MSc. SOFTWARE ENGINEERING and ARTIFICIAL INTELLIGENCE

Proactive network load distribution and link utilization optimization using machine-learning techniques

Fakhraddin Mohammed JAF

Supervisors:

Ezequiel López Rubio and *Juan Carlos Burguillo Rial*

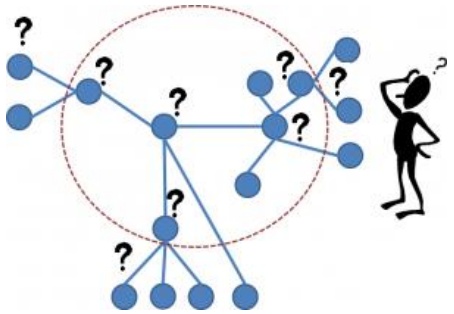
Department of Computer Languages and Computer Science

Universidad de Málaga

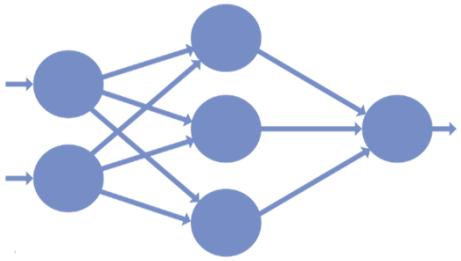
18 July 2017

Introduction

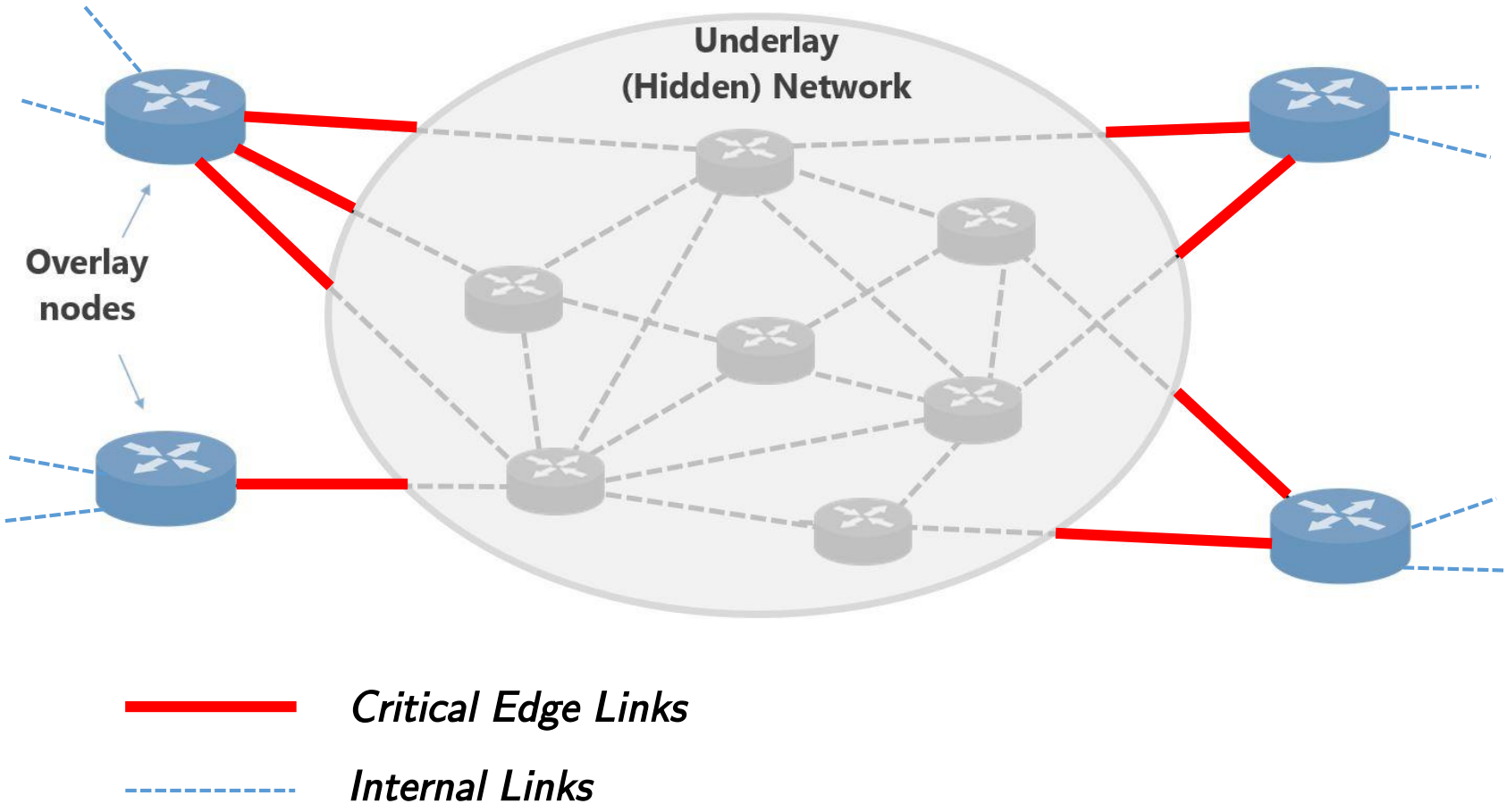
- Challenges
  - The growth of complexity in networks
  - Bandwidth demands vary over time
  - Passive measurements are not end-to-end accurate



- Potential solution
  - Machine learning can detect hidden patterns and understand historical changes



Overlay / Underlay



Previous Works

1. Analytical models on performance measurements

Investigations on how performance metrics such as delay, loss, and throughput vary over time for given Internet paths.

