Κατηγοριοποίηση Δικτυακών Επιθέσεων με μεθόδους Μηχανικής Μάθησης

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΟΡΕΣΤΗΣ ΑΛΠΟΣ

oralpos@gmail.com
Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Κίνητρα Διπλωματικής

- Αναγνώριση Επιθέσεων
- Διάκριση της κακόβουλης κίνησης από καλόβουλη
- Εφαρμογή των Νευρωνικών Δικτύων, και κυρίως των Βαθιών Νευρωνικών Δικτύων, στον τομέα της Ασφάλειας Δικτύων.

Προηγούμενη Δουλειά

- Πολλά Papers και πολλή ενασχόληση με το θέμα.
- Το NetMODe έχει ασχοληθεί σε βάθος με Ανίχνευση και Αντιμετώπιση δικτυακών επιθέσεων.
- Οι τεχνικές που χρησιμοποιούνται στην Ανίχνευση των επιθέσεων:
  - Εντροπία
  - Τεχνικές βασισμένες στη συμμετρία της κίνησης
  - Στατιστικές Μέθοδοι (Bayes)
  - Μέθοδοι Μηχανικής Μάθησης (Clustering, SOM, SVM, Neural Networks, Auto Encoders)

Προηγούμενη Δουλειά

- Αυτό που δεν είδαμε να έχει μελετηθεί σε βάθος είναι τα Βαθιά Νευρωνικά Δίκτυα (Deep Neural Networks), δηλαδή τα Νευρωνικά Δίκτυα με ένα ή και περισσότερα κρυφά επίπεδα.

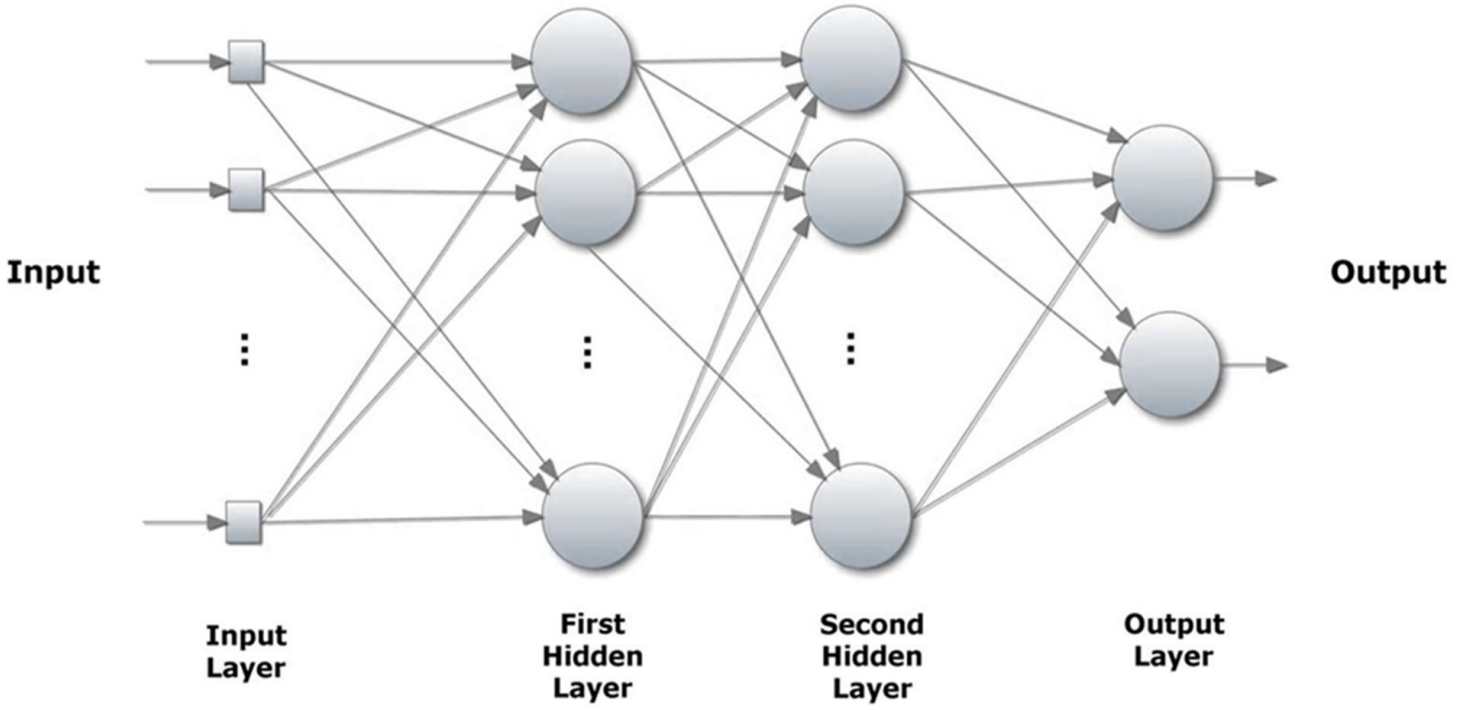
Προηγούμενη Δουλειά

- Σημαντική ενασχόληση του επιχειρηματικού κόσμου με συστήματα ανίχνευσης και αποτροπής δικτυακών επιθέσεων.
- Incapsula DDoS Protection
- Cloudflare
- Arbor Networks

