performance at the lower levels of the hierarchy, where the CPI indexes are more volatile and the hierarchical priors are most effective.

Table 5 compares the results of HRNN(4) across different sectors. Again, the results are averaged over all disaggregated components and presented as normalized gains with respect to the AR(1) model as before. The best improvement of HRNN appears to be in the Food and Beverages group. This can be explained by the fact that the Food and Beverages sub-hierarchy is the deepest and most elaborate hierarchy in the CPI tree. When the hierarchy is deeper and more elaborate, HRNN advantages are emphasized.

Finally, Figure 7 depicts specific examples of three disaggregated indexes: Gasoline, Medical Care, and Newspapers. These indexes are located down at the bottom of the CPI hierarchy and suffer from relatively high volatility. Figure 7 presents next month predictions on the *test* period only (i.e., months that were not used to train the model). The HRNN(4) model seems to track and predict the trends of the real indexes accurately. Figure 8 and Figure 9 in Appendix A depict additional examples for a large variety of disaggregated CPI components.

6.5. In the Wake of the GFC

In order to study the GFC effect on HRNN's performance, we removed the data from 2008 onward and repeated the experiment of Table 6, using only the data from 1997 up to 2008. The results of this experiment are summarized in Table 6. In terms of RMSE, the gains of HRNN in Table 6 vary from 0.85 up to 0.8, in contrast to Table 6 where the gains vary from 0.89 to 0.95, revealing that during the turmoil of the GFC, when the demand for reliable and precise forecasting tools is enhanced, HRNN's forecasting abilities remain robust. In fact, its forecasting superiority was somewhat enhanced during the GFC when compared to the AR(1) baseline.

Table 6 also provides a glimpse into HRNN's nonlinearity at work: While the Distance correlation for the HRNN model is around 0.16 over the full sample (Table 6), it lowers to 0.12 over the sample excluding the GFC. Hence, the correlation between HRNN's forecasts and the true changes in the CPI was actually enhanced during the GFC. This may further explain HRNN's superiority during the GFC.