Ott et al [1996]

Mathis et al [1997]

Paxson [1997],

Barford & Crovella [2000]

Zhang et al [2002] and [2001]

Ott [2005],

...

Previous Works

2. History based time-series prediction

Used time-series analysis of historical flow statistics on a given path, to predict future metrics, using machine-learning forecasting approaches.

Zhang et al [2001]

Swany et al. [2002]

Lu et al. [2005]

Mirza, et al. [2010]

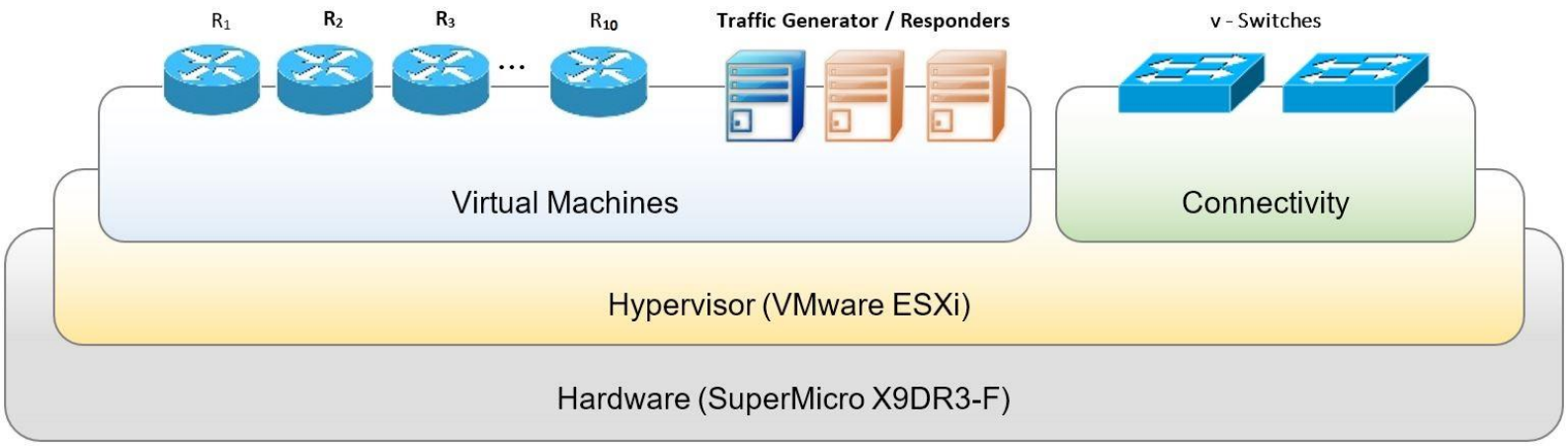
...

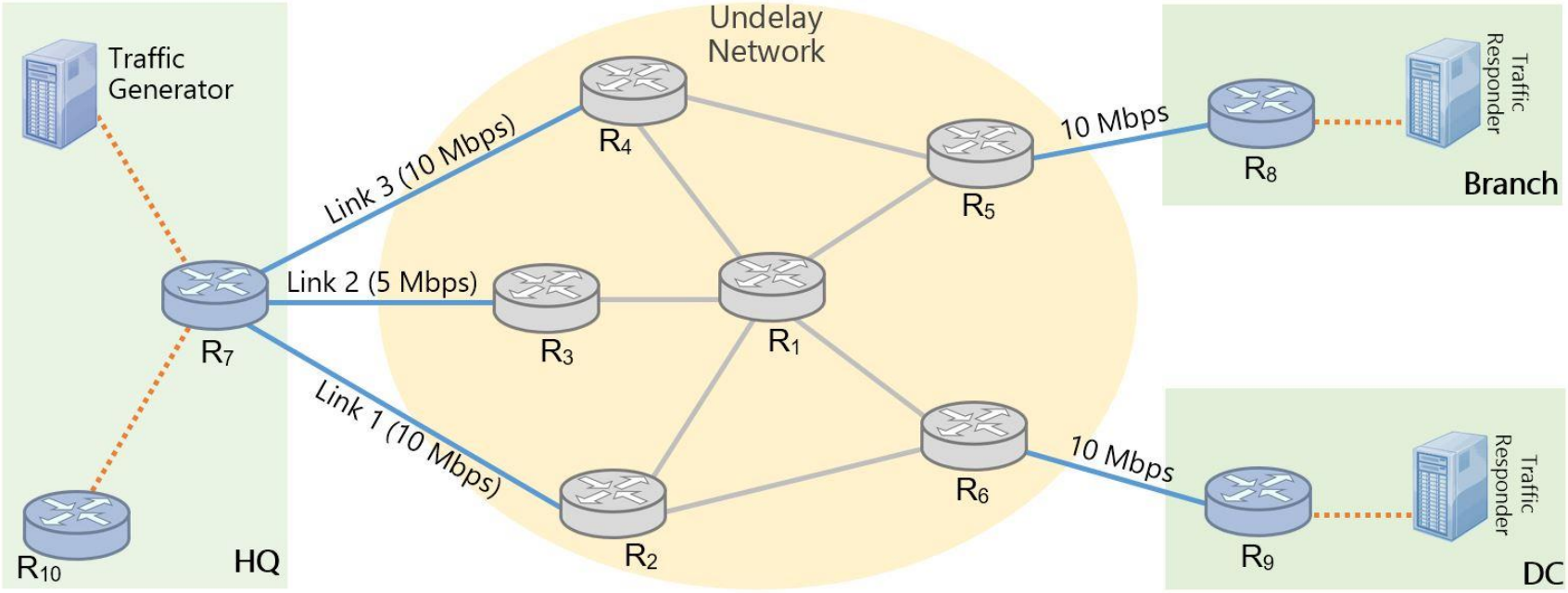
Aims of the work

- Study of dynamic and proactive methods for flow distribution and utilization optimization on critical links, instead of using traditional passive measurements

