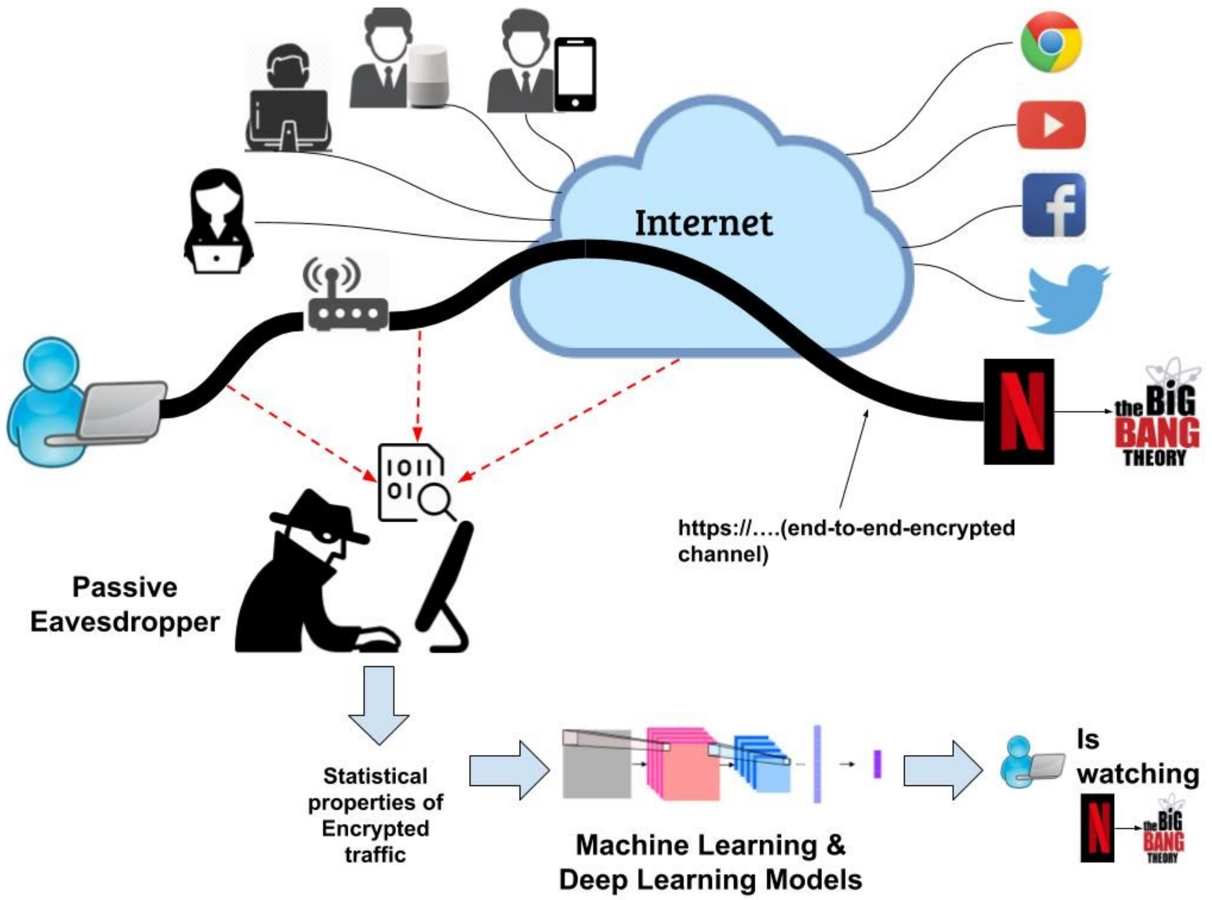
Understanding Traffic Fingerprinting CNNs

Thilini Dahanayaka,[1] Guillaume Jourjon,[2] Suranga Seneviratne[1]

[1]University of Sydney

[2]Data61-CSIRO

Motivation – Vulnerabilities of End-to-End Encryption



Side channel information leaks
(packet size, packet timing etc.)

- Websites visited
- Videos streamed
- Messenger app activities
- …

Motivation – Traffic Fingerprinting Attacks

- Most recent traffic fingerprinting attacks leverage **deep learning models**
    - E.g. MLPs, CNNs, RNNs
    - CNNs outperform other deep learning models (*in almost all the studies*)
- Applications of traffic fingerprinting:
    - Network measurements / performance analysis
    - Network surveillance
    - Network censorship
- Understanding the inner workings of traffic fingerprinting attacks is essential to:
    - Improve the attacks / better network intelligence
    - Develop protocols resilient to traffic fingerprinting

Our Contributions

- We methodically dissect network traffic fingerprinting CNNs to understand their inner workings.

- We use three existing datasets to:
    - Characterize patterns that traffic fingerprinting CNNs look for at different depths of the network.
    - Provide insights on parts of the input traces that contribute significantly towards the classifier's decision.
    - Show traffic fingerprinting CNNs demonstrate transfer learning capabilities.
    - Show why CNNs outperform RNNs at traffic fingerprinting

Nguyen et al. 2016, Arxiv[1]

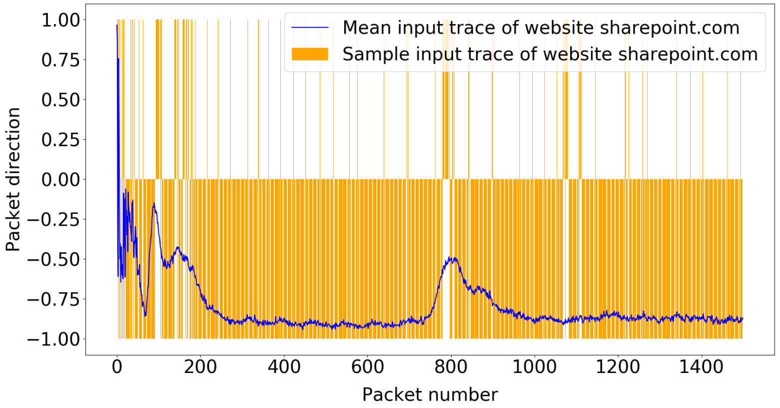
Nguyen, Anh, Jason Yosinski, and Jeff Clune. "Multifaceted feature visualization: Uncovering the different types of features learned by each neuron in deep neural networks." arXiv preprint arXiv:1602.03616 (2016).