Θεωρητικό Υπόβαθρο - Νευρωνικά Δίκτυα

MLP (Multi Layer Perceptron

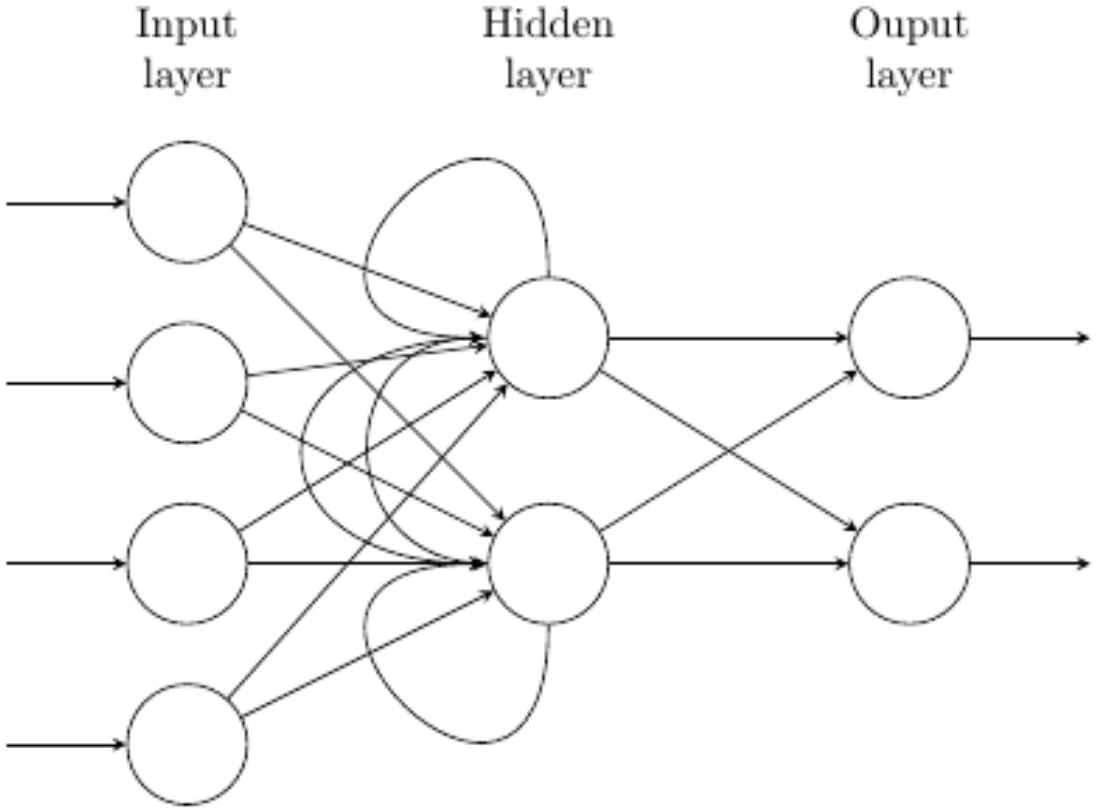
- Αποκλειστικά εμπρόσθια τροφοδότηση (feedforward), καμία ανάδραση.
- Πλήρως συνδεδεμένα επίπεδα, κάθε νευρώνας ενός επιπέδου συνδέεται με όλους τους νευρώνες του επομένου
- Μη-γραμμική συνάρτηση ενεργοποίησης (activation function).



Θεωρητικό Υπόβαθρο - Νευρωνικά Δίκτυα

RNN (Recurrent Neural Networks)