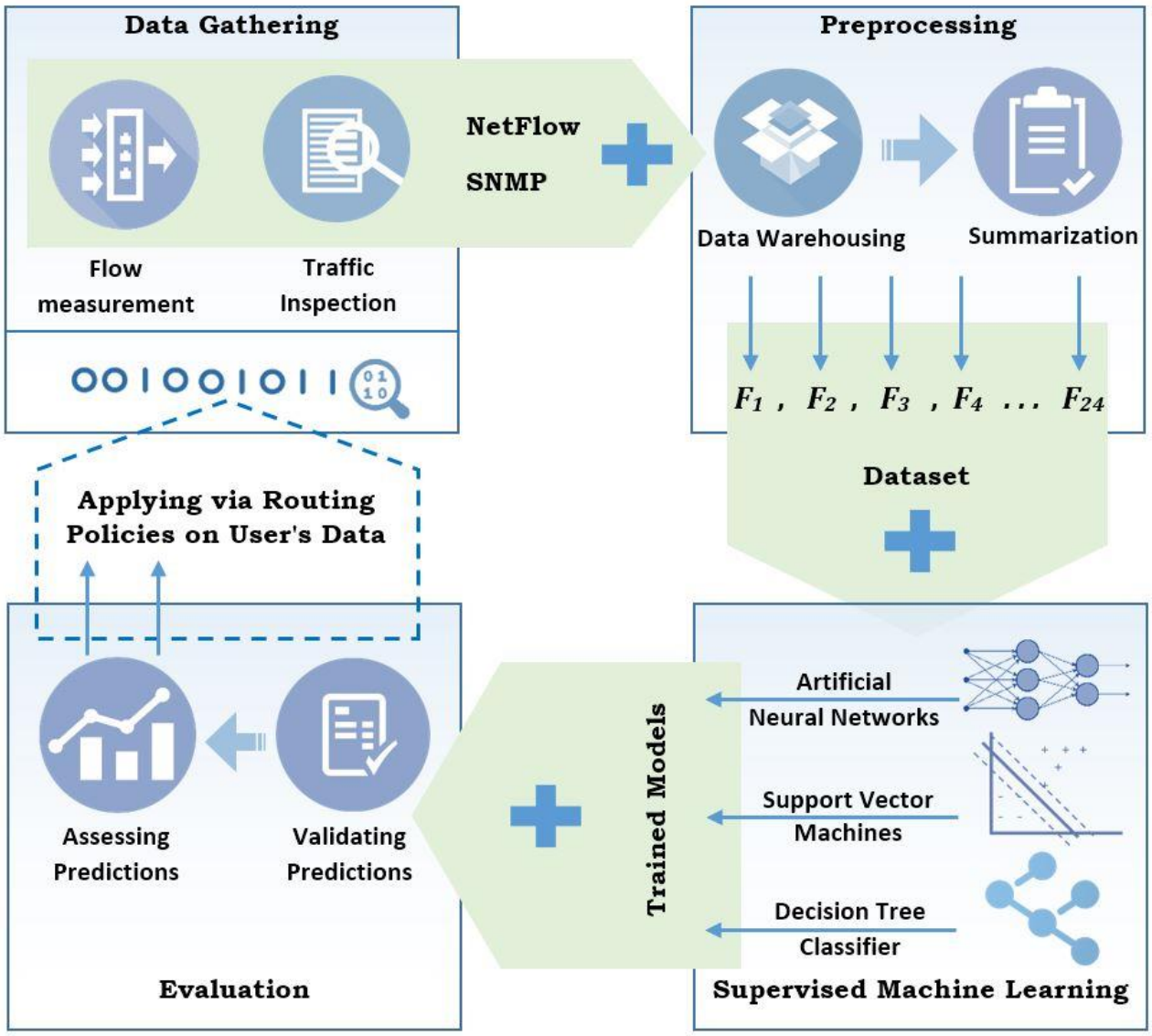
- Employing dynamic models learned by machine learning techniques for per-flow routing policy deployment (decision making)