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

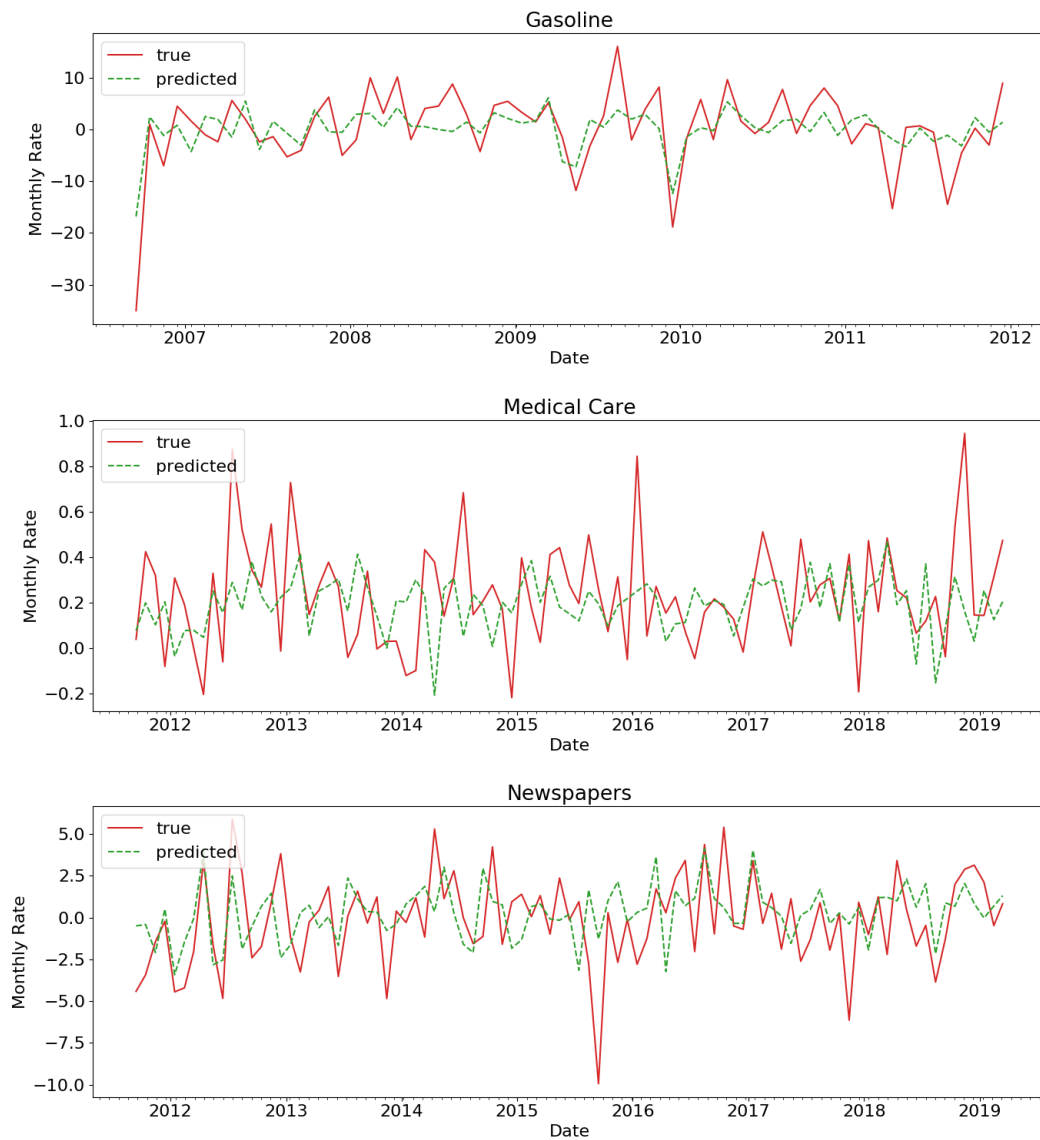


Table 6: Average Results on Disaggregated CPI Components Prior to The GFC

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	0.07	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	0.09	0.07
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	0.09	0.07
AR-GAP(4)	1.00	1.00	1.00	1.00	1.00	1.00	0.10	0.07
RW(4)	1.00	1.00	1.00	1.00	1.00	1.00	-0.03	-0.03
Phillips(4)	1.00	1.00	0.99	0.99	1.00	1.00	-0.02	-0.02
VAR(1)	1.04	1.04	1.04	1.05	1.05	1.06	0.04	0.03
VAR(2)	1.03	1.04	1.04	1.04	1.05	1.05	0.05	0.03
VAR(3)	1.03	1.03	1.03	1.04	1.04	1.05	0.06	0.03
VAR(4)	1.02	1.03	1.03	1.03	1.03	1.04	0.06	0.04
LSTAR($\rho = 4, c = 2, \gamma = 0.3$)	1.05	1.06	1.05	1.08	1.09	1.10	0.08	0.06
FC(4)	0.99	0.99	1.00	1.00	1.02	1.05	0.11	0.08
HRNN(1)	0.90	0.92	0.93	0.95	0.95	0.97	0.15	0.11
HRNN(2)	0.90	0.92	0.92	0.95	0.95	0.97	0.15	0.11
HRNN(3)	0.90	0.92	0.93	0.94	0.95	0.96	0.15	0.11
HRNN(4)	0.89	0.91	0.91	0.93	0.93	0.95	0.16	0.12
S-GRU(4)	1.05	1.09	1.09	1.10	1.09	1.10	0.09	0.07
I-GRU(4)	0.93	0.94	0.94	0.93	0.94	0.98	0.15	0.11
KNN-GRU(1)	0.94	0.96	0.96	0.96	0.97	0.98	0.10	0.07
KNN-GRU(2)	0.94	0.96	0.95	0.96	0.97	0.98	0.11	0.08
KNN-GRU(3)	0.93	0.96	0.95	0.96	0.96	0.98	0.11	0.08
KNN-GRU(4)	0.93	0.96	0.96	0.95	0.96	0.97	0.12	0.09

Notes: Average results across all 424 inflation indexes that make up the headline CPI. In contrast to Table 6, here we focus on results up to the GFC of 2008. The RMSE results are relative to the AR(1) model and normalized according to its results, i.e., $\frac{RMSE_{Model}}{RMSE_{AR(1)}}$.

