Unsupervised Real Time Anomaly Detection for Streaming Data

César Augusto Velásquez Pineda

26. Januar 2022

Overview and Problem Definition

Streaming Data

$X_t = ..., x^{(t-3)}, x^{(t-2)}, x^{(t-1)}, x^{(t)}$ at time t,
$X_{t+1} = ..., x^{(t-3)}, x^{(t-2)}, x^{(t-1)}, x^{(t)}, x^{(t+1)}$ at time $t+1$. **Data not analyzed in batches but updated constantly**.

Issues for Anomaly Detection in Streaming Data

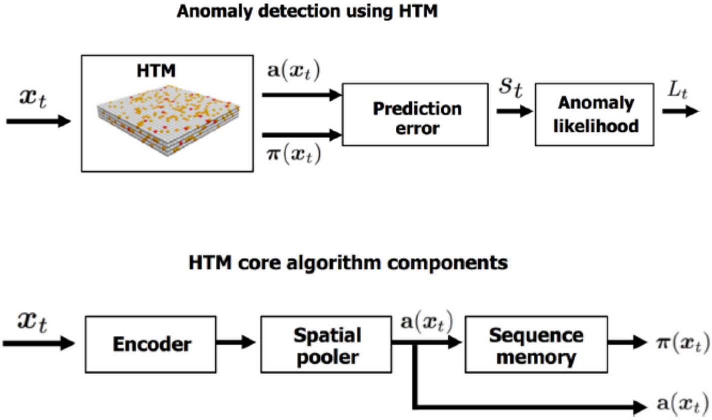
1. Online processing
2. Adaptiveness: algorithm is continuous and not dependent of the entire dataset
3. Human interaction in general implausible. Unsupervised Learning
4. Adaptability to concept drift
5. Minimization of false positives and false negatives

Overview and Problem Definition

Anomaly detection streaming data paradigms

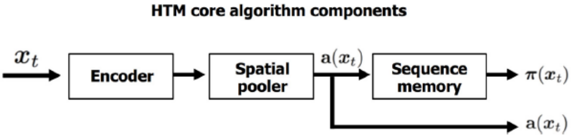
- **Forecast**
  - **HTM: Hierarchical Temporal Memory**
  - RNNs: Recurrent Neural Networks
  - LSTMs: Long Short term Memory
  - Other time series or sliding window based approaches
- **Reconstruction**
  - Autoencoders: Convolutional Autoencoders, LSTM Autoencoders

The HTM Algorithm



Taken from: (Ahmad et al., 2017)

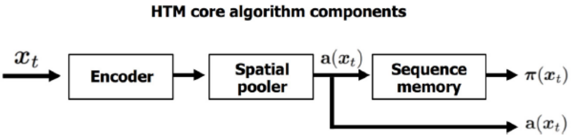
The HTM Algorithm: Encoder and Spatial Pooling

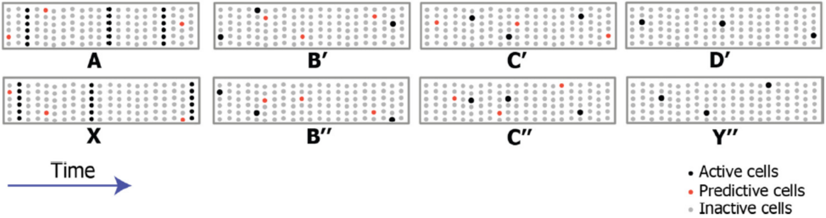


Taken from: (Ahmad et al., 2017)

Input	Encoding			Spatial Pooling		
1	0	1	1	1	0	0
2	1	0	0	0	1	0
1	0	1	1	1	0	0
2	1	0	0	0	1	0
1	0	1	1	1	0	0

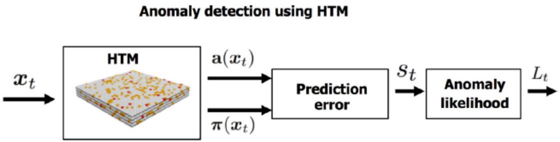
The HTM Algorithm: Sequence Memory





Taken from: (Ahmad et al., 2017)

The HTM Algorithm: Prediction Error



Taken from: (Ahmad et al., 2017)

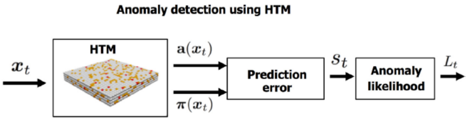
Prediction Error

$$S_t = 1 - \frac{\pi(x_{t-1}).a(x_t)}{|a(x_t)|}$$

0 if $\pi(x_{t-1})$ is exactly the same as $a(x_t)$ i.e. predicted equals observation.