Implementation

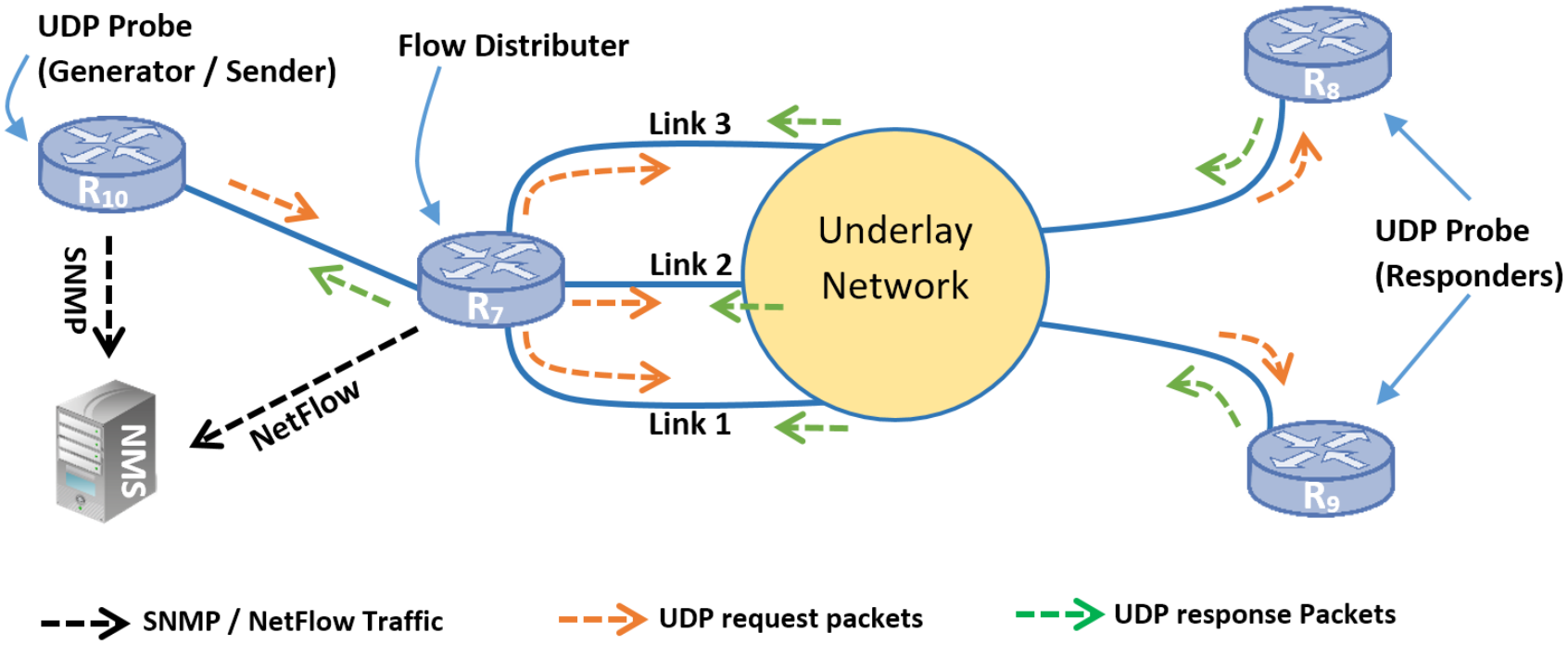




Methodology



Data Gathering – via 24 UDP flow operations



Parameter (per each UDP flow)	*Value*
Number of packets (N)	*10 packets*
Payload size per request packet (S)	*1480 bytes*
Payload size per response packet (P)	*1480 bytes*
Time between packets, in milliseconds (T)	*10 msec*
Elapsed time before the operation repeats, in seconds (F)	*2 seconds*
Timeout of each flow	*2000 msec*

Preprocessing

A sample of raw dataset

Flow id	Destination	Outgoing interface	Avg RTT	Avg Jitter	Packets out of Seq	Avg Latency Src-Dest	Avg Latency Dest-Src	Packets Lost(%)
10830	Branch	Link 1	130.68	19.08	0.24	153.6	11.96	26.7636
10950	Datacenter	Link 1	4.3000	1.2000	0.0	2.1000	0.4000	7.5000
…	…	…	…	…	…	…	…	…

- ***TCP throughput*** *(based on Mathis et.al):* $\longrightarrow \dfrac{\text{maximum segment size}}{RTT * \sqrt{packet\ loss}}$
- ***MOS Score*** *(based on ITU-T P.80021 and P.91022):*

Latency + Jitter → *Effective latency* → *R value* → *MOS Score*

A sample of processed dataset

Time stamp	Flow id	Destination	Outgoing interface	TCP throughput	Effective latency	R value	MOS score
42888.72222	10830	Branch	Link 1	55001.71	201.7600	18.11500	1.1991151
42888.72292	10950	Datacenter	Link 1	3157610.28	14.5000	74.08750	3.7823780
…	…	…	…	…	…	…	…

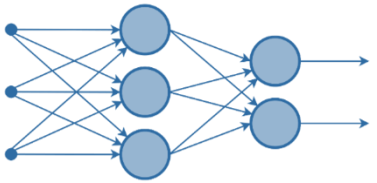
ML Training and Validation Processes

- *Three different techniques and ML algorithms*
- *Different structures per algorithm (spot-check)*
- *2/3 – 1/3 training-validation split*
- *Repeated holdout method*
- *Accuracy Calculation:*

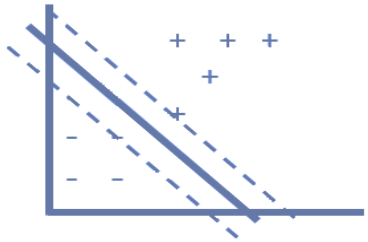
  Correctly predicted (x) divided by total number of testing instances (y).