7. Policy Implications

Policymakers already have at their disposal an extended set of predictive tools aimed at forecasting the headline inflation: survey data, expert forecasts, inflation swaps, economic and econometric models, etc. However, policy institutions lack models or data to assist CPI components' forecasting, which are key for a deeper understanding of underlying dynamics. Understanding disaggregated inflation trends may point to the nature of inflation pressures, their transitory factors (seasonal factors, energy, etc.), and other factors reflected in the headline inflation that influences market-makers and the conduct of monetary policy, among other decision-makers. Hence, our approach is geared towards CPI forecasting at the disaggregated level based on endogenous historical data. Indeed, our procedure's hierarchical characteristic is not targeted at forecasting the headline inflation, even if we perform well (Ibarra, 2012).

The business cycle plays an important role in inflation dynamics, in particular through specific product classes. Some CPI inflation dynamics are driven by components unrelated to central bank policy objectives i.e., food, and energy prices, for instance. Forecasting disaggregated CPI makes it possible to identify the sources and features of inflation pressures in the economy more accurately, thereby improving monetary

response efficiency. Indeed, forecasting sectoral inflation may improve the optimization problem faced by the central bank (Ida, 2020).

While similar headline inflation forecasts may correspond to various underlying economic factors, a disaggregated perspective allows understanding and analyzing the decomposition of these inflation forecasts at the sectoral or component level. Instead of disaggregating inflation to forecast the headline inflation (Stock and Watson, 2019), our approach allows policy and market makers to forecast specific sector and component prices, where information is less available: almost no component or sectoral-specific survey forecasts, expert forecast, or market-based forecasts exist. A central bank for example, could use such modeling features to avoid components affecting the inflation rate (military, food, cigarettes, and energy) that are unrelated to their primary inflation objectives. Economic policy recommendations at the sectoral level should also benefit from component-specific inflation forecasting, and market makers can better guide and tune their investment strategies (Swinkels, 2018).

Traditional approaches for inflation forecasting often employ a theoretical or a linear model which inescapably bias estimation and forecasting. Our novel approach may overcome the usual drawbacks which bias traditional forecasts, giving policy makers new insights from a “different angle”. Our structural approach includes explanatory variables with hierarchy in the disaggregated forecasts mitigating measurement errors at the component levels. Our model structure also attenuates component-specific residuals derived from each level and sector, which may improve forecasting performance. For all these reasons, we believe that HRNN can become a valuable tool for asset managers, policy institutions, and market makers lacking component-specific price forecasts critical to their decision processes.

The current debate on the Phillips curve have given rise to new models and data-based approaches aimed at checking the validity of the relation between inflation and unemployment and the changes in its dynamics, i.e., the Phillips curve flattening. However, the debate among academia and policymakers is still ongoing (Hooper et al., 2020). Disaggregating inflation is useful to understanding the microeconomic drivers of the Phillips curve (Luengo-Prado et al., 2018), and our method provides an efficient approach for inflation forecasting that may assist in better understanding this debate—how headline and sectoral inflation forecast dynamics are decomposed among components and levels of prices.

Last but not least, HRNN models have shown to perform well during the GFC compared to other models, which may be attributed to their nonlinear characteristics. This may prove beneficial to policymakers and private investors, who require reliable and precise forecasting tools during a crisis and high volatility periods, where linear or more standard models cannot adapt.

8. Conclusion

The HRNN model we presented to forecast disaggregated inflation changes significantly outperforms all the baseline models at predicting sectoral and component-specific prices. The model’s hierarchical nature enables information propagation from higher levels in the inflation hierarchy to improve predictions for the disaggregated components at lower levels. Extensive evaluations on the US CPI-U index against a wide array

of baselines demonstrate the HRNN model's superiority, especially at lower levels of the CPI hierarchy. The applications for policymakers that aim to forecast the relevant components of inflation and market makers interested in specific sectoral prices to guide their investment policies are promising. It is also crucial for central banks to easily identify and forecast the source of inflation pressures in the economy and for fixed-income investors to forecast sectoral-specific prices, while little information is available about level- or component-specific prices.

9. Acknowledgments

This research was supported by the Israel Science Foundation (Grant No. 2243/20).