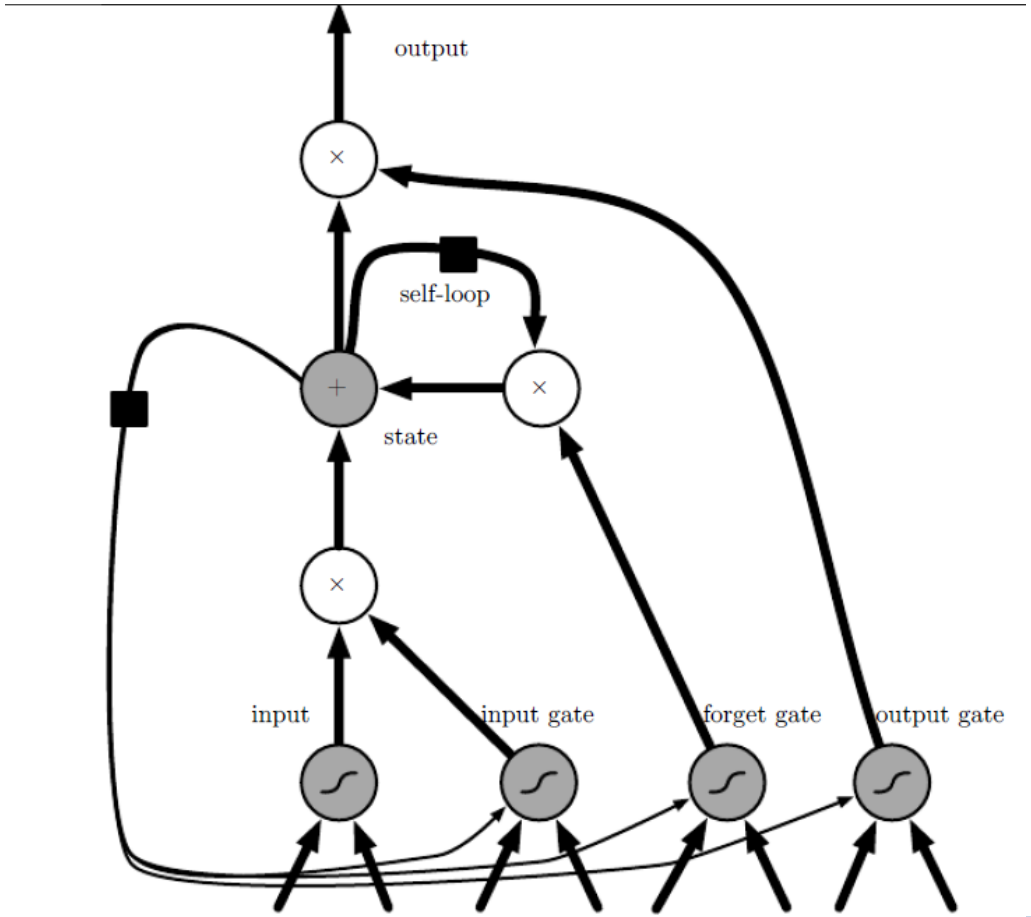
- Ικανά να επεξεργαστούν ακολουθίες εισόδων.
- Σημαντικά πλεονεκτήματα για δεδομένα που έχουν κάποια χρονική εξέλιξη (time series).
- Προσθέτουν στο σύστημα τη δυνατότητα να θυμάται προηγούμενες εισόδους.



Θεωρητικό Υπόβαθρο - Νευρωνικά Δίκτυα

LSTM (Long Short-Term Memory)

- Η έκφραση των Long-Term Dependencies σε ένα νευρωνικό δίκτυο αποτελεί ένα ζήτημα.
- Η πιο αποτελεσματική λύση αυτή τη στιγμή στην πράξη δίνουν τα gated RNNs, κατηγορία στην οποία ανήκουν και τα LSTM.
- Εξωτερικά παραμένει ίδιο με το κλασικό κύτταρο, εσωτερικά χρησιμοποιεί πύλες, με βάρη που μαθαίνει, για ελέγξει την ανάδραση.

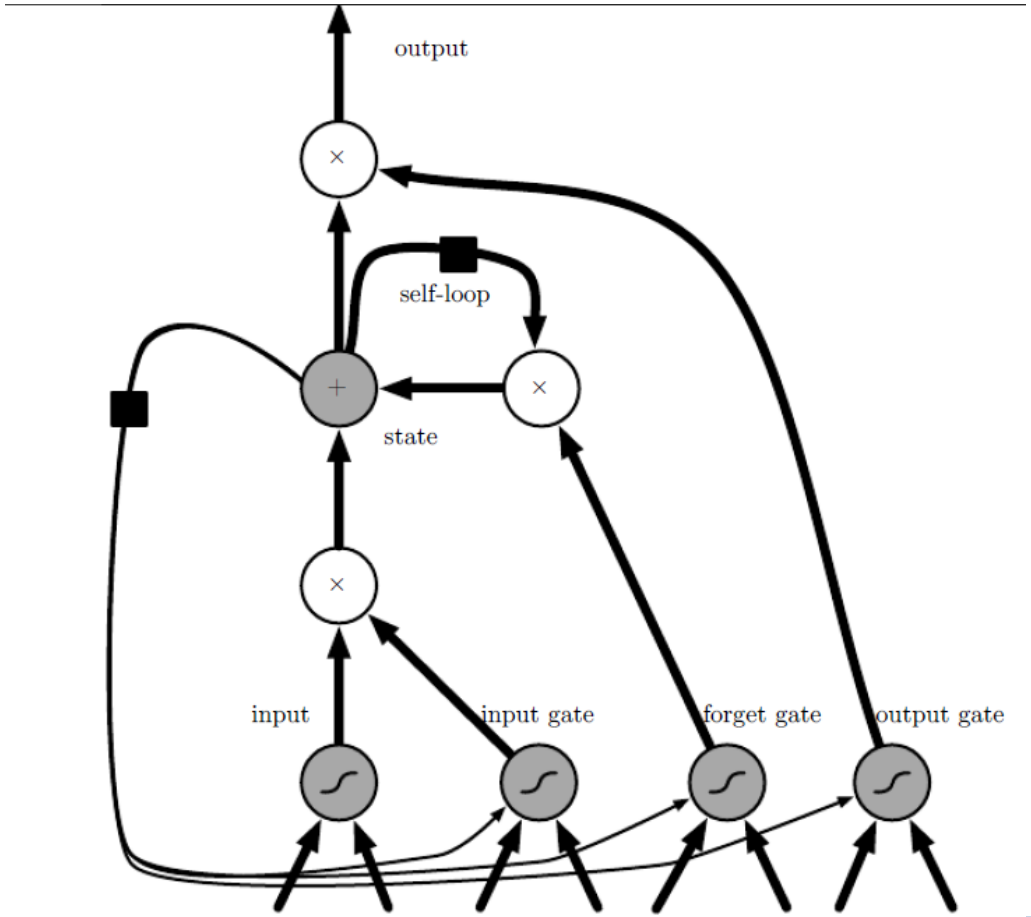


Θεωρητικό Υπόβαθρο – Νευρωνικά Δίκτυα

LSTM (Long Short-Term Memory)