Supervised ML algorithms

Artificial Neural Network (ANN) - nnet package (R)
four configuration sets, each with three different values for "neuron-size" and "max-itr" arguments
(= 9 structure sets per each configuration set)

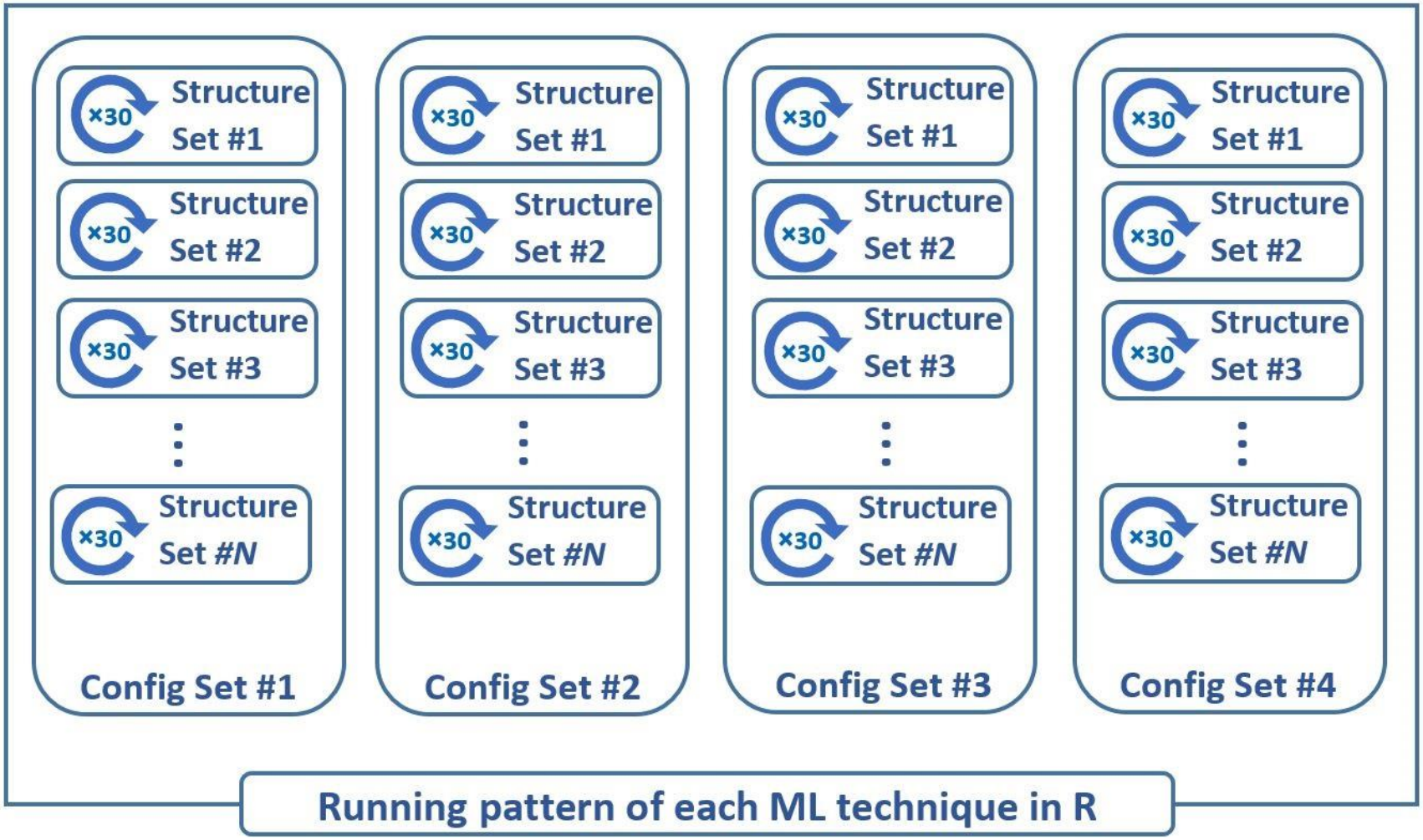


Support Vector Machine (SVM) - e1071 package (R)
four configuration sets, each with three different values for "Cost" and "Gamma" arguments
(= 9 structure sets per each configuration set)