References

- Almosova, A., Andresen, N., 2019. Nonlinear inflation forecasting with recurrent neural networks. Technical Report. European Central Bank (ECB).
- Aparicio, D., Bertolotto, M.I., 2020. Forecasting inflation with online prices. *International Journal of Forecasting* 36, 232–247.
- Athey, Susan, 2018. The impact of machine learning on economics, in: *The Economics of Artificial Intelligence: An Agenda*. University of Chicago Press, pp. 507–547.
- Atkeson, A., Ohanian, E. L., 2001. Are phillips curves useful for forecasting inflation? *Federal Reserve Bank of Minneapolis Quarterly Review* 25, 2–11.
- Bernanke, B.S., Laubach, T., Mishkin, F.S., Posen, A.S., 2018. *Inflation targeting: lessons from the international experience*. Princeton, NJ: Princeton University Press.
- Chakraborty, C., Joseph, A., 2017. Machine learning at central banks. Bank of England working papers, number 674 .
- Chen, X., Racine, J., Swanson, N.R., 2001. Semiparametric arx neural-network models with an application to forecasting inflation. *IEEE Transactions on Neural Networks* 12, 674–683.
- Choudhary, M.A., Haider, A., 2012. Neural network models for inflation forecasting: an appraisal. *Applied Economics* 44, 2631–2635.
- Chung, J., Gulcehre, C., Cho, K., Bengio, Y., 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555* .
- Coulombe, P.G., Leroux, M., Stevanovic, D., Surprenant, S., et al., 2020. How is Machine Learning Useful for Macroeconomic Forecasting? *Papers 2008.12477*. *arXiv.org*.
- Dey, R., Salemt, F.M., 2017. Gate-variants of gated recurrent unit (gru) neural networks, in: *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, IEEE. pp. 1597–1600.
- Diebold, F.X., Mariano, R.S., 1995. Comparing Predictive Accuracy. *Journal of Business & Economic Statistics* 13, 253–263.
- van Dijk, D., Terasvirta, T., Franses, P.H., 2002. Smooth Transition Autoregressive Models — A Survey Of Recent Developments. *Econometric Reviews* 21, 1–47.
- Faust, J., Wright, J.H., 2013. Forecasting Inflation, in: Elliott, G., Granger, C., Timmermann, A. (Eds.), *Handbook of Economic Forecasting*. Elsevier. volume 2 of *Handbook of Economic Forecasting*. chapter 1, pp. 2–56.
- Friedman, M., 1961. The Lag in Effect of Monetary Policy. *Journal of Political Economy* 69, 447–447.
- Gilchrist, S., Schoenle, R., Sim, J., Zakrajšek, E., 2017. Inflation dynamics during the financial crisis. *American Economic Review* 107, 785–823.
- Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. *Neural Computation* 9, 1735–1780.
- Hooper, P., Mishkin, F.S., Sufi, A., 2020. Prospects for inflation in a high pressure economy: Is the Phillips curve dead or is it just hibernating? *Research in Economics* 74, 26–62.
- Ibarra, R., 2012. Do disaggregated CPI data improve the accuracy of inflation forecasts? *Economic Modelling* 29, 1305–1313.
- Ida, D., 2020. Sectoral inflation persistence and optimal monetary policy. *Journal of Macroeconomics* 65.

