IP Geolocation from DNS and BGP Data with Deep Learning

Jason Wei
Dartmouth College
jason.20@dartmouth.edu

Alin Popescu
Oracle
alin.popescu@oracle.com

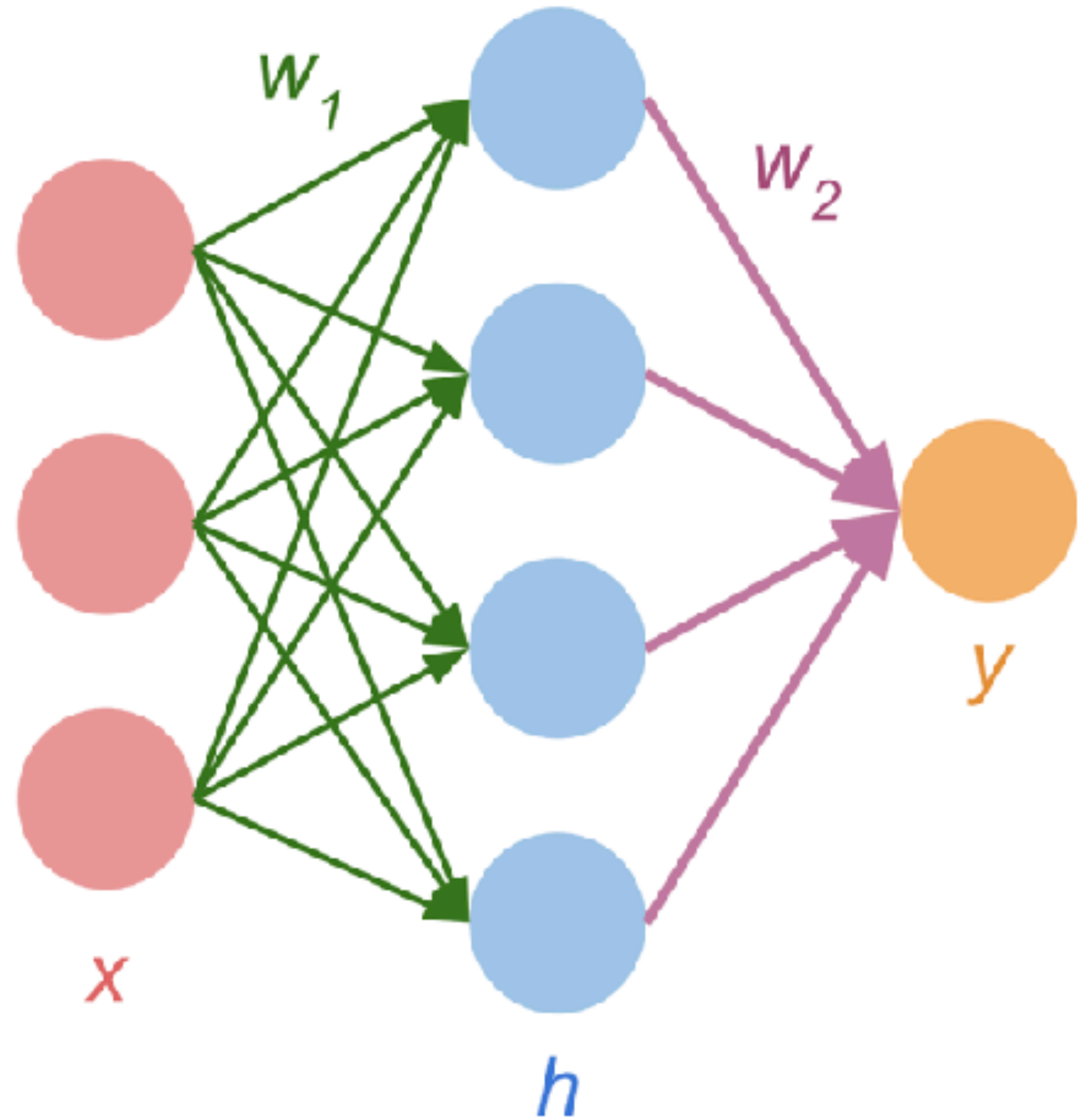
Spring 2018

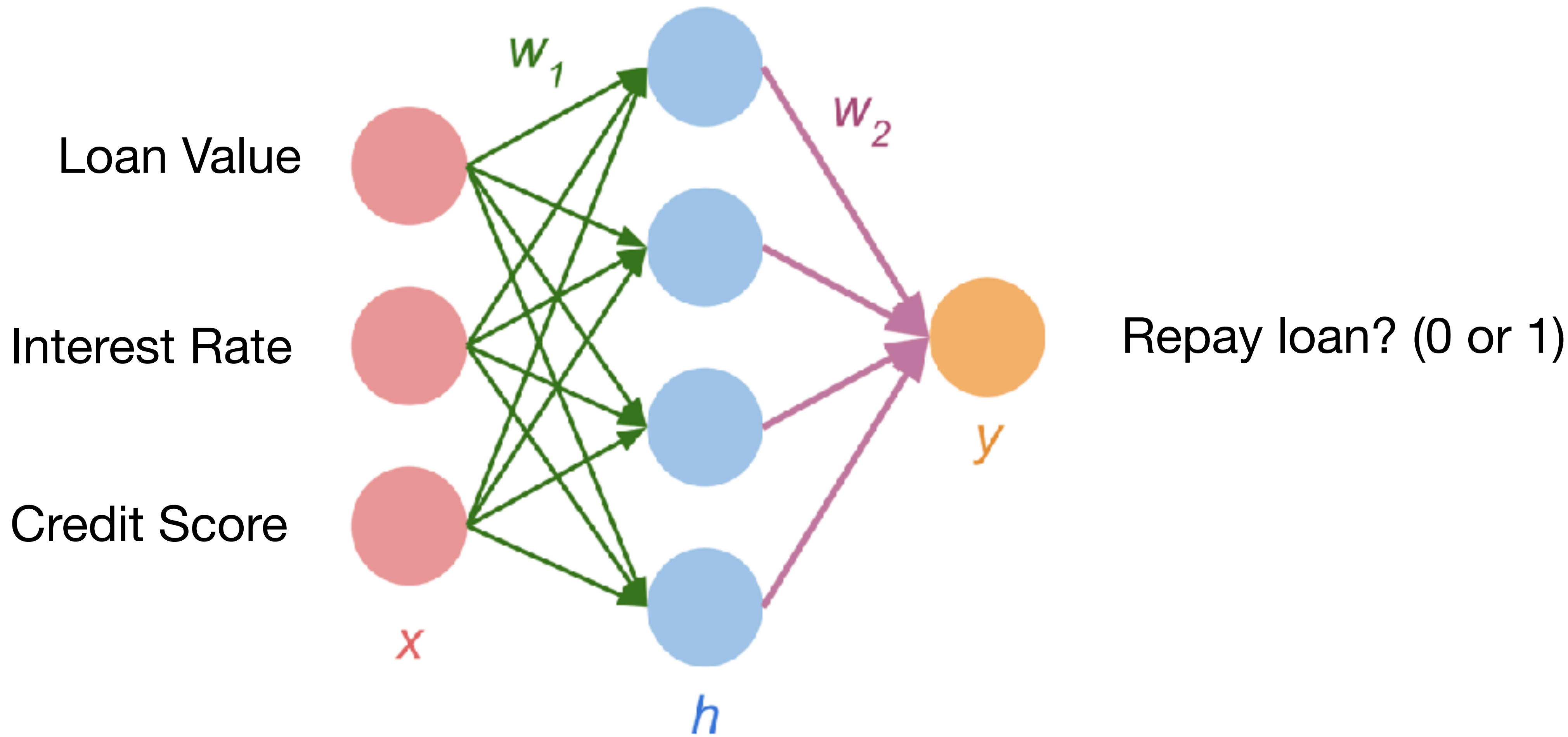
Introduction

- Our internet intelligence datasets are typically quite large.
- Big data is conducive to machine learning and deep learning.
    - Deep learning has grown exponentially in the past five years.
- Goals
    - Develop techniques that demonstrate applications of deep learning in the internet intelligence domain.
    - Shed light on these techniques with the hope that they can be applied to other datasets.