Datasets

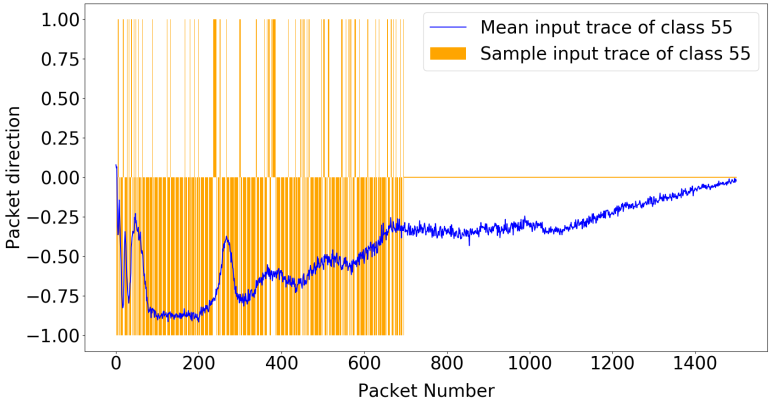
- Three publicly available datasets:

Dataset	Source	Traffic category	No. of classes	Traces per class	Training set size	Test set size	Validation set size
AWF	Rimmer et al. [NDSS ‘18]	Website visits	200	2,500	350,000	75,000	75,000
DF	Sirinam et al. [CCS ‘18]	Website visits	95	1,000	76,000	9,500	9,500
DC	Li et al. [NCA ‘18]	Video streaming	10	320	2,510	640	50