- Lipton, Z.C., Berkowitz, J., Elkan, C., 2015. A critical review of recurrent neural networks for sequence learning. CoRR.
- Luengo-Prado, M.J., Rao, N., Sheremirov, V., 2018. Sectoral inflation and the Phillips curve: What has changed since the Great Recession? *Economics Letters* 172, 63–68.
- Makridakis, S., Assimakopoulos, V., Spiliotis, E., 2018. Objectivity, reproducibility and replicability in forecasting research. *International Journal of Forecasting* 34, 835–838.
- Makridakis, S., Spiliotis, E., Assimakopoulos, V., 2020. The m4 competition: 100,000 time series and 61 forecasting methods. *International Journal of Forecasting* 36, 54–74.
- Mandic, D., Chambers, J., 2001. *Recurrent neural networks for prediction: learning algorithms, architectures and stability*. Wiley.
- McAdam, P., McNelis, P., 2005. Forecasting inflation with thick models and neural networks. *Economic Modelling* 22, 848–867.
- Medeiros, M., Vasconcelos, G., Veiga, A., Zilberman, E., 2021. Forecasting inflation in a data-rich environment: the benefits of machine learning methods. *Journal of Business & Economic Statistics* 39.
- Mullainathan, S., Spiess, J., 2017. Machine learning: an applied econometric approach. *Journal of Economic Perspectives* 31, 87–106.
- Nakamura, E., 2005. Inflation forecasting using a neural network. *Economics Letters* 86, 373–378.
- Ramachandran, P., Zoph, B., Le, Q.V., 2017. Searching for activation functions. CoRR abs/1710.05941.
- Stock, J.H., Watson, M.W., 2007. Why has us inflation become harder to forecast? *Journal of Money, Credit and Banking* 39, 3–33.
- Stock, J.H., Watson, M.W., 2010. Modeling inflation after the crisis. Technical Report. National Bureau of Economic Research.
- Stock, J.H., Watson, M.W., 2019. Trend, Seasonal, and Sectoral Inflation in the Euro Area. Working Papers Central Bank of Chile 847. Central Bank of Chile.
- Swinkels, L., 2018. Simulating historical inflation-linked bond returns. *Journal of Empirical Finance* 48, 374–389.
- Woodford, M., 2012. Inflation targeting and financial stability. *Sveriges Riksbank Economic Review* 1, 7–32.
- Yu, Y., Si, X., Hu, C., Zhang, J., 2019. A review of recurrent neural networks: LSTM cells and network architectures. *Neural Computation* 31, 1235–1270.
- Zahara, S., Ilmiddaviq, M., et al., 2020. Consumer price index prediction using long short term memory (lstm) based cloud computing, in: *Journal of Physics: Conference Series*, IOP Publishing. p. 012022.

Appendix A Additional Tables and Figures

Table 7: Indexes Level 0 And 1

Level	Index	Parent
0	All items	-
1	All items less energy	All items
1	All items less food	All items
1	All items less food and energy	All items
1	All items less food and shelter	All items
1	All items less food, shelter, and energy	All items
1	All items less food, shelter, energy, and used cars and trucks	All items
1	All items less homeowners costs	All items
1	All items less medical care	All items
1	All items less shelter	All items
1	Apparel	All items
1	Apparel less footwear	All items
1	Commodities	All items
1	Commodities less food	All items
1	Durables	All items
1	Education and communication	All items
1	Energy	All items
1	Entertainment	All items
1	Food	All items
1	Food and beverages	All items
1	Fuels and utilities	All items
1	Household furnishings and operations	All items
1	Housing	All items
1	Medical care	All items
1	Nondurables	All items
1	Nondurables less food	All items
1	Nondurables less food and apparel	All items
1	Other goods and services	All items
1	Other services	All items
1	Recreation	All items
1	Services	All items
1	Services less medical care services	All items
1	Services less rent of shelter	All items
1	Transportation	All items
1	Utilities and public transportation	All items