Neural Networks

- Large number of (x, y) mappings.
- Optimization over thousands of training samples.





Email →	Spam (0 or 1)
Image →	Object (0, 1, …, 1000)
Audio →	Text
English →	French
FQDN →	***Geolocation***
Peer ASN, Prefix →	***AS Path***

Recurrent Neural Networks

- Capture sequential data by storing hidden states at each time step.
- Long Short Term Memory (LSTM) Cell:
  - Uses three gates to decide what information to keep in the hidden state.

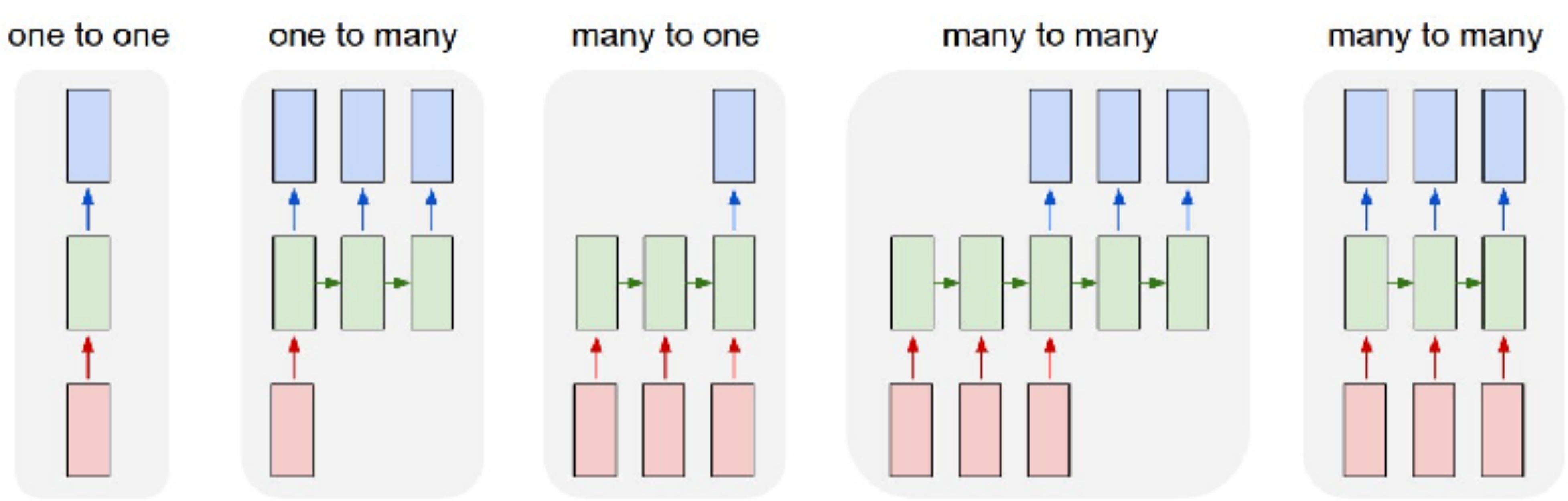


Figure Credit: Andrej Karpathy 'The Unreasonable Effectiveness of Recurrent Neural Networks'