Τρεις πύλες

- Η *input gate* καθορίζει το συντελεστή με τον οποίο η εξωτερική είσοδος του LSTM κυττάρου θα πολλαπλασιαστεί.
- Η *forget gate* καθορίζει το συντελεστή με τον οποίο η εσωτερική ανάδραση του LSTM κυττάρου θα πολλαπλασιαστεί για να καθορίσει την εσωτερική του κατάσταση.
- Η *output gate*, καθορίζει το συντελεστή με τον οποίο η εσωτερική κατάσταση του κυττάρου θα περάσει στην έξοδο (μπορεί άρα να απενεργοποιήσει συνολικά το LSTM cell)



Θεωρητικό Υπόβαθρο - Νευρωνικά Δίκτυα

LSTM (Long Short-Term Memory)

Τρεις πύλες

- Η *input gate* καθορίζει το συντελεστή με τον οποίο η εξωτερική είσοδος του LSTM κυττάρου θα πολλαπλασιαστεί.
- Η *forget gate* καθορίζει το συντελεστή με τον οποίο η εσωτερική ανάδραση του LSTM κυττάρου θα πολλαπλασιαστεί για να καθορίσει την εσωτερική του κατάσταση.
- Η *output gate*, καθορίζει το συντελεστή με τον οποίο η εσωτερική κατάσταση του κυττάρου θα περάσει στην έξοδο (μπορεί άρα να απενεργοποιήσει συνολικά το LSTM cell)

$$g_i^{(t)} = \sigma\left(b_i^g + \sum_j U_{i,j}^g x_j^{(t)} + \sum_j W_{i,j}^g h_j^{(t-1)}\right).$$

$$f_i^{(t)} = \sigma\left(b_i^f + \sum_j U_{i,j}^f x_j^{(t)} + \sum_j W_{i,j}^f h_j^{(t-1)}\right)$$

$$s_i^{(t)} = f_i^{(t)} s_i^{(t-1)} + g_i^{(t)} \sigma\left(b_i + \sum_j U_{i,j} x_j^{(t)} + \sum_j W_{i,j} h_j^{(t-1)}\right)$$

$$q_i^{(t)} = \sigma\left(b_i^o + \sum_j U_{i,j}^o x_j^{(t)} + \sum_j W_{i,j}^o h_j^{(t-1)}\right)$$

$$h_i^{(t)} = \tanh\left(s_i^{(t)}\right) q_i^{(t)}$$

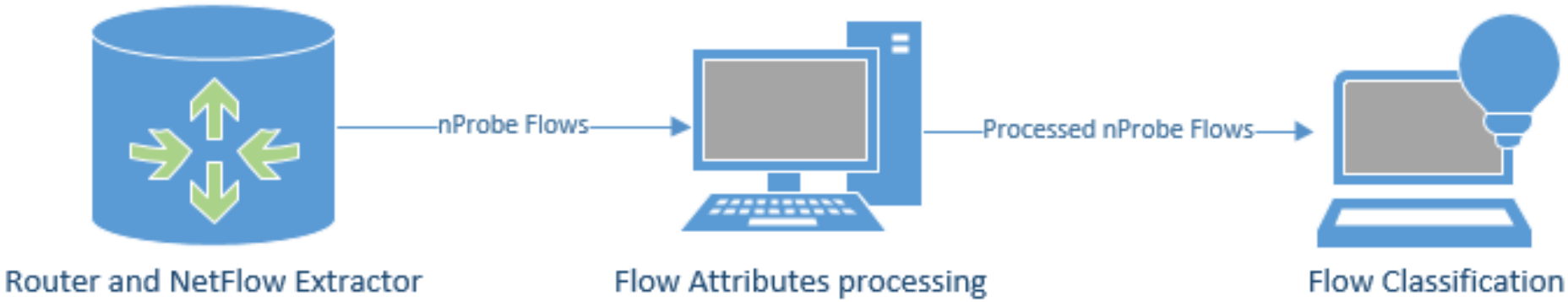
Τα δεδομένα που χρησιμοποιήθηκαν

Είδος Κίνησης	Πηγή Κίνησης	Πλήθος Ροών (x1000)
Benign	Πραγματική κίνηση από Switch του εργαστηρίου	450
Ping Attack	CAIDA	400
TCP SYN Attack	CAIDA	400
UDP Attack	Τεχνητή, παραγωγή με Scapy	400
Port Scanning	Τεχνητή, παραγωγή με nmap	90