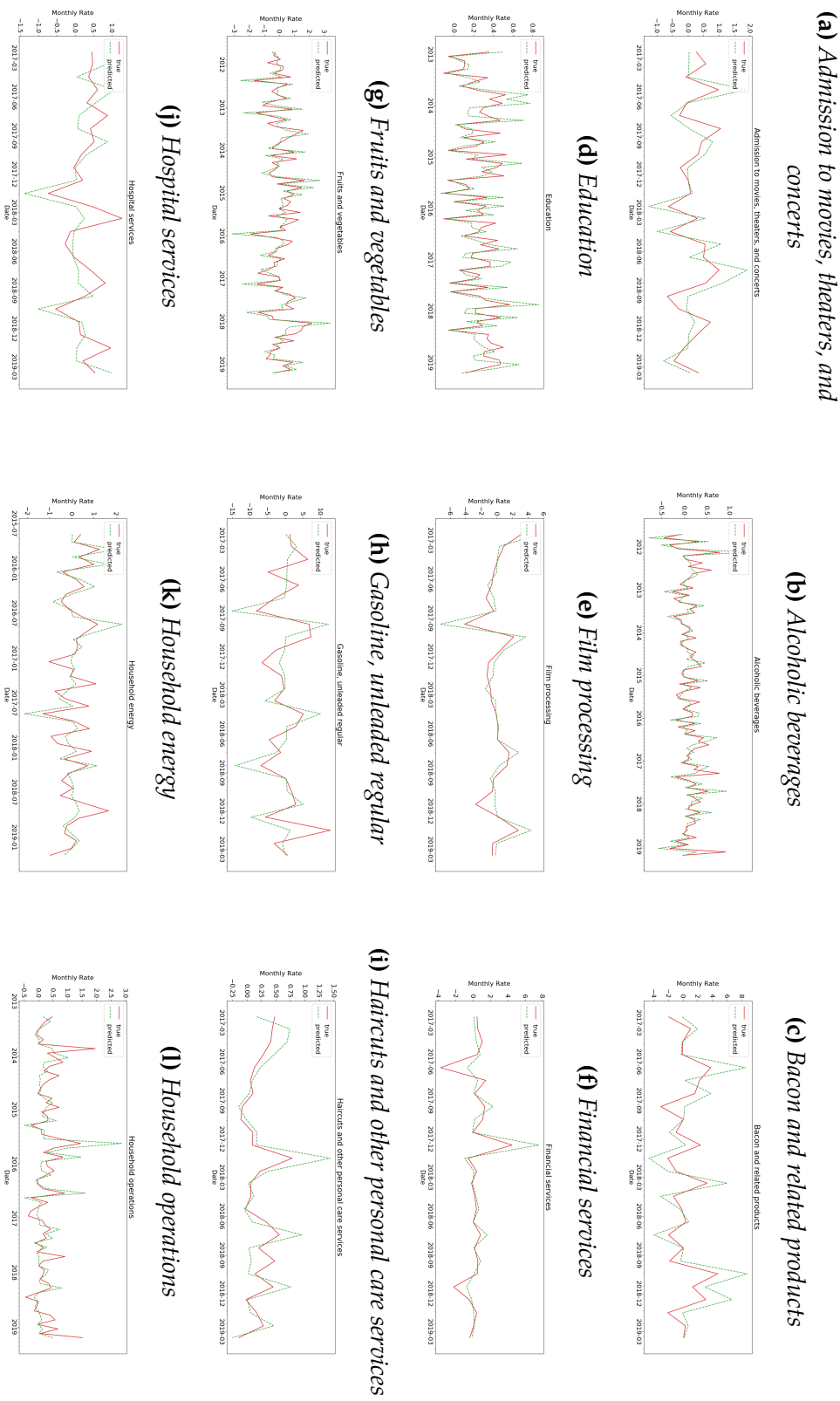
Note: Levels and Parents of Indexes might change through time

Table 8: Indexes Level 2

Level	Index	Parent
2	All items less food and energy	All items less energy
2	Apparel commodities	Apparel
2	Apparel services	Apparel
2	Commodities less food	Commodities
2	Commodities less food and beverages	Commodities
2	Commodities less food and energy commodities	All items less food and energy
2	Commodities less food, energy, and used cars and trucks	Commodities
2	Communication	Education and communication
2	Domestically produced farm food	Food and beverages
2	Education	Education and communication
2	Energy commodities	Energy
2	Energy services	Energy
2	Entertainment commodities	Entertainment
2	Entertainment services	Entertainment
2	Food	Food and beverages
2	Food at home	Food
2	Food away from home	Food
2	Footwear	Apparel
2	Fuels and utilities	Housing
2	Homeowners costs	Housing
2	Household energy	Fuels and utilities
2	Household furnishings and operations	Housing
2	Infants' and toddlers' apparel	Apparel
2	Medical care commodities	Medical care
2	Medical care services	Medical care
2	Men's and boys' apparel	Apparel
2	Nondurables less food	Nondurables
2	Nondurables less food and apparel	Nondurables
2	Nondurables less food and beverages	Nondurables
2	Nondurables less food, beverages, and apparel	Nondurables
2	Other services	Services
2	Personal and educational expenses	Other goods and services
2	Personal care	Other goods and services
2	Pets, pet products and services	Recreation
2	Photography	Recreation
2	Private transportation	Transportation
2	Public transportation	Transportation
2	Rent of shelter	Services
2	Services less energy services	All items less food and energy
2	Services less medical care services	Services
2	Services less rent of shelter	Services
2	Shelter	Housing
2	Tobacco and smoking products	Other goods and services
2	Transportation services	Services
2	Video and audio	Recreation
2	Women's and girls' apparel	Apparel