Character Level Modeling (Char-RNN)

- Use a LSTM recurrent neural network to model a text corpus on the character level.
- Basically:
  - For each character in a string, generate some hidden state vector. Use this vector to predict the next character.
  - Read through the data many times and optimize these fancy functions for generating a hidden state and for predicting the next character.

Sonar Data:

```
ec2-52-42-105-10.us-west-2.compute.amazonaws.com   1551  60f
162-227-185-250.lightspeed.elpstx.sbcglobal.net    1536  600
93.160.56.59.broad.fz.fj.dynamic.163data.com.cn    0285  11d
14.143.100.113.static-pune.vsnl.net.in  0117  075
static-68-129-212-164.nycmny.fios.verizon.net      1327  52f
43.84.85.117.broad.wx.js.dynamic.163data.com.cn    0198  0c6
host-209-214-83-176.mem.bellsouth.net   0983  3d7
ec2-34-218-143-190.us-west-2.compute.amazonaws.com 1551  60f
75-142-7-74.dhcp.mdfd.or.charter.com     1557  615
99-45-230-241.lightspeed.wepbfl.sbcglobal.net      0738  2e2
dialup-4.197.106.137.dial1.detroit1.level3.net     1216  4c0
ec2-54-184-139-47.us-west-2.compute.amazonaws.com  1551  60f
pool-74-104-169-151.bstnma.fios.verizon.net  1156  404
pool-71-183-106-71.nycmny.fios.verizon.net   1327  52f
115.111.51.195.static-mumbai.vsnl.net.in     0130  082
pool-71-167-193-166.nycmny.east.verizon.net  1327  52f
```

Train on
entire dataset
many times

Generated Data

```
adsl-65-9-138-97.mia.bellsouth.net       0705     2c1
s0106a84e3f6a2903.cg.shawcable.net       1603     643
69-243-125-173.lightspeed.wepbfl.sbcglobal.net  0738     2e2
server-52-46-42-74.ystl.revd001.uk.gq1.yahoo.com         0561     210
dialup-4.228.93.250.dial1.cincinnati1.level3.net         0927     39f
adsl-99-49-82-51.dsl.stlsmo.sbcglobal.net       0871     367
adsl-67-33-207-175.chs.bellsouth.net     0892     37c
109x194x221x224.dynamic.ryazan.ertelecom.ru     0033     021
3.52.78.218.dial.xw.sh.dynamic.163data.com.cn   0220     0dc
adsl-69-214-14-44.dsl.chcgil.ameritech.net      1117     45d
13.8.82.218.broad.xw.sh.dynamic.163data.com.cn  0220     0dc
75-1-204-49.lightspeed.bcvloh.sbcglobal.net     1334     536
ec2-34-218-204-121.us-west-2.compute.amazonaws.com       1551     60f
ec2-52-211-87-120.eu-west-1.compute.amazonaws.com        0552     228
```

Given any *variable-length* piece of text, we can generate a *fixed-length vector* that captures the information in that text.

Let's do something more useful.