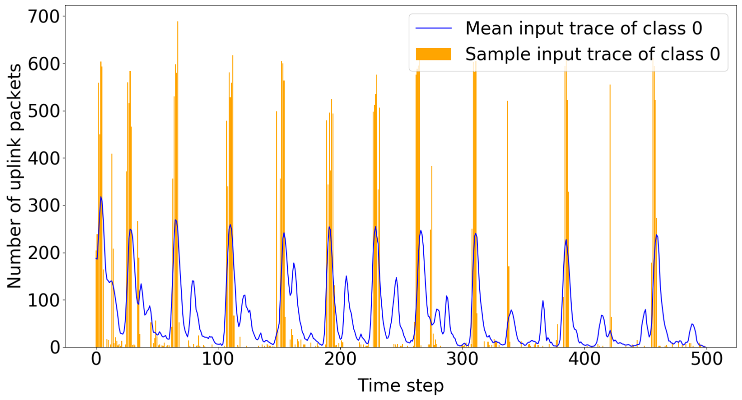
Datasets – Example Data Points



AWF



DF

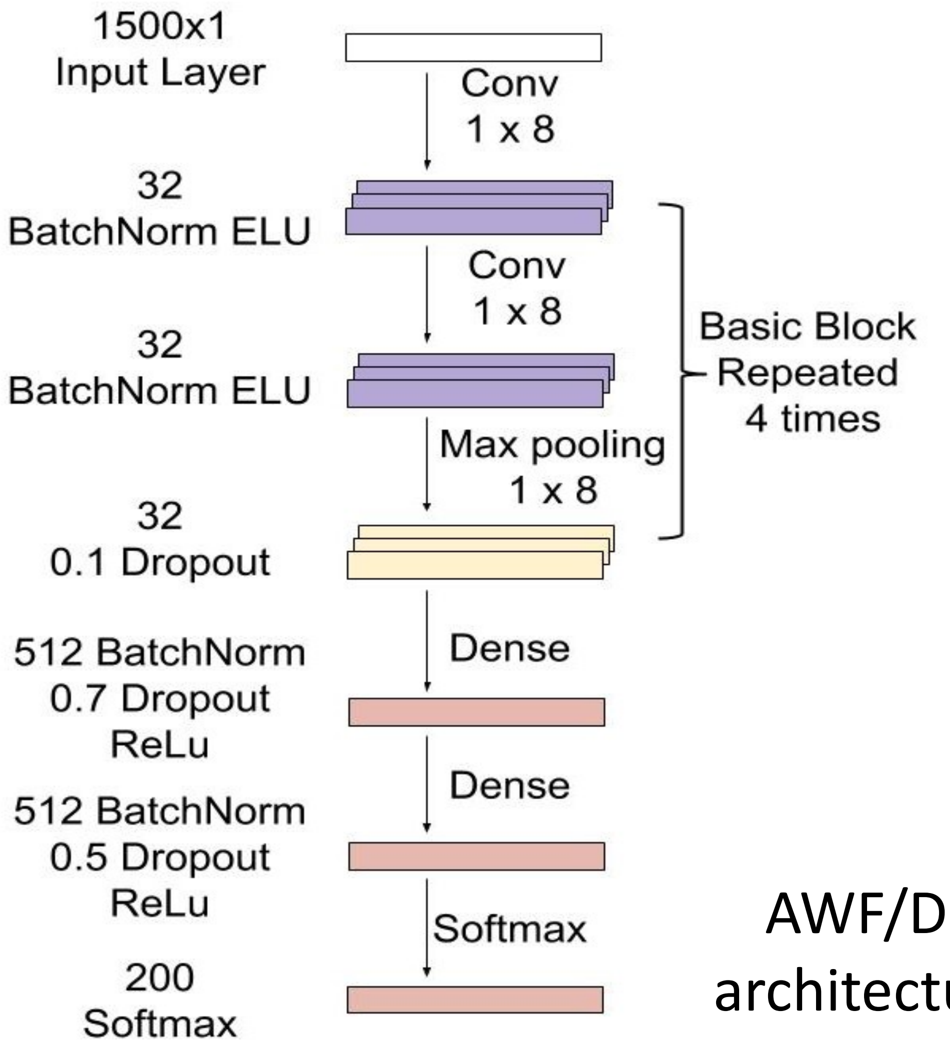


DC

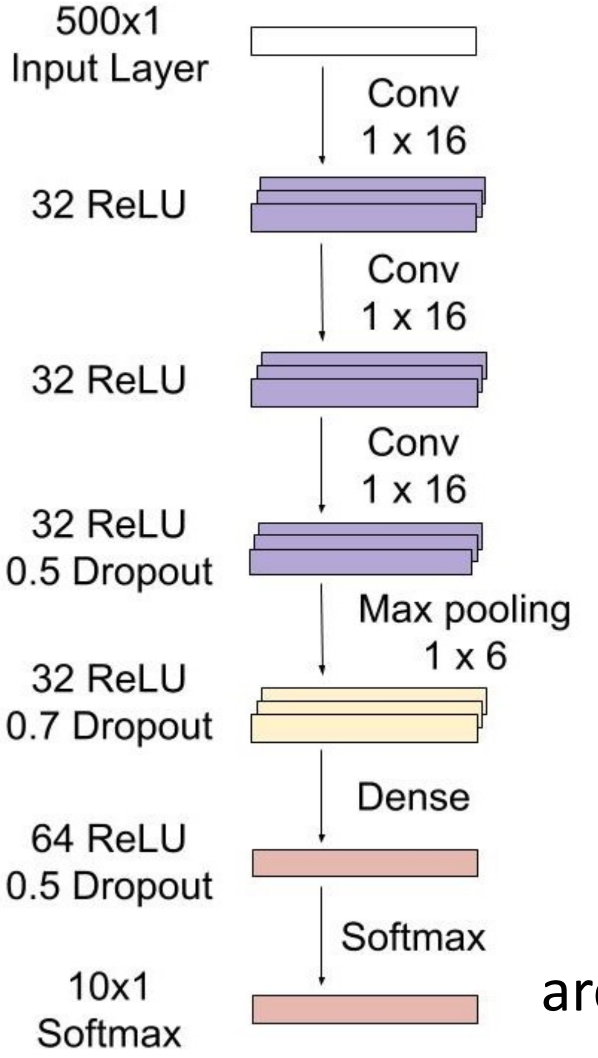
- AWF and DF Datasets
  - +1s in the initial part (HTTP GET requests sent to the web server)
  - Middle and later parts are mostly -1s (downloading website content)

- DC Dataset
  - Sequence of integers between 0 and 736
  - Periodic patterns that correspond to DASH chunk fetching.

CNN Architectures



AWF/DF architecture



DC architecture

Key Idea - Visualizing 1-D Convolution Filters

- **1-D Convolution**:

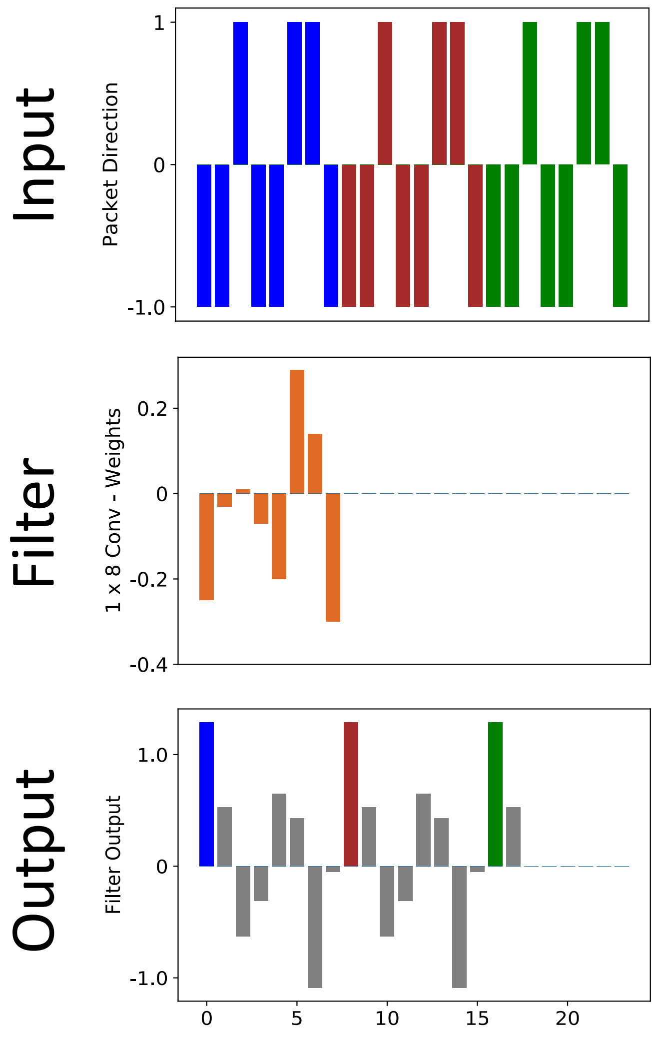
$$\{x_1, x_2, x_3, \dots, xN\} * \{w_1, w_2, w_3, \dots, wN\} = \sum_{i=1}^{i=N} w_i x_i + b$$

Where $\{x_1, x_2, x_3, \dots, xN\}$ is input sequence, $\{w_1, w_2, w_3, \dots, wN\}$ is the filter and b is bias term

- 1-D convolution is analogous to ***cross correlation***

- Convolution between an input and a filter can be seen as finding sections of the input that match the pattern of the filter

Visualizing 1-D Convolution Filters



AWF/DF Model:

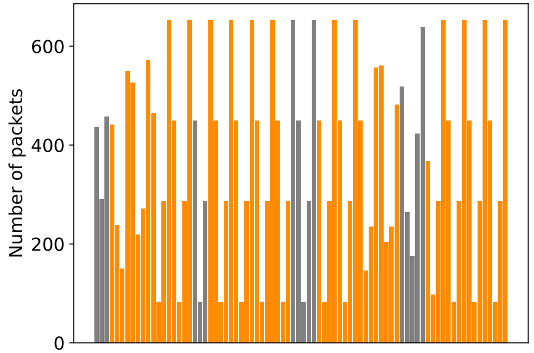
- Input is +1 or -1 only
- Output value takes *maximum possible value* if the signs (positive or negative) of the input is same as that of the weights in the filter for all positions.
- Output will take the *least possible value (largest negative)* when the signs of the values of the input and the filter are exact opposites.

This intuition can help identify patterns learnt by filters of 1st layer only.