1 if $\pi(x_{t-1})$ have no bits in common with $a(x_t)$ i.e. if they are orthogonal (no prediction matches its observation).

The HTM Algorithm: Anomaly Likelihood



Taken from: (Ahmad et al., 2017)

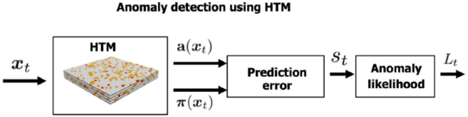
Anomaly Likelyhood

Intuitively: probability of an event being anomalous or not (lies within 0 and 1). Its calculation is based on a rolling normal distribution over the last predicted error values within last W observations.

$$\mu_t = \frac{\sum_{i=0}^{i=W-1} s_{t-i}}{W}$$

$$\sigma_t^2 = \frac{\sum_{i=0}^{i=W-1} (s_{t-i} - \mu_t)^2}{W-1}$$

The HTM Algorithm: Anomaly Likelihood



Taken from: (Ahmad et al., 2017)

Anomaly Likelyhood

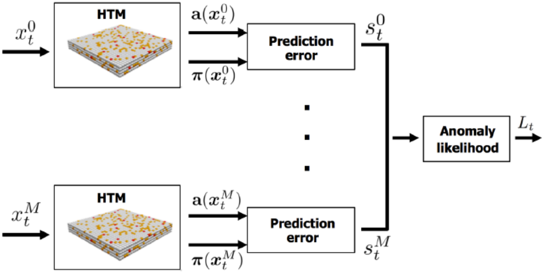
Long term and short term prediction error distributions are compared. Where the short term mean is (being $W' << W$):

$$\tilde{\mu_t} = \frac{\sum_{i=0}^{i=W'-1} s_{t-i}}{W'}$$

And the anomaly likelihood calculated as:

$$L_t = 1 - Q(\frac{\tilde{\mu_t} - \mu_t}{\sigma_t})$$

The HTM Algorithm: Multivariate Time Series Data



Taken from: (Ahmad et al., 2017)

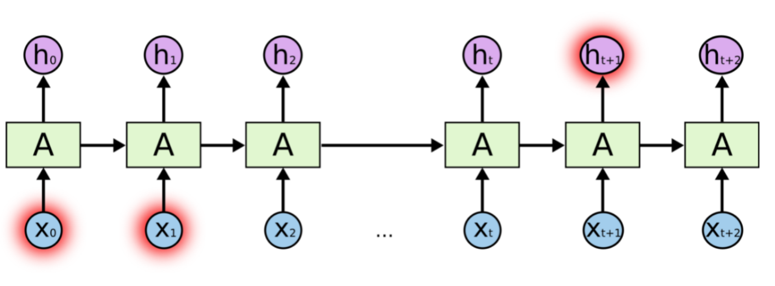
Simultaneous data sources

Assuming that the underlying distributions of each of the $M+1$ prediction errors are independent:

$$L_t = 1 - \prod_{i=0}^{M-1} Q\left(\frac{\tilde{\mu}_{t_i} - \mu_{t_i}}{\sigma_{t_i}}\right)$$

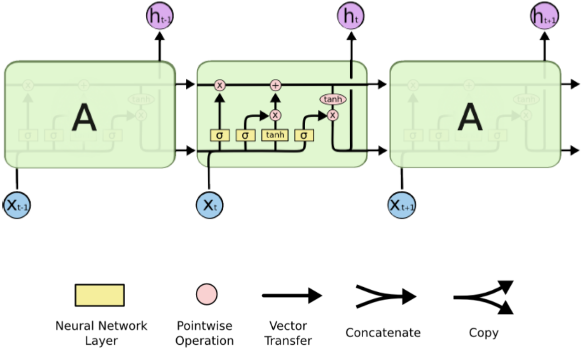
Related Approaches: Long Short Term Memory (LSTM)