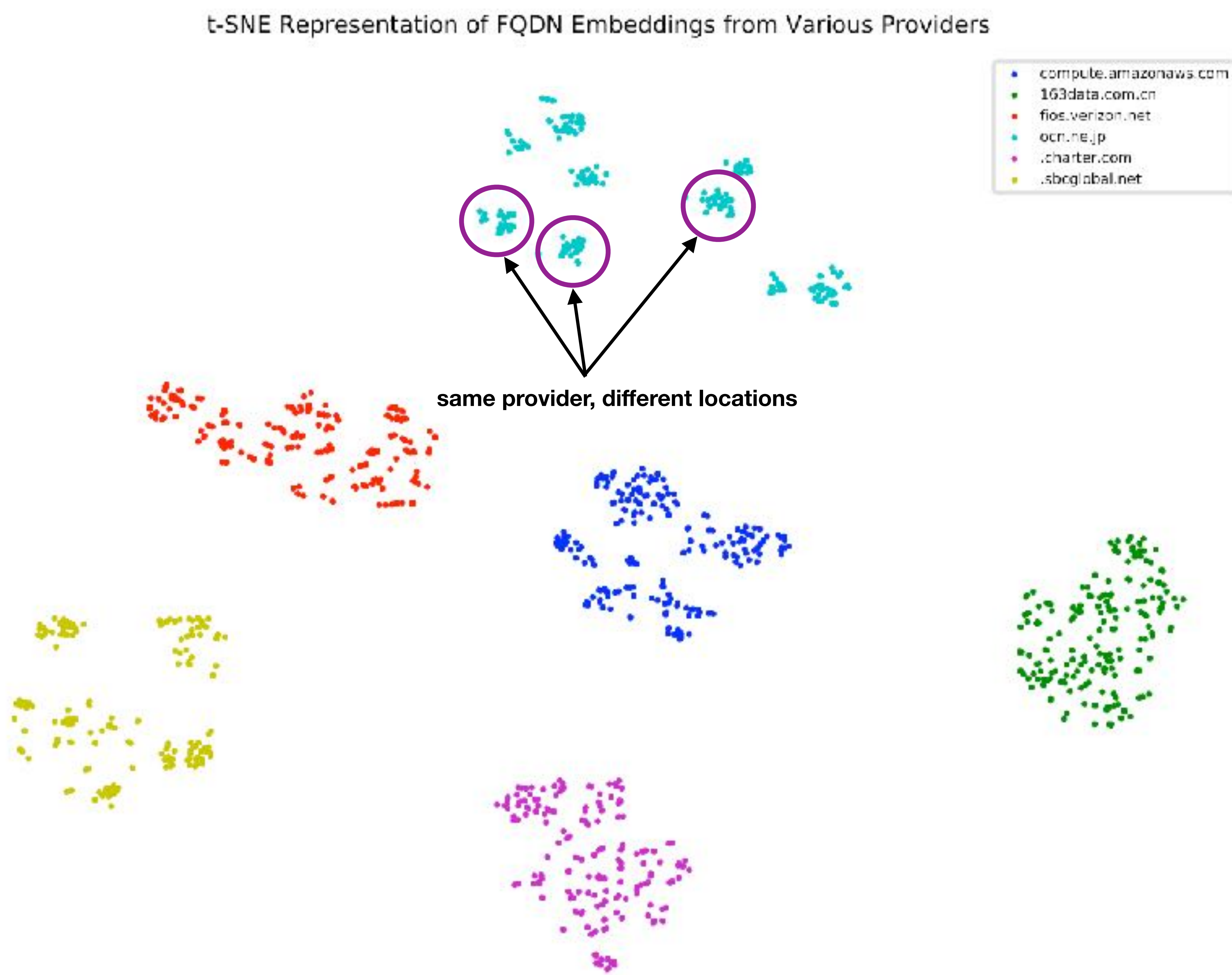
- Using char-RNN, we generate a 128-dimensional vector using each FQDN as input.
- What does this embedding look like?
  - To a human, just random numbers.
  - But, **similar FQDNs get similar embeddings.**

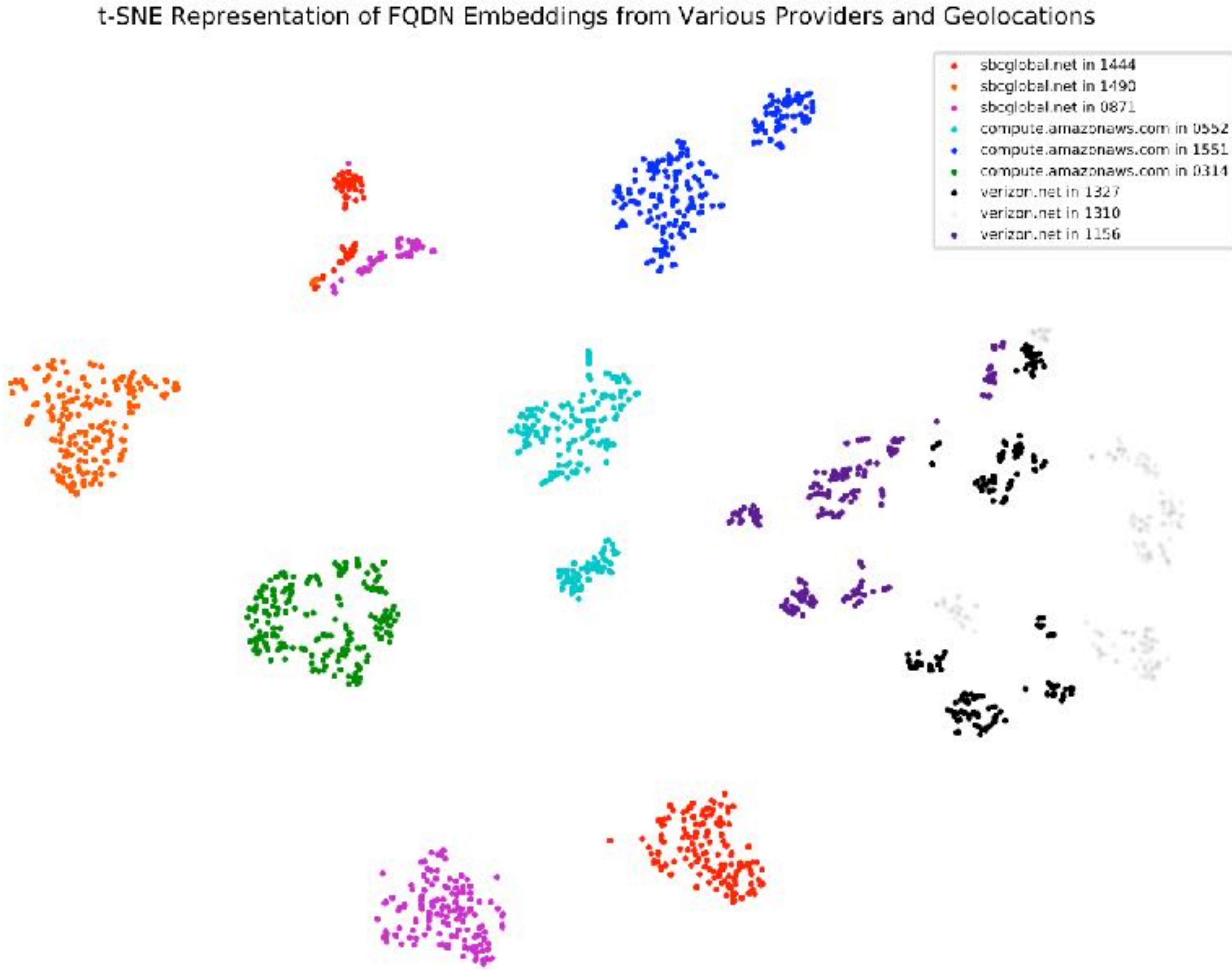
FQDN	Geolocation
ec2-52-194-101-41.ap-northeast-1.compute.amazonaws.com	0314
187.40.30.117.broad.xm.fj.dynamic.163data.com.cn	0197
14.143.79.54.static-hyderabad.vsnl.net.in	0124
adsl-072-156-044-189.sip.bct.bellsouth.net	0672
ec2-35-162-185-148.us-west-2.compute.amazonaws.com	1551
p232037-ipngn1701akita.akita.ocn.ne.jp	0399
107-214-70-135.lightspeed.chrlnc.sbcglobal.net	0893