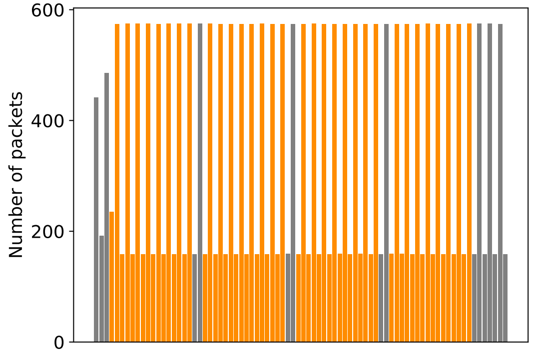
Visualizing DC model filters with ***Gradient Ascent***

Gradient Ascent: Optimizes noisy input to maximize mean activation for filter considered

Layer 1

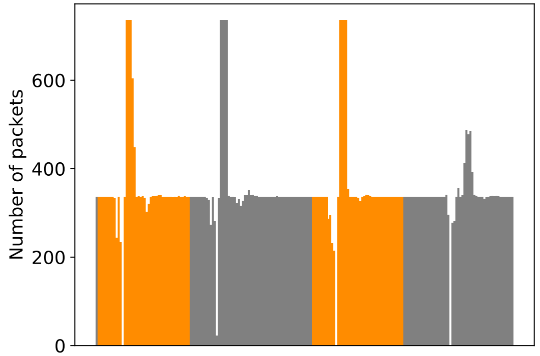


(a)

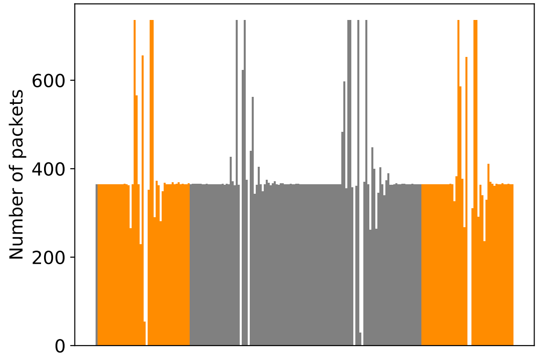


(b)

Layer 3



(a)



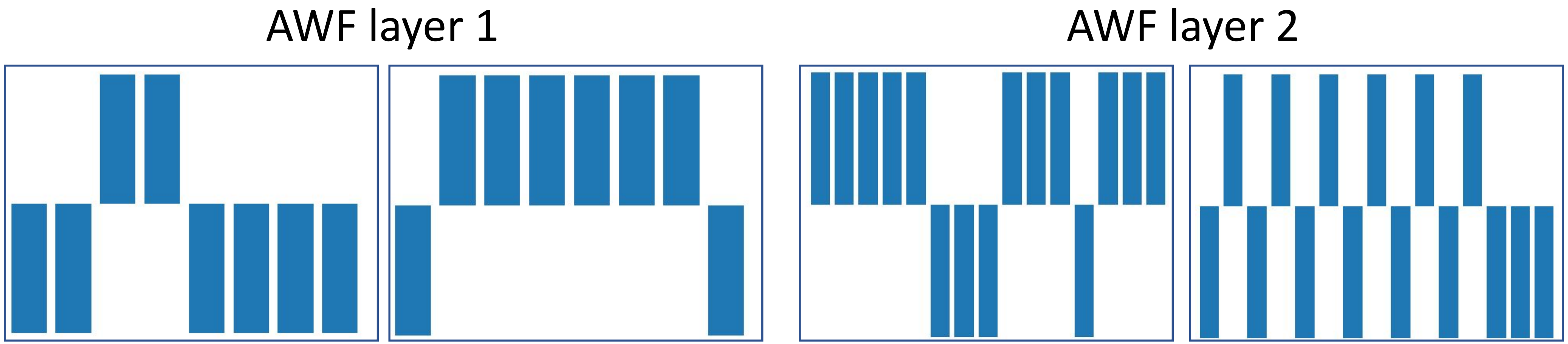
(b)

- **Receptive field**: Section of original input that affects given position of output
- Receptive fields are highlighted in orange.
- Repetitive high activations suggest that video fingerprinting CNNs respond to bursts in their inputs with specific shapes and lengths.

Visualizing AWF and DF Model Filters

- Gradient Ascent method does not work for AWF and DF methods as each step would flip the sign of the input value without converging.

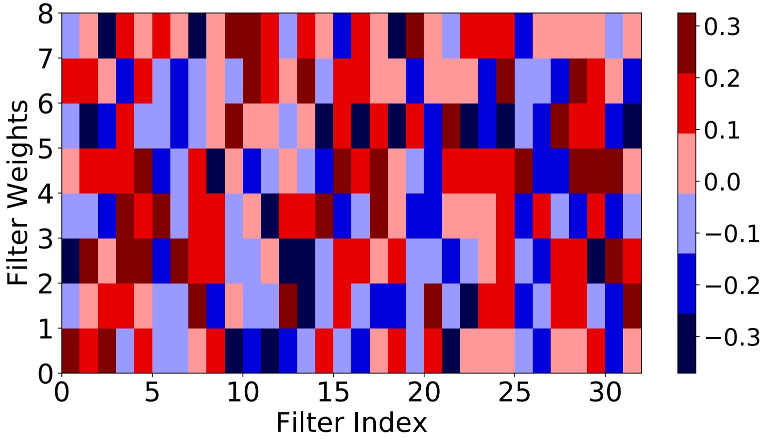
- Use the input trace with the highest filter activation value from the training set to approximate the features.



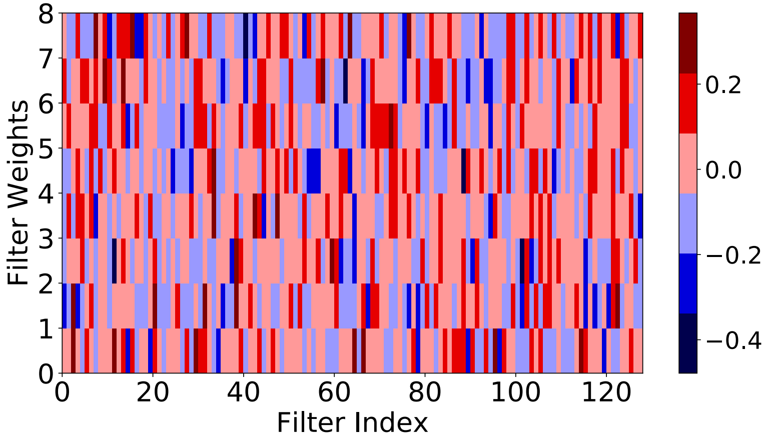
- All filters look for specific patterns with combinations of +1s and -1s in the input.