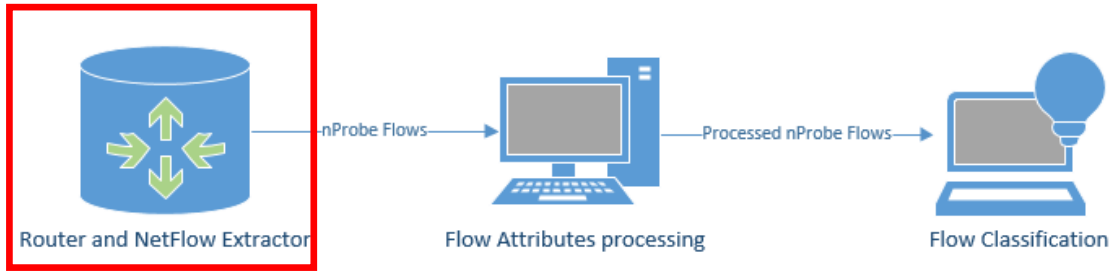
Τα δεδομένα που χρησιμοποιήθηκαν

DataSet	Είδη Κίνησης που περιέχει
DataSet 0	Όλα τα είδη κίνησης
DataSet 1	Καλόβουλη και Ping Attack
DataSet 2	Καλόβουλη και TCP SYN Flood
DataSet 3	Καλόβουλη και UDP Flood
DataSet 4	Καλόβουλη και Port Scanning
DataSet 5	Καλόβουλη, TCP SYN Flood και Port Scanning

Αρχιτεκτονική του Συστήματος



Εξαγωγέας NetFlow

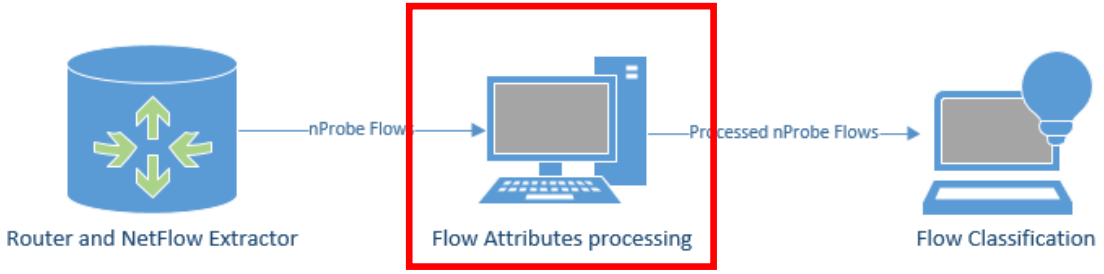


Χρησιμοποιήθηκε το nProbe, που είναι υλοποίηση του NetFlow v9 από τη Cisco.

Δημιουργεί Ροές με βάση 5 χαρακτηριστικά της κίνησης, IP προέλευσης και προορισμού, Θύρα προέλευσης και προορισμού και Πρωτόκολλο.

OUT_PKTS	PORT_TARGET	RETRANSMITTED_OUT_PKTS
IN_PKTS	TCP_FLAGS	NUM_PKTS_UP_TO_128B
PROTOCOL	MIN_TTL	FLOW_START_MILLISEC
IPV4_SOURCE	MAX_TTL	FLOW_END_MILLISEC
IPV4_TARGET	ICMP_TYPE	SRC_AS
PORT_SOURCE	RETRANSMITTED_IN_PKTS	DST_AS

Επεξεργασία των Χαρακτηριστικών των Ροών



Τα Χαρακτηριστικά της κίνησης που εξήχθησαν από το nProbe τροποποιήθηκαν, ώστε να μπορούν να δοθούν ως είσοδος στα Νευρωνικά.

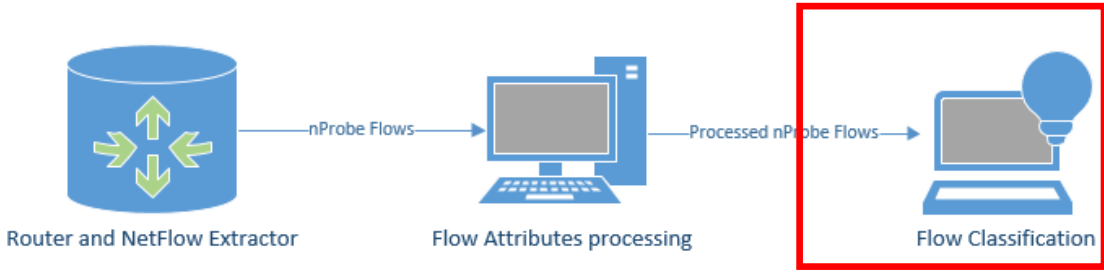
Για παράδειγμα:

Ανάλυση του πεδίου TCP_FLAGS στα πεδία ACK, PSH, RST, SYN, FIN, καθένα από τα οποία έπαιρνε την τιμή 0 ή 1, ανάλογα με το αντίστοιχο bit στο αρχικό πεδίο TCP_FLAGS.

Αντίστοιχα για τα πεδία PROTOCOL, PORT_SOURCE, PORT_TARGET, ICMP_TYPE, SRC_AS, DST_AS.