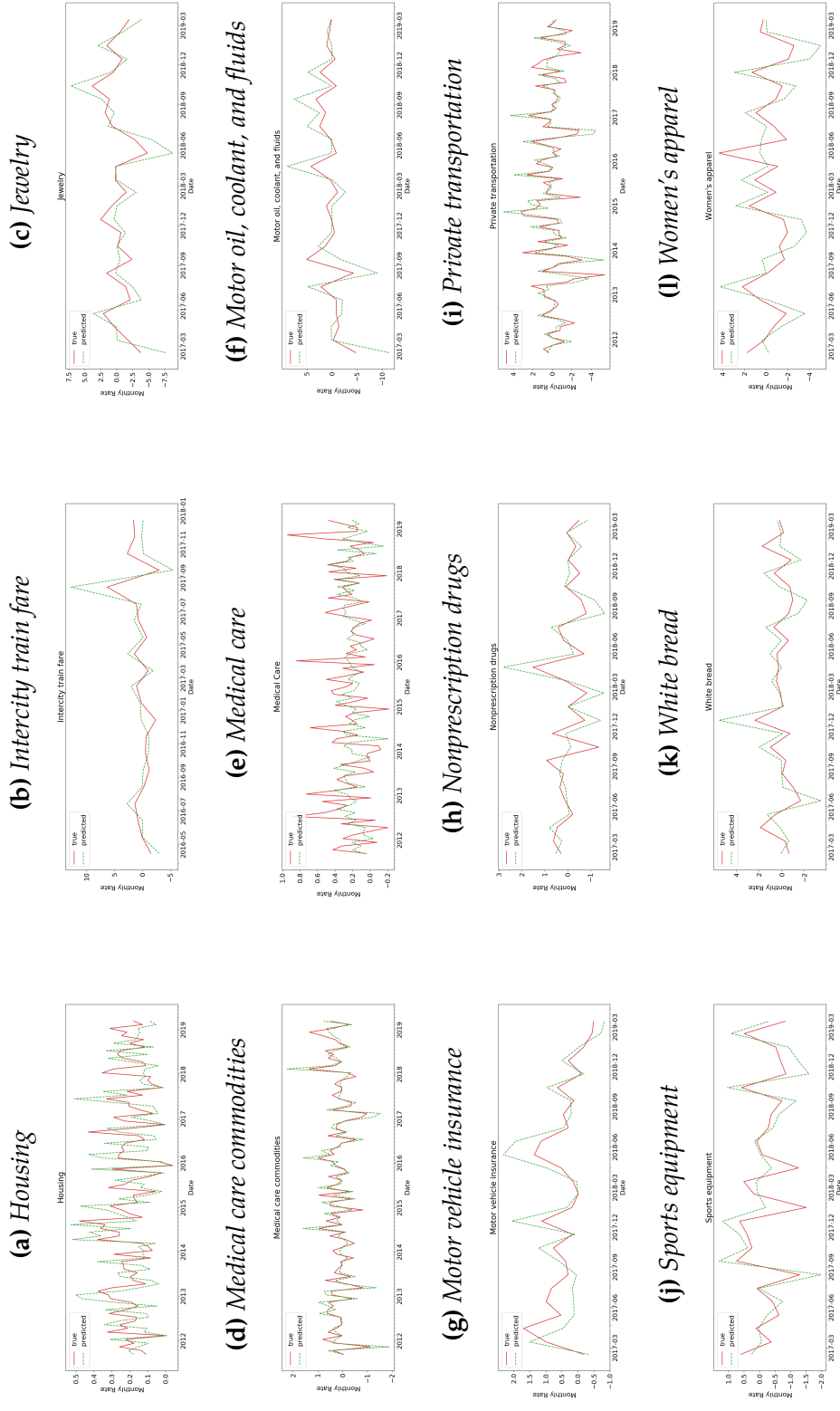
Note: Levels and Parents of Indexes have changed over the years.

Figure 8. Additional Examples of HRNN(4) predictions for disaggregated indexes



Figures 1-12, indexes were selected from different hierarchies and sectors

Figure 9. Additional Examples of HRNN(4) predictions for disaggregated indexes (different hierarchies and sectors)



Figures 13-24, indexes were selected from different hierarchies and sectors