RQ 1: What patterns do traffic fingerprinting CNNs learn?

- Visualize the filter weight distribution of all filters for selected convolutional layers



AWF layer 1

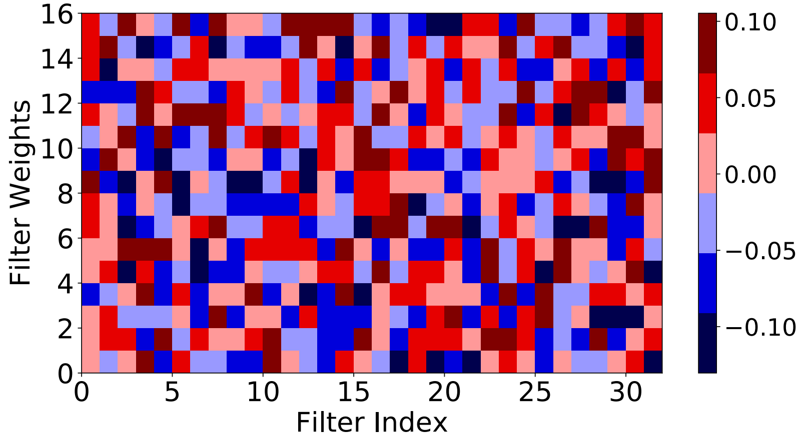


AWF layer 2

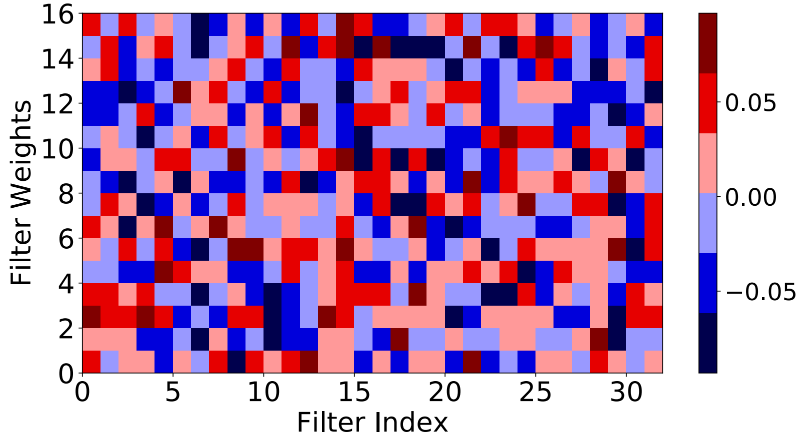
- Filters look for sequences with a combination of +1s and -1s (uploads and downloads).
- Do not look for continuous sequences of +1s and -1s.
  - Counterintuitive – many existing defence mechanisms add noise when there is no activity.

RQ 1: What patterns do traffic fingerprinting CNNs learn?

- Visualize the filter weight distribution of all filters for selected convolutional layers



DC layer 1



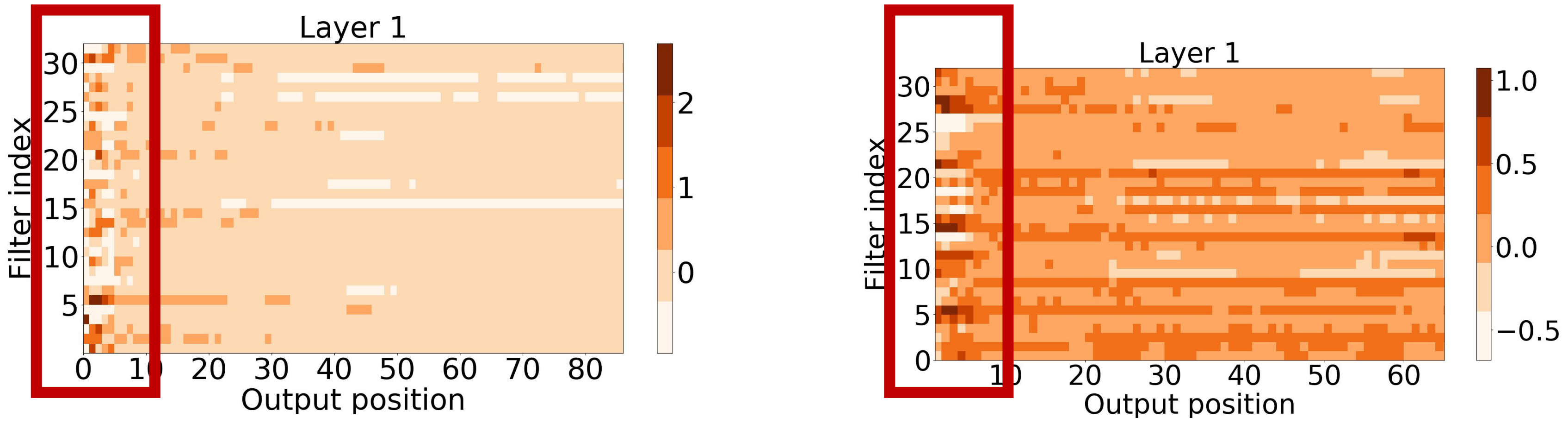
DC layer 2

- Filters look for sequences of different number of packets per unit time

- Filters focus not only on the envelope of the burst signal, but also on the finer sub-bursts associated with a major burst.

RQ 2: Is there any part of the trace CNNs focus more on?