Κατηγοριοποίηση των Ροών

- Χρήση μόνο νευρωνικών δικτύων
- Υλοποίηση με τη βιβλιοθήκη Keras σε Python



Είδος Δικτύου	Συνδυασμοί Παραμέτρων που δοκιμάστηκαν
MLP	• Πλήθος κρυφών επιπέδων • Πλήθος Νευρώνων ανά επίπεδο • Ποσοστό Dropout • Συνάρτηση Κόστους (MSE, Crossentropy) • Μέθοδος Εκπαίδευσης (SGD, RMSProp, AdaGrad)
RNN	• Πλήθος κρυφών επιπέδων • Συνάρτηση Κόστους (MSE, Crossentropy) • Μέθοδος Εκπαίδευσης (SGD, RMSProp, AdaGrad)
LSTM	• Πλήθος κρυφών επιπέδων • Συνάρτηση Κόστους (MSE, Crossentropy) • Μέθοδος Εκπαίδευσης (SGD, RMSProp, AdaGrad)

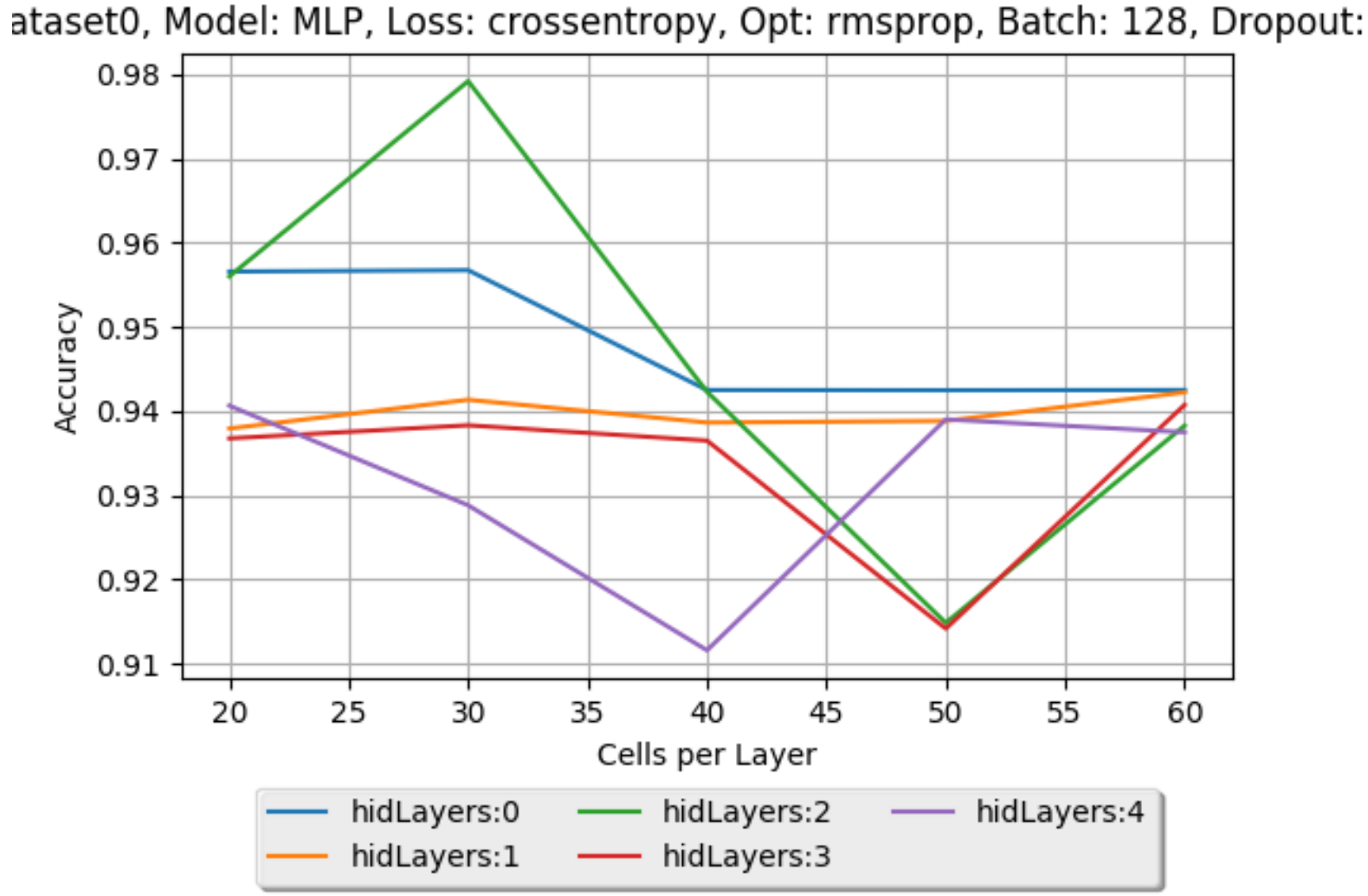
Συμπεράσματα - Βέλτιστα Δίκτυα

Είδος δικτύου	Κρυφά επίπεδα	Νευρώνες ανά επίπεδο	Activation Function	Dropout Rate	Cost Function	Optimizer	Ακρίβεια στο DS0
MLP	2	30	ReLU	0.4	Crossentropy	RMSProp	98 %
RNN	2	50	ReLU	0	Crossentropy	RMSProp	99 %
LSTM	2	50	ReLU	0	Crossentropy	RMSProp	97.7%