We have over 100 million FQDN-Geolocation pairs.

Visualizing Embeddings

- t-SNE: project high dimensional data into 2D space while preserving clustering.

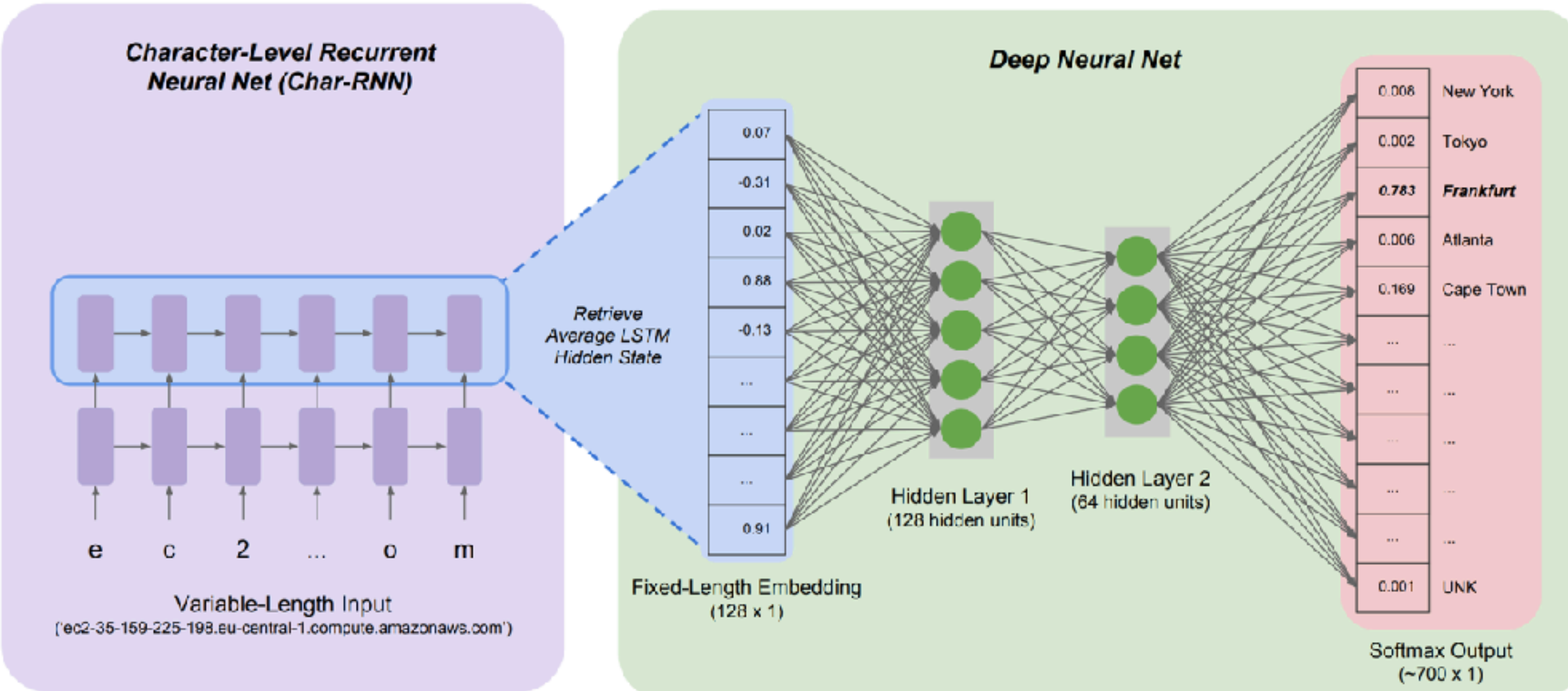

t-SNE Representation of FQDN Embeddings from Various Providers
compute.amazonaws.com
163data.com.cn
fios.verizon.net
ocn.ne.jp
.charter.com
.sbcglobal.net
same provider, different locations


t-SNE Representation of FQDN Embeddings from Various Providers and Geolocations
sbcglobal.net in 1444
sbcglobal.net in 1490
sbcglobal.net in 0871
compute.amazonaws.com in 0552
compute.amazonaws.com in 1551
compute.amazonaws.com in 0314
verizon.net in 1327
verizon.net in 1310
verizon.net in 1156

Geolocation Prediction

1. Given a FQDN, use Char-RNN to generate an embedding.

2. Run the embedding through a deep neural net followed by a softmax output to predict the geolocation associated with the original FQDN.