Recurrent Neural Networks



Taken from: (Olah, 2015)

Related Approaches: Long Short Term Memory (LSTM)

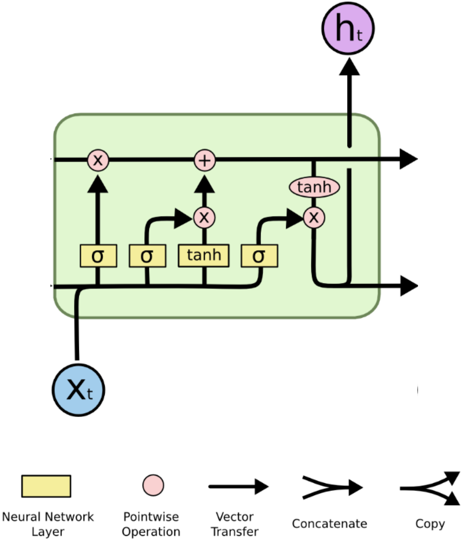


Taken from: (Olah, 2015)

LSTM Neural Networks

- LSTMs add or remove information to the **cell state** with the **gates** composed of sigmoid and multiplication
- Gates are: forget gate, input gate and output gate

Related Approaches: Long Short Term Memory (LSTM)



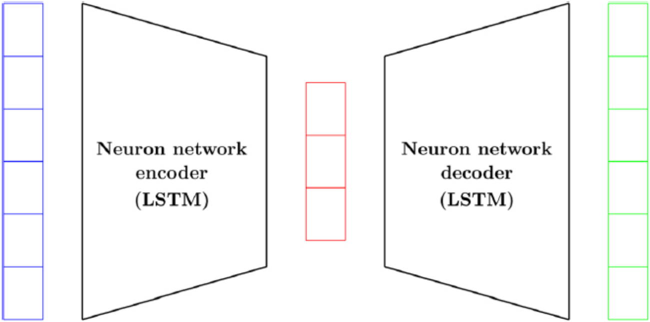
Taken from: (Olah, 2015)

Related Approaches: Long Short Term Memory (LSTM)

LSTM for Anomaly Detection

- Prediction approach: Forecasting
- Learning mechanism: Supervised
- Training mechanism: per Batches
- Easy handle of multidimensional data
- Not so fast adaptation to concept drift

Related Approaches: Autoencoders

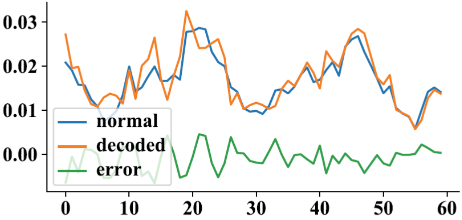


Taken from: (Nguyen et al., 2020)

Autoencoders

- Reconstruction vs forecast (LTSM, HTM) approach
- Feature selection and/or dimensionality reduction with potential highly nonlinear functions

Related Approaches: Autoencoders



Taken from: (Agarwal et al., 2021)

Autoencoders for Anomaly Detection

- Prediction approach: Reconstruction
- Learning mechanism: Unsupervised
- Training mechanism: per Batches
- Easy handle of multidimensional data
- Not so fast adaptation to concept drift