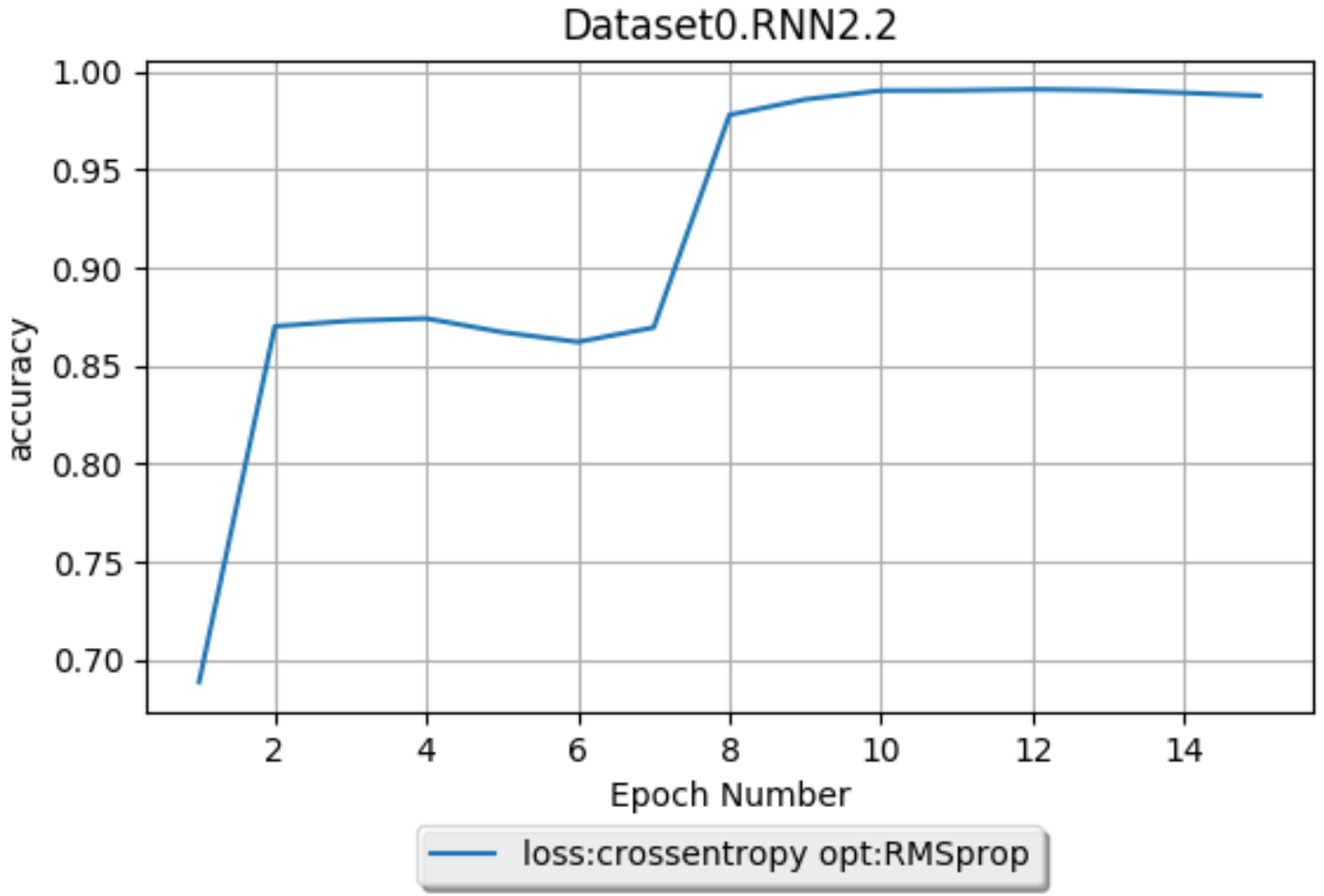
Συμπεράσματα – Βέλτιστα Δίκτυα

- Όλα τα δίκτυα που κρίθηκαν ως βέλτιστα είχαν 2 κρυφά επίπεδα. Ένα δίκτυο με 2 κρυφά επίπεδα μπορεί (για τα συγκεκριμένα δεδομένα τουλάχιστον) να μάθει πολύ καλά τα δεδομένα εκπαίδευσης, άρα δε φαίνεται να υπάρχει λόγος να χρησιμοποιηθούν πολύ βαθιές αρχιτεκτονικές (με 4 και πάνω επίπεδα).
- Η μετάβαση από MLP σε RNN είχε ως συνέπεια αύξηση στο χρόνο εκπαίδευσης αλλά και βελτίωση της ακρίβειας πρόβλεψης. Αντίθετα, η μετάβαση σε LSTM δεν είχε ως συνέπεια αντίστοιχη αύξηση στην ακρίβεια των προβλέψεων.
- Ο χρόνος που απαιτείται για την εκπαίδευση και την κατάταξη μιας Ροής, καθώς και οι υπολογιστικοί πόροι ήταν σε κάθε περίπτωση ικανοποιητικοί, με εξαίρεση τα LSTM.