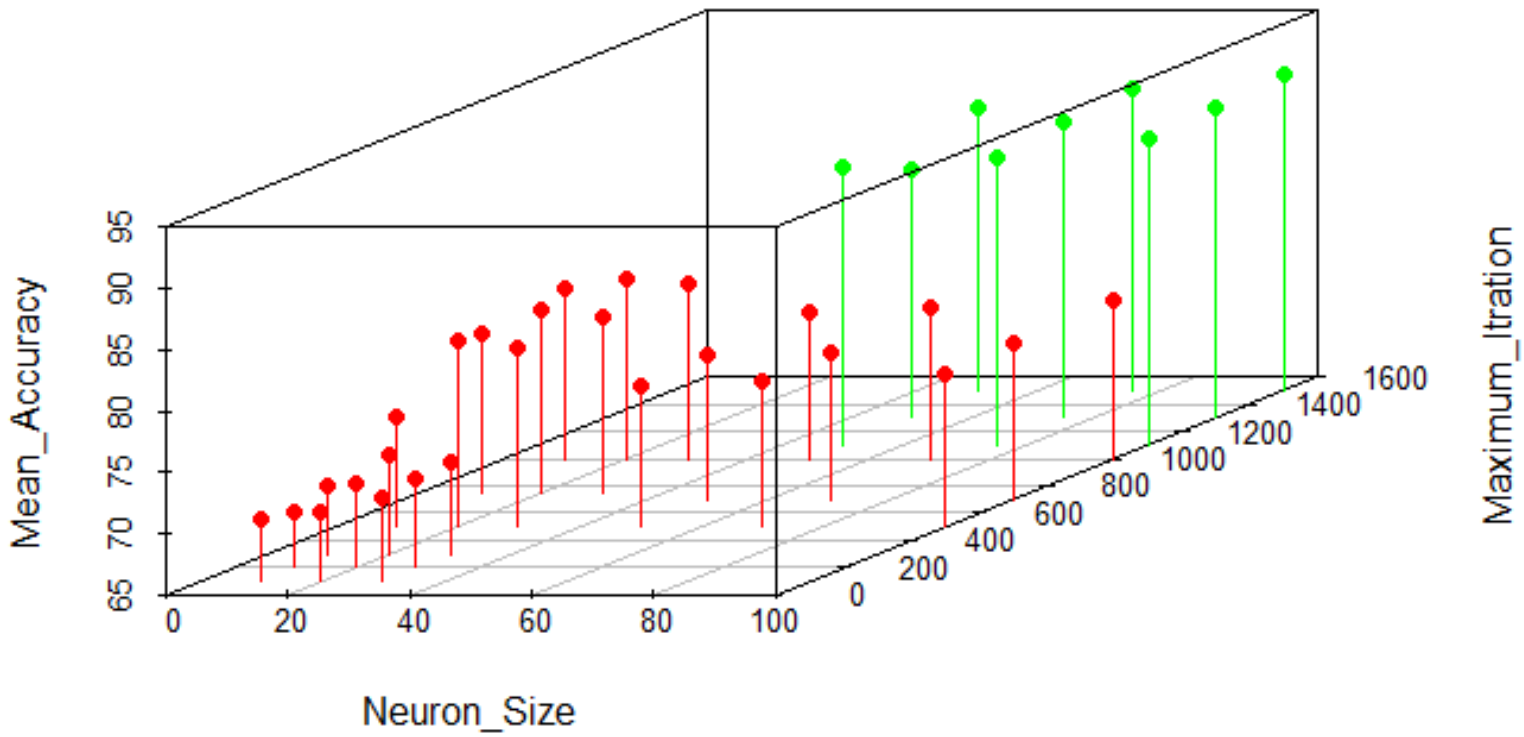


Decision Tress (DT) - rpart package (R)
four configuration sets, each with four different values for the argument called "complexity parameter (cp)"
(= 4 structure sets per each configuration set)

Supervised ML algorithms



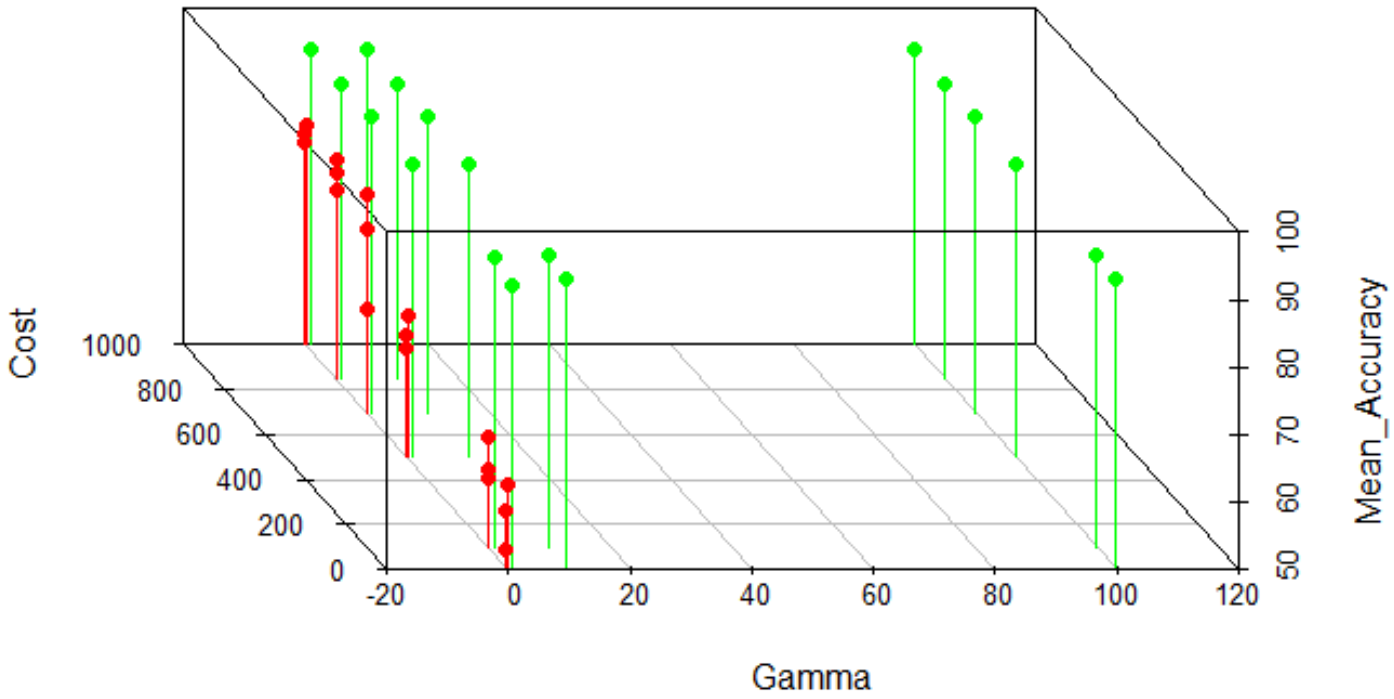
ANN Accuracy result



- *Higher accuracies in structure sets with **neuron size** of 50 or higher, and **maximum iteration** of 1000 and higher*