Character-Level Recurrent Neural Net (Char-RNN)
e
c
2
...
o
m
Variable-Length Input
('ec2-35-159-225-198.eu-central-1.compute.amazonaws.com')
Retrieve Average LSTM Hidden State
Deep Neural Net
0.07
-0.31
0.02
0.88
-0.13
0.91
Fixed-Length Embedding
(128 x 1)
Hidden Layer 1
(128 hidden units)
Hidden Layer 2
(64 hidden units)
0.008
New York
0.002
Tokyo
0.783
Frankfurt
0.006
Atlanta
0.169
Cape Town
0.001
UNK
Softmax Output
(~700 x 1)

Dataset 1: Reverse DNS (Sonar)

- (x, y) = (fqdn, geolocation)
- Subset of DNSGEO data (city-level rules)
- 108 million data points
- Compress to 1.8 million samples from 673 geolocations where:
    - Between 500 and 3000 samples for each geolocation

Training

Char-RNN Metrics

Training Samples	108 million
Training Time	2 days
Epochs	1
Number of Unrollings	20

Neural Net Training Metrics

Total Samples	1.8 million
Training Samples	1.75 million
Validation Samples	50,000
Embedding Generation Time	12 hours
Training Time	10 min
Epochs	10
Training Accuracy	93%
Validation Accuracy	**92%**

Looking at some examples...

FQDN	Prediction	Real
71-13-212-176.dhcp.mrqt.mi.charter.com	**1229 (99%)** 1557 (1%) 1243 (10^{-6}) 0866 (10^{-7}) 1278 (10^{-7})	1229
adsl-072-148-062-151.sip.ilm.bellsouth.net	**0661 (57%)** 0678 (35%) 0877 (7%) 1610 (0.5%) 0356 (0.1%)	0661
144.69.219.222.broad.bs.yn.dynamic.163data.com.cn	0181 (27%) **0137 (21%)** 0294 (18%) 0136 (16%) 0244 (4%)	0137

Our neural network predicts the geolocation of a FQDN with 92% accuracy.