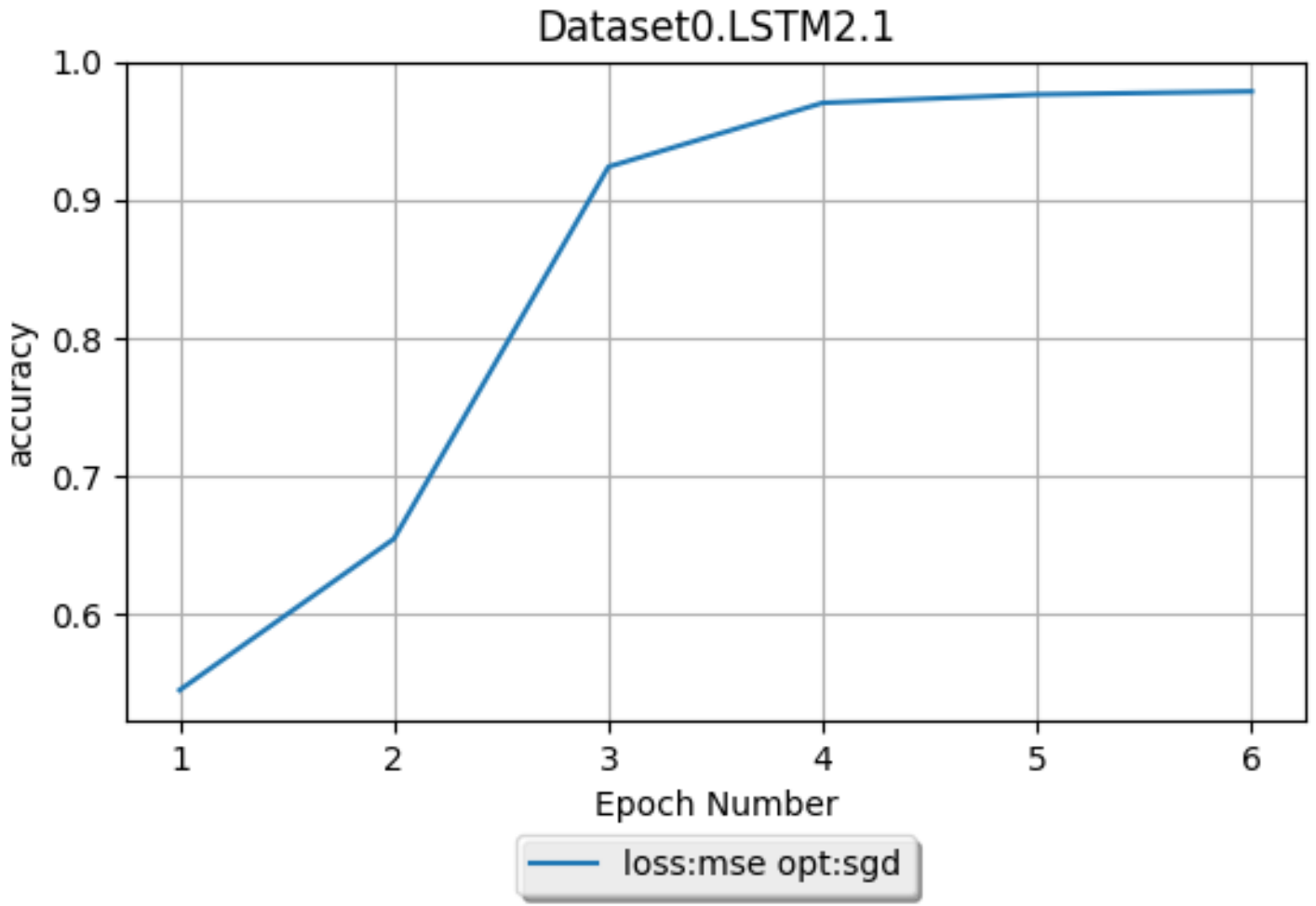
Συμπεράσματα - Βέλτιστα Δίκτυα



Συμπεράσματα - Βέλτιστα Δίκτυα



Συμπεράσματα - Βέλτιστα Δίκτυα



Μελλοντική δουλειά

1. Προσθήκη ενός επιπλέον χαρακτηριστικού (Attribute) στην είσοδο των Νευρωνικών, σχετικό με τις IP προέλευσης. Χρήση αλγορίθμου για IP Clustering, με βάση ένα πρόθεμα. Το νέο χαρακτηριστικό της ροής θα φανερώνει πόσες ακόμα ροές – εκτός από την τρέχουσα – έχουν την ίδια IP με την τρέχουσα ως διεύθυνση προέλευσης.
2. Πώς μπορεί να γίνει επανεκπαίδευση του συστήματος από τη στιγμή που θα τεθεί σε λειτουργία. Με βάση, δηλαδή, την πραγματική κίνηση, σε ποιο χρονικό σημείο θα μπορούσαμε να επαναεκπαιδεύσουμε το δίκτυο ώστε να προσαρμοστεί στα πραγματικά δεδομένα της κίνησης.

Ερωτήσεις - Απορίες


thank you!