Evaluation Metrics

Confusion Matrix

		True Class	
		0	1
Estimated	0	TN	FN
Class	1	FP	TP

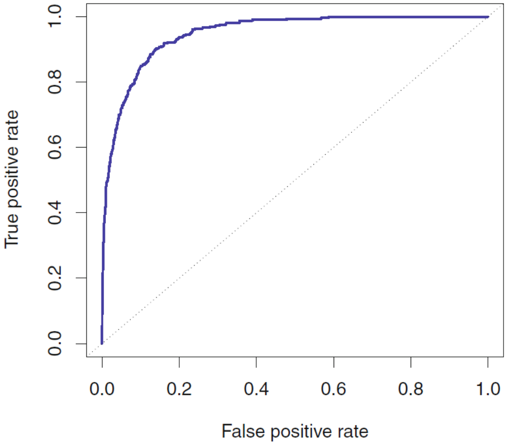
Evaluation Metrics

- Total error rate: $F = \frac{FP+FN}{n}$
- Sensitivity (True Positive Rate): $TPR = \frac{TP}{TP+FN}$
- 1-Specificity (False Positive Rate): $FPR = \frac{FP}{TN+FP}$
- Precision: $\frac{TP}{TP+FP}$
- F1-Score: $\frac{2}{\frac{1}{Precision}+\frac{1}{Recall}} = 2\frac{Precision*Recall}{Precision+Recall}$ (Harmonic Mean)

Evaluation Metrics

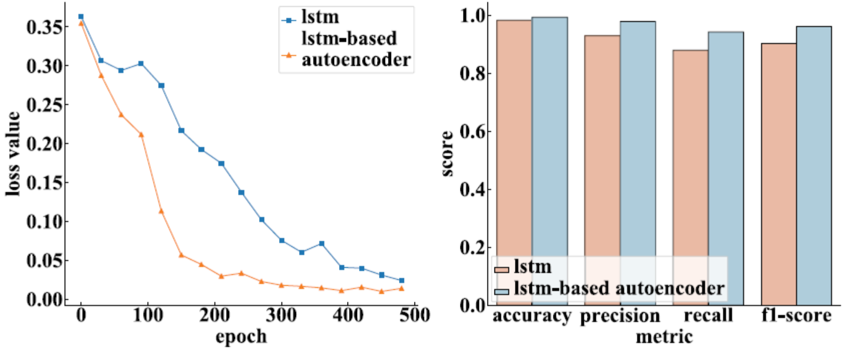
ROC Curve

Points on the diagonal are randomly assigned



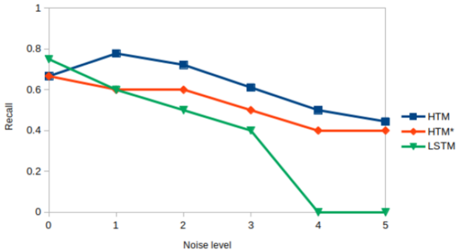
Taken from: (James et al., 2017)

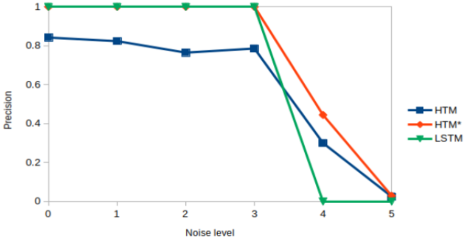
Experimental results



Taken from: (Agarwal et al., 2021)

Experimental results





Taken from: (Haddad and Piehl, 2019)

Comparison and Discussion

Discussion

- HTMs and LSTMs proceed under forecasting paradigm whereas Autoencoders are reconstructive
- HTMs have the highest capacity of updating themselves fast in an online learning application (concept drift)
- LSTMs learn in a supervised manner
- Typically, LSTMs and Autoencoders require more data
- HTMs appear to achieve a higher recall in several applications, whereas LSTMs and Autoencoders are more precise
- LSTMs and Autoencoders have a more flexible and built-in way to adapt to several related data sources

References

Agarwal, R., Nagpal, T., Roy, D., and D, A. (2021). A novel anomaly detection for streaming data using lstm autoencoders.

Ahmad, S., Lavin, A., Purdy, S., and Agha, Z. (2017). Unsupervised real-time anomaly detection for streaming data.

Haddad, J. and Piehl, C. (2019). *Unsupervised anomaly detection in time series with recurrent neural networks*.

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2017). *An Introduction to Statistical Learning*. Springer.

Nguyen, H., Tran, K., Thomassey, S., and Hamad, M. (2020). Forecasting and anomaly detection approaches using lstm and lstm autoencoder techniques with the applications in supply chain management.

Olah, C. (2015). Understanding lstm networks. `http://colah.github.io/posts/2015-08-Understanding-LSTMs/`. Accessed: 2022-01-05.