- Visualize activation maps for each layer from 500 random samples



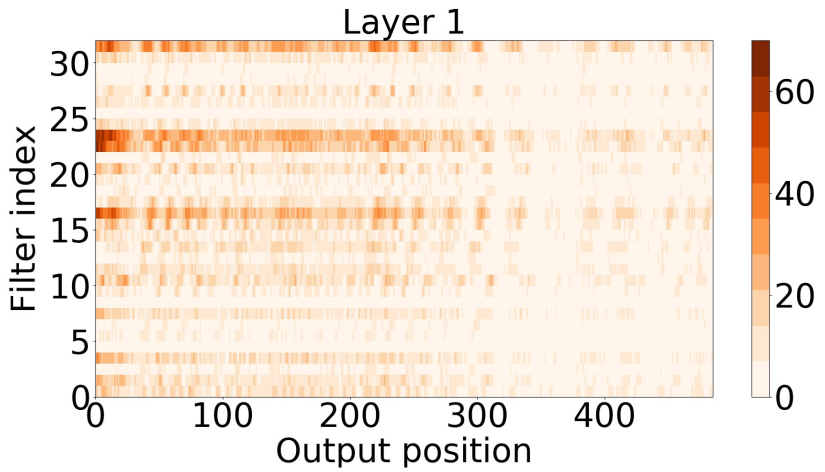
AWF model

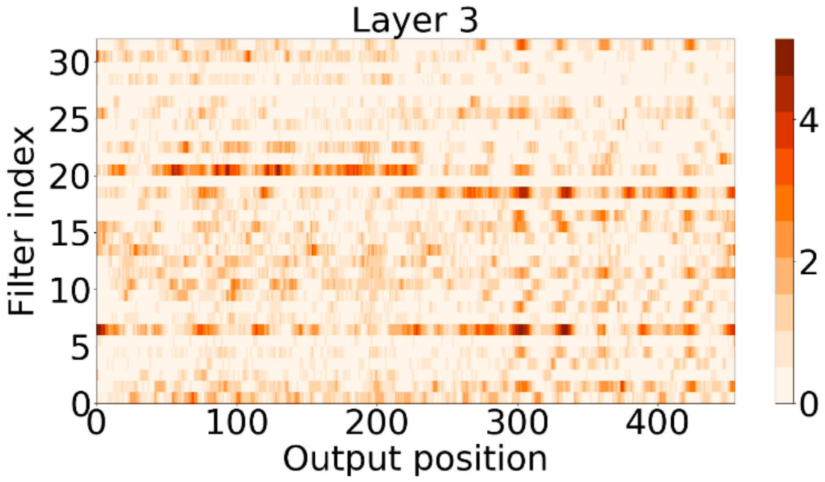
DF model

- For website visits, highest filter activations correspond to the beginning of trace.
- More defensive noise must be added at the beginning

RQ 2: Is there any part of the trace CNNs focus more on?

Visualize activation maps for each layer from 500 random samples



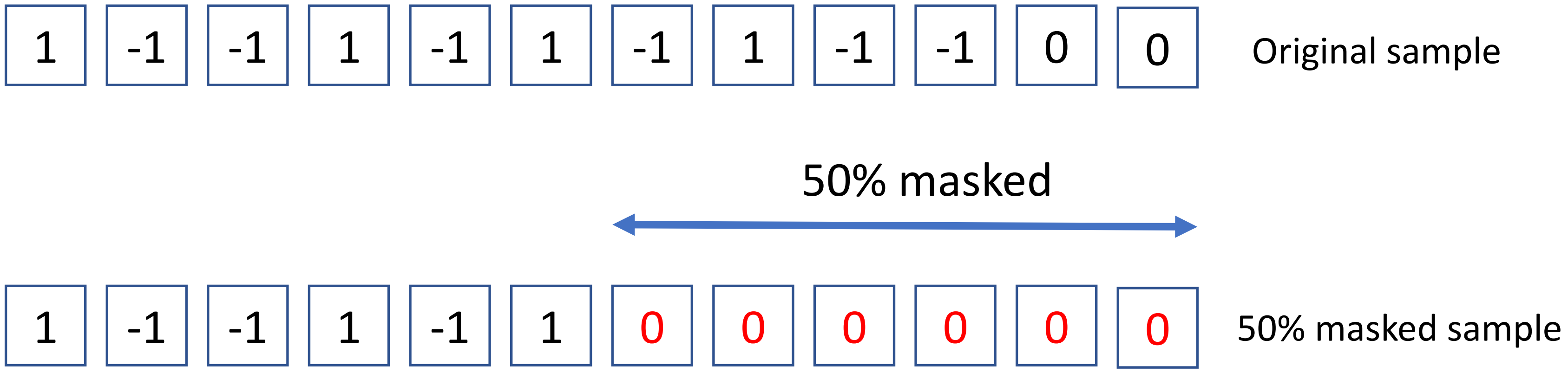


DC model

- For video streaming, high variations visible throughout the trace.

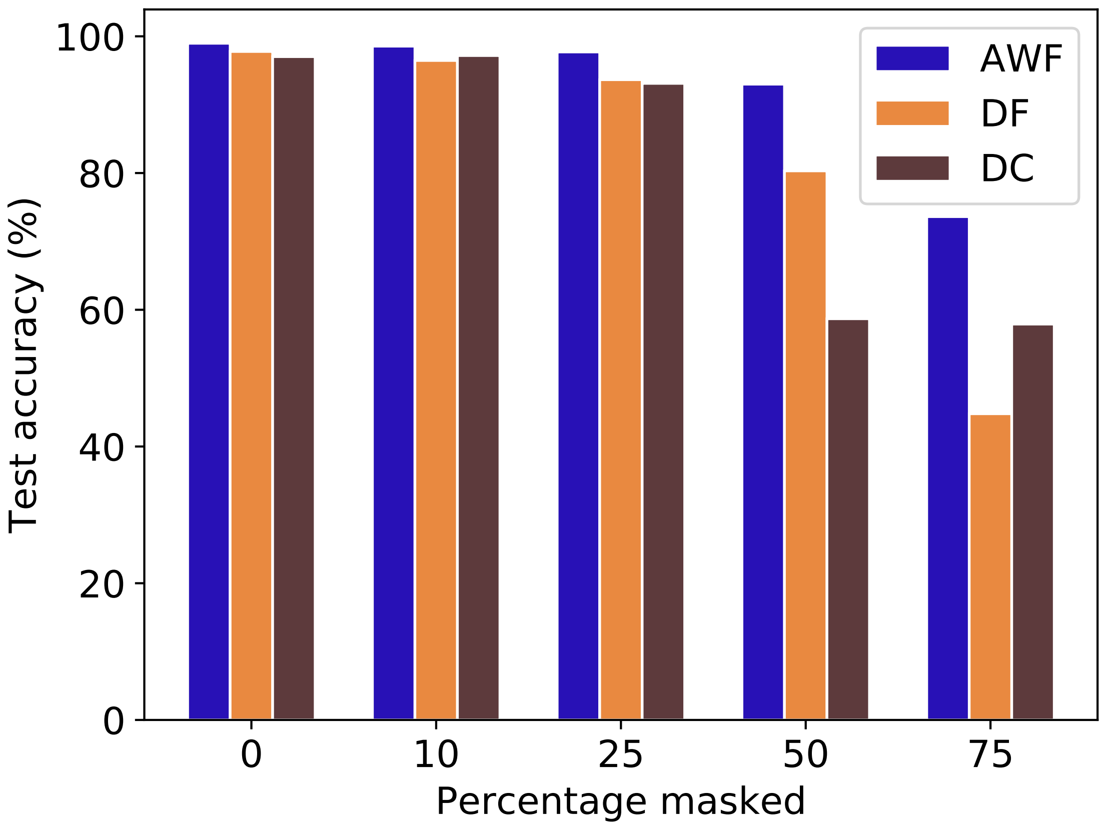
RQ 3: Would the initial trace portion alone be sufficient?