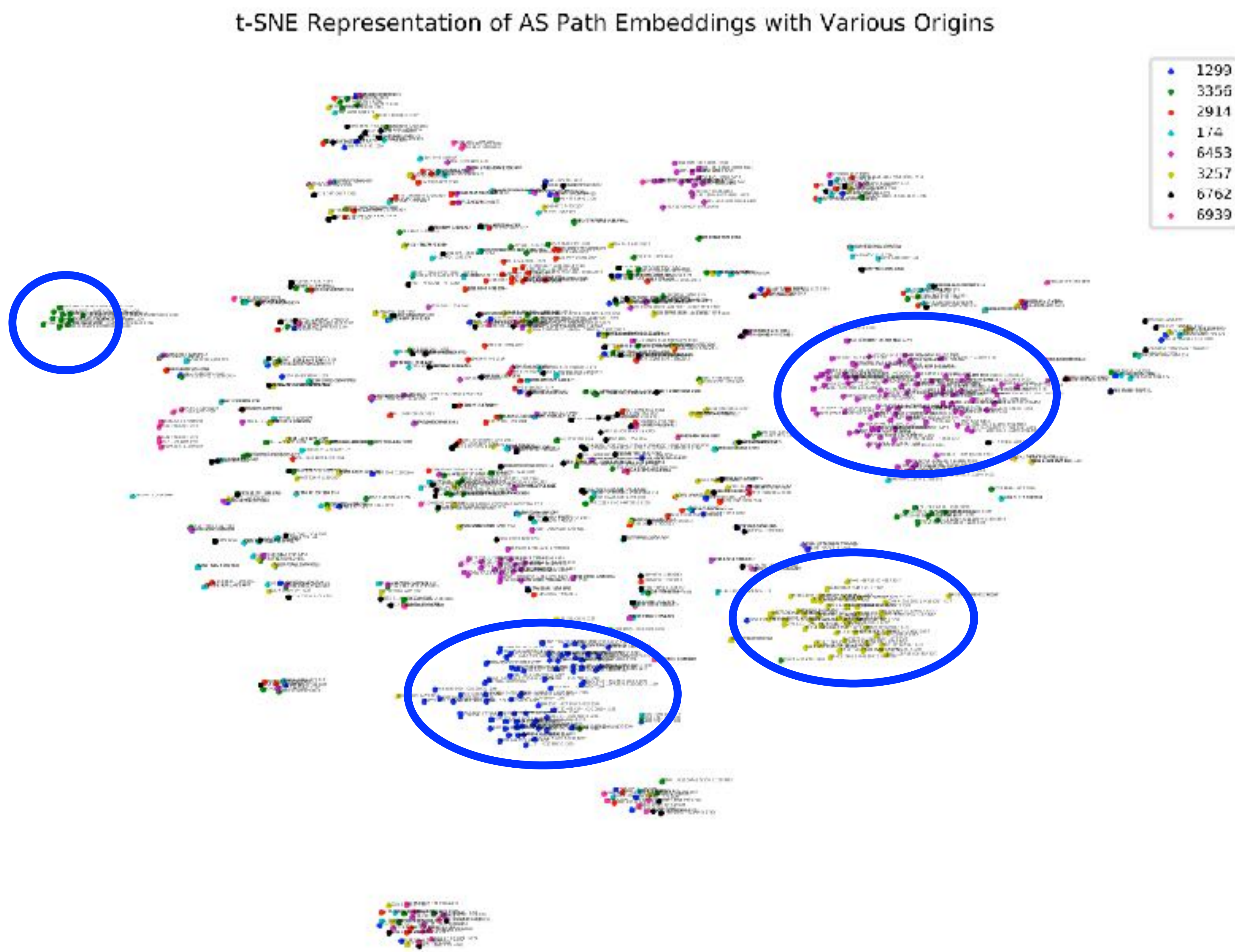
Dataset 2: Select Cities

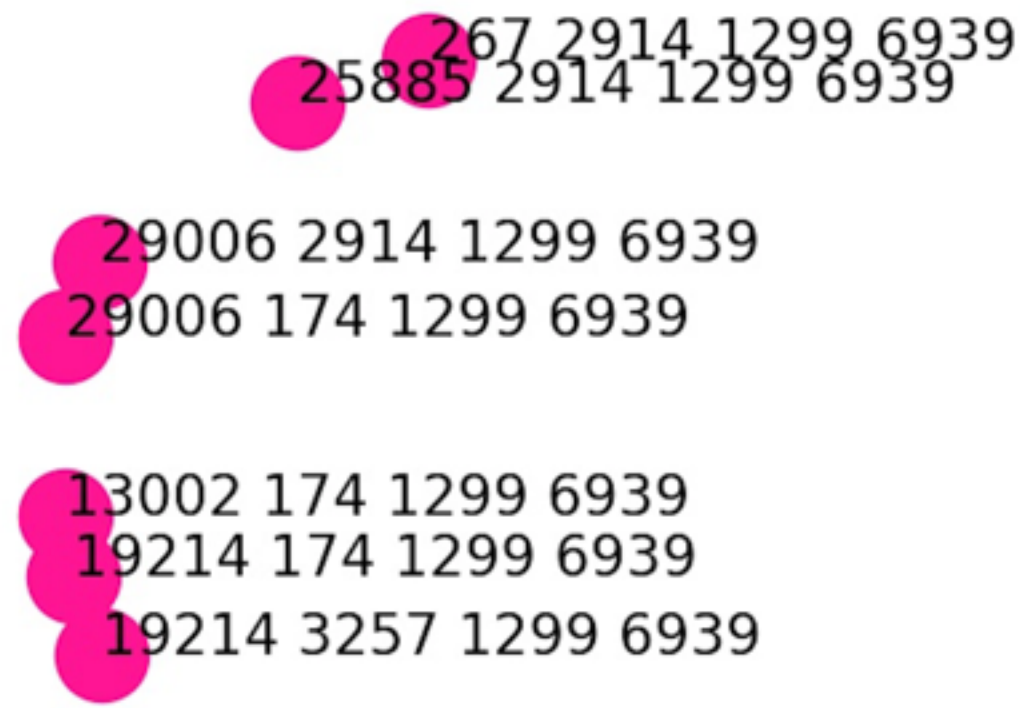
- (x, y) = ([fqdn, associated bgp paths], geolocation)
- 2.8 million samples from 213 cities
    - Around 13,000 samples per city
- Varied geolocation confidence
    - DNS Geo data (city-level rules)
    - Airport code substrings
    - Newt geolocation system
- “Harder” dataset

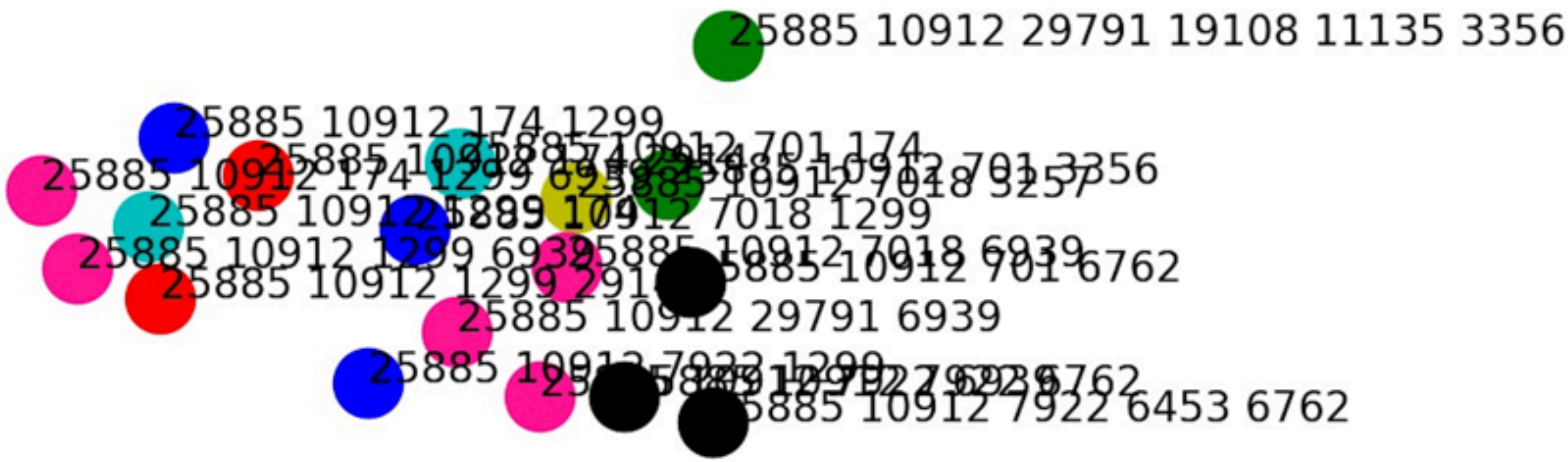
- Variable-length paths from peer to origin.