- *Green items are structure sets with ACC more than 85 %*

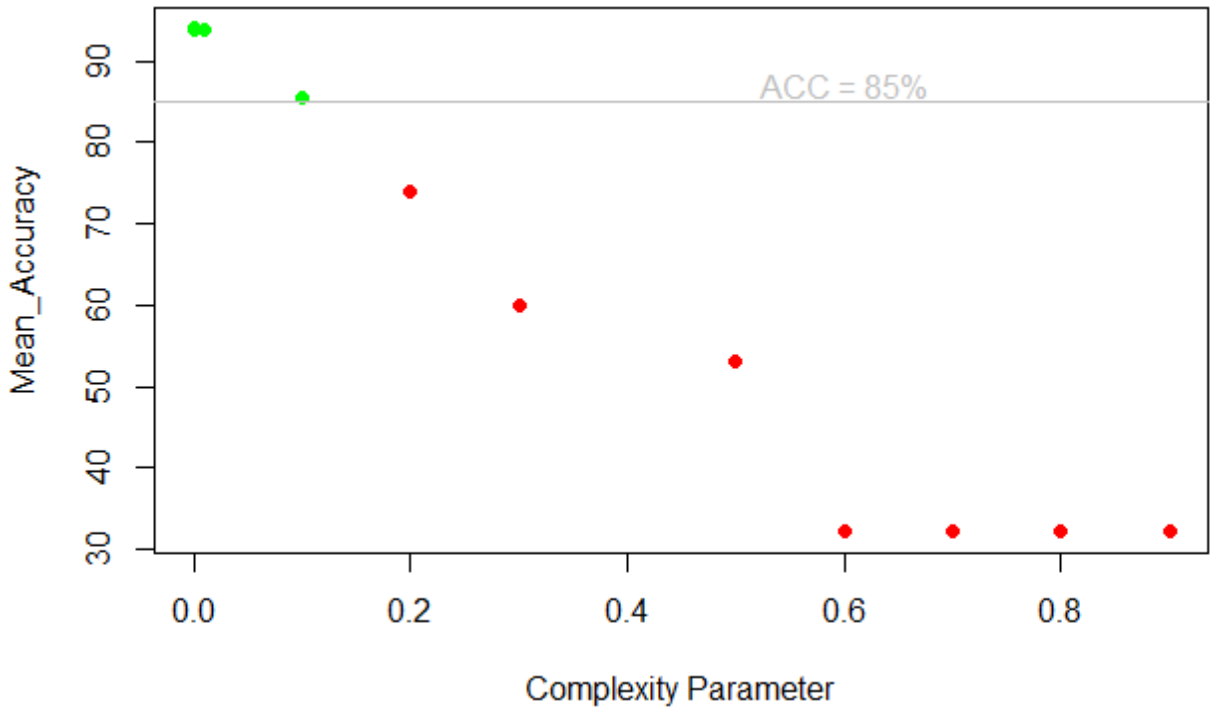
SVM Accuracy result



- *Higher accuracies in structure sets with **Gamma** value of 1 or higher. The **Cost** argument had less impact on accuracy*

- *Green items are structure sets with ACC more than 85 %*

DT Accuracy result



- *Higher accuracies in structure sets with* ***Complexity Parameter*** *of 0.1 or lower.*

- *Green items are structure sets with ACC more than 85 %*

Accuracy results comparison

	Number of Structure sets	Repetition per each set	Mean ACC (Best)	Max ACC (Best)	Avg. Runtime
ANN	36	30	90.79183	98.7684	1.394665 hours
SVM	36	30	93.96141	95.68966	6.437531 minutes
DT	16	30	94.04762	95.93596	6.809009 seconds

Per flow predication for user traffic

Flow ID	Destination	Application / Service	Protocol	Required MOS Score	Required TCP Throughput
18	172.16.18.8 (Branch)	VoIP	RTP / SIP / H232	3.8	1920000
28	172.16.28.8 (Branch)	Biz - Data	HTTPS/POP3/SMTP	3.0	6000000
19	172.16.19.9 (Datacenter)	Video Surv.	RTP / RTCP/ RTSP	4.0	4000000
29	172.16.29.9 (Datacenter)	Backup	FTP / WebDAV	3.0	4000000
30	1.1.1.130 (External)	Internet call	Skype/UDP/TCP	3.7	2000000
31	1.1.1.131 (External)	Web Surfing	P2P/HTTP/HTTPS	3.0	2500000

```
> predict(ANN_Model, traffic_flows[1,], type="raw")
    link_1         link_2          link_3
1   0.003980481    4.14895e-06     0.9960154
```