- Analyze accuracy of model on masked inputs.
- Given an original test set sample with length 10 where classifier input length is 12, process of masking is as follows.



RQ 3: Would the initial trace portion alone be sufficient?

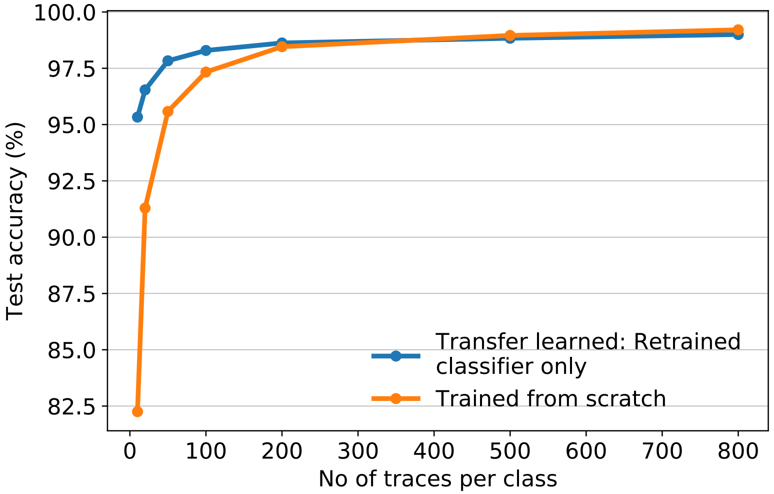
- Analyze accuracy of model on masked inputs



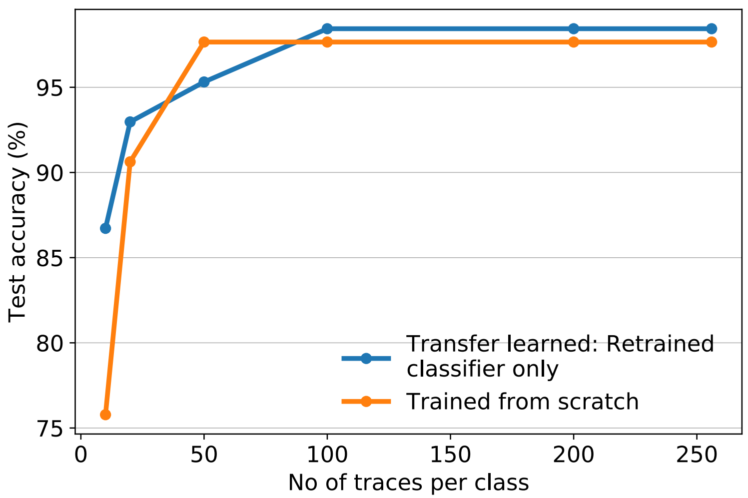
- AWF and DF models record >80% accuracy even at 50% masking.
- Website fingerprinting models focus more on initial parts → strong resilience to masking.
- DC model accuracy degrades after 25% masking.
- Video fingerprinting models focus on periodic patterns → highly susceptible to masking.

RQ 4: Can we do transfer learning?

- May be you need 1-2 points here, how you did that?



AWF model



DC model

- Transfer learned model reaches the accuracy plateau with lesser number of samples
- Less training data for fine-tuning for new classes
- Similar results were observed for DC dataset as well

RQ 5: Why CNNs outperform RNNs?

- Many work show that CNNs outperform RNNs in traffic fingerprinting
  - List some here X et al.
  - Y et al.

CNN vs RNN: Test set accuracy gap	
Dataset	Performance gap between CNN and LSTM
AWF	3.35%
DF	2%
DC	13%

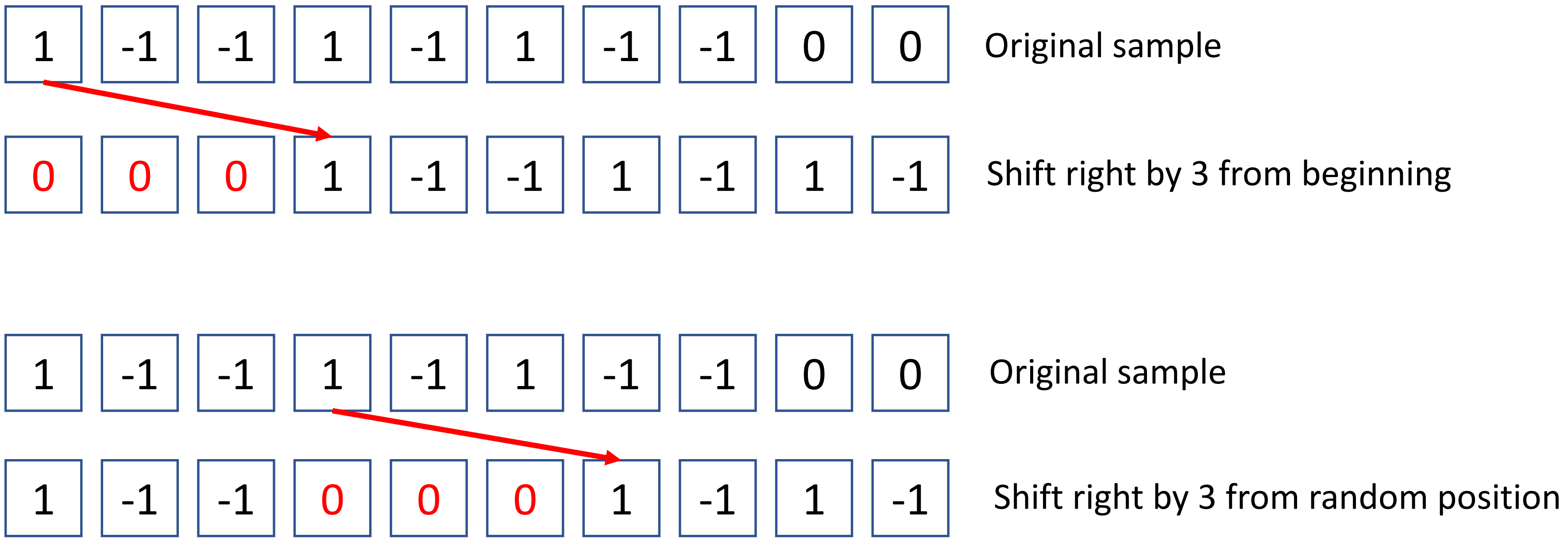
CNN vs RNN: Resilience to concept drift (AWF)					
LSTM model	Number of days between test set and train set capture				
	3	10	14	28	42
CNN	99.80%	97.90%	94.00%	89.00%	87.40%
LSTM_150	92.87%	88.91%	84.01%	77.25%	76.20%
LSTM_1500	93.50%	90.10%	84.80%	76.30%	73.20%

- CNNs were also shown high resilience to ***concept-drift.***

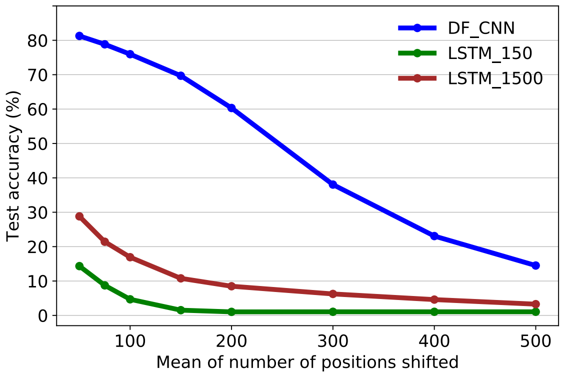
RQ 5: Why CNNs outperform RNNs?

- Evaluate the classifier performance when test set samples are shifted in multiple ways.

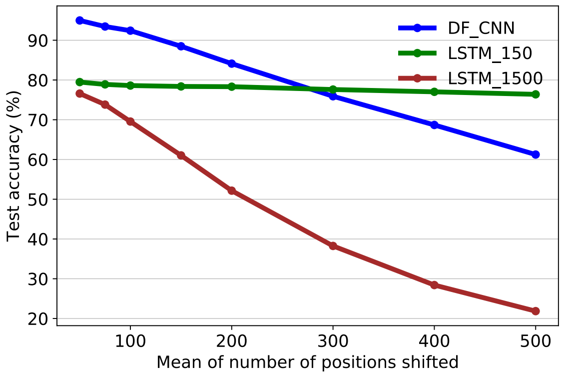
- **Example:** Given an original test set sample with length 8 where classifier input length is 10 the process of shifting right is as follows.



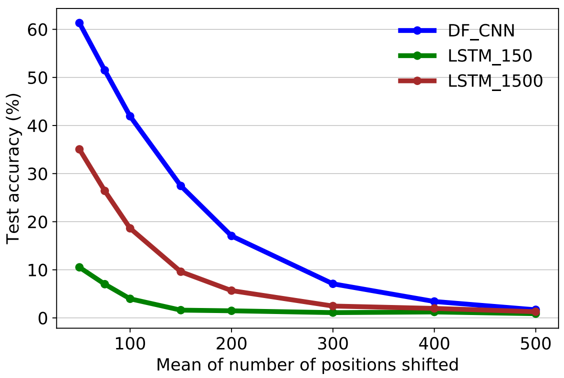
RQ 5: Why CNNs outperform RNNs?



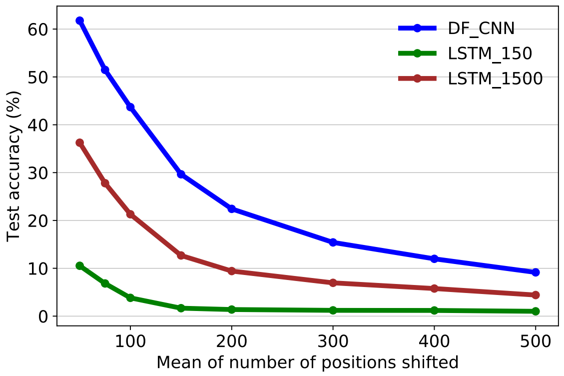
Shifted right from start



Shifted right from random



Shifted left from end



Shifted left from random

- CNNs are more resilient to shifts in burst patterns than RNNs
- Network traces contain noise due to delays and CNNs which can work better with such data perform better.

Takeaway Messages

- Website fingerprinting CNNs,
  - Give more weight to the initial part of a traffic trace which contains a high concentration of transitions
  - Can make a reasonable prediction with just the initial part of a traffic trace itself
- Video fingerprinting CNNs focus on periodic sections of uploads and downloads that correspond to periodic bursts in video streaming.
- These insights help to,
  - Design better classifiers
  - Design efficient defenses
  - More adaptive noise must be added into the parts where there are more activities

Takeaway Messages

- Traffic fingerprinting CNNs show the same transfer learning capabilities as image classifying CNNs.
  - This helps scaling up traffic fingerprinting CNNs with respect to the number of classes can be done with much less training data and time.

- Resilience of CNNs to random variations in traffic flows and bursts that occur due to varying network conditions is the main contributing factor for their success compared to RNNs.
  - Training process of traffic fingerprinting RNNs could be improved by augmenting data
  - Combination of CNN and LSTM could perform well with traffic fingerprinting