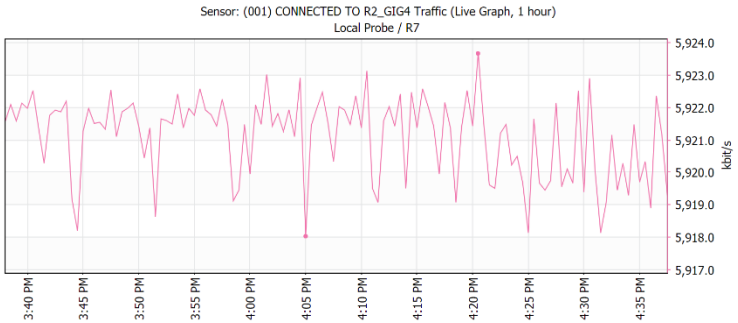
```
> predict(SVM_Model, traffic_flows[1,], probability = TRUE)
    link_1         link_2         link_3
1   0.03189795     0.05071545     0.9173866
```

```
> predict(DT_Model, traffic_flows[1,], type="prob")
    link_1         link_2         link_3
1    0              0              1
```

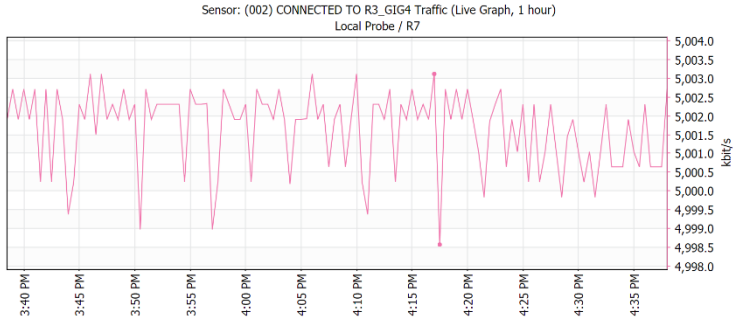
Link utilization status

Before ML

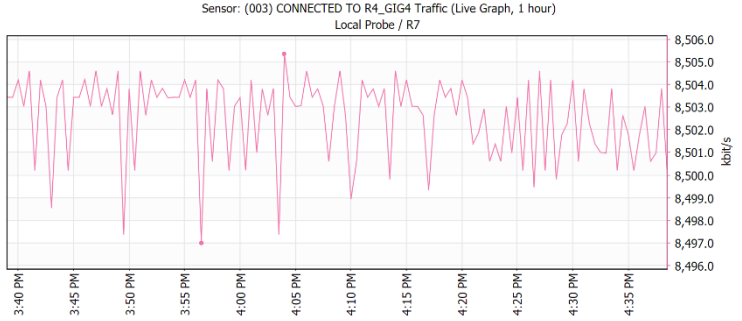
Link 1



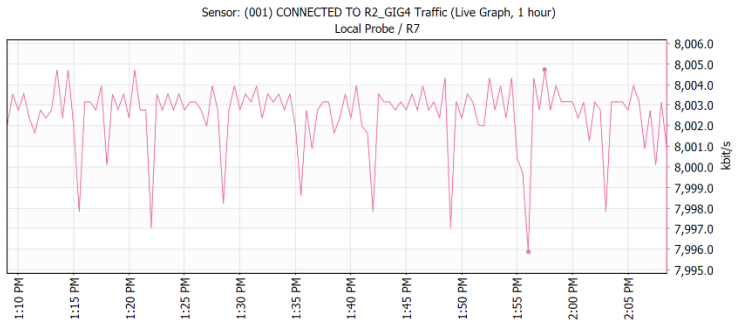
Link 2

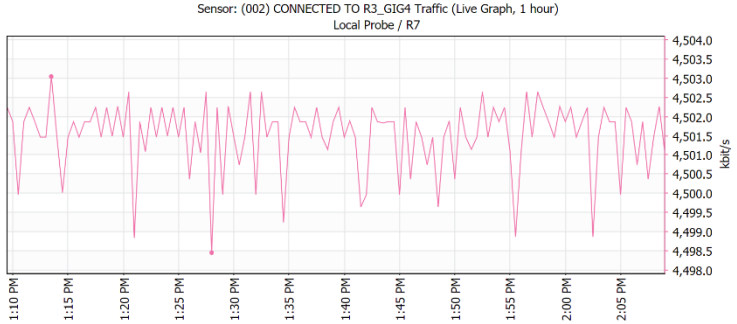


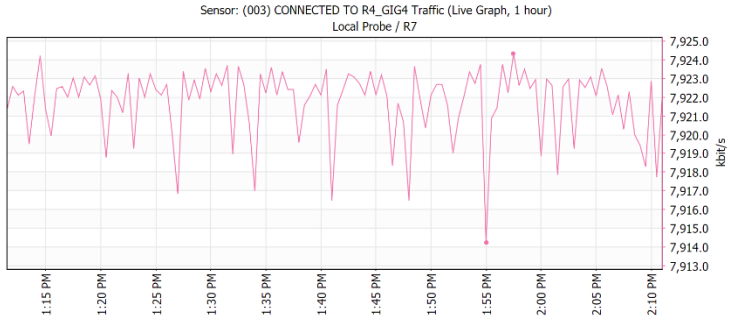
Link 3



After ML







Conclusions

- *ML models could provide more accurate patterns and understandings of underlay paths compare to passive measurements*
- *Using historical flow data, ML helped to build large scale outlook instead of having only partial views acquired by traditional methods*
- *By the use of proactive measurement / prediction framework, routing policies made by ML models*

Future directions

- *An advice for future works can be studies on larger environments where estimation errors could be perfectly considered due to higher possible range of changes in the network.*

- *Another potential challenge for similar works could be applying proposed methodology on heterogeneous network environments in order to include all possible path metrics such as MTU, in measurements.*

Thank You

And special thanks to GTI group / University of Vigo