BGP Data:

```
51185 174 14277 395819
2381 3356 12956 10429 28606
2119 3356 197541
8001 3257 200612 12880 59703 16322 60976 48551 12697
209 3257 8455 40490
12389 1299 16735 262354 262688 262821
1248 1299 174 16735 28666 53018 61901
3265 2914 3356 3549 28598 262993 264493
15772 6939 31133 24955 42498
39912 6762 1299 7843 11427 14736
8473 8400 25144 16178 9146
20473 3257 6453 30844 327744
10158 1299 7018 21547 396522
8359 2497 9605
19255 2828 3491 23947 45727
23148 1273 30722 44957
7598 2914 9002 39775 47271
29017 3356 7018 36753
30844 6939 12956 22927 264634
```

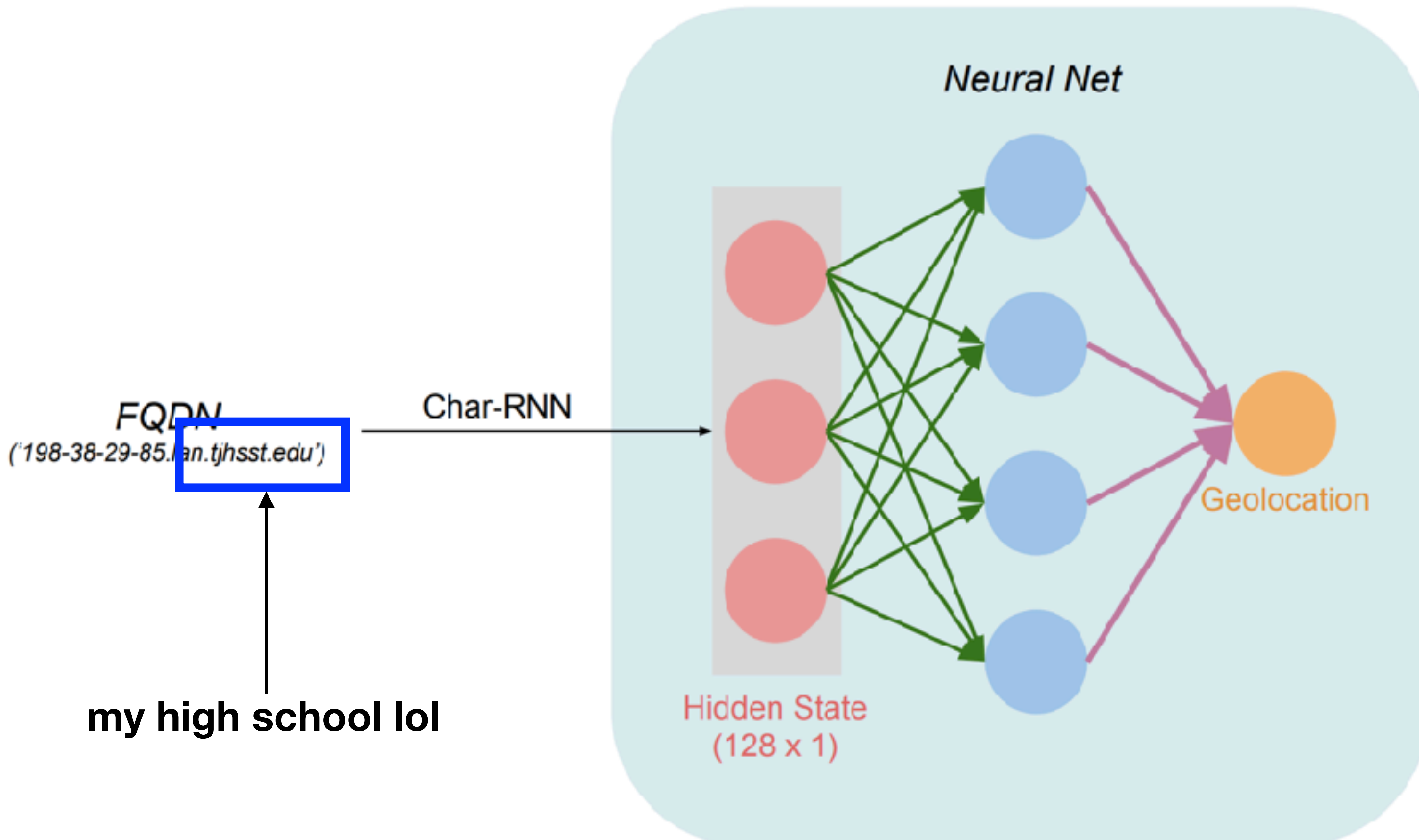

t-SNE Representation of AS Path Embeddings with Various Origins
1299
3356
2914
174
6453
3257
6762
6939


267 2914 1299 6939
25885 2914 1299 6939
29006 2914 1299 6939
29006 174 1299 6939
13002 174 1299 6939
19214 174 1299 6939
19214 3257 1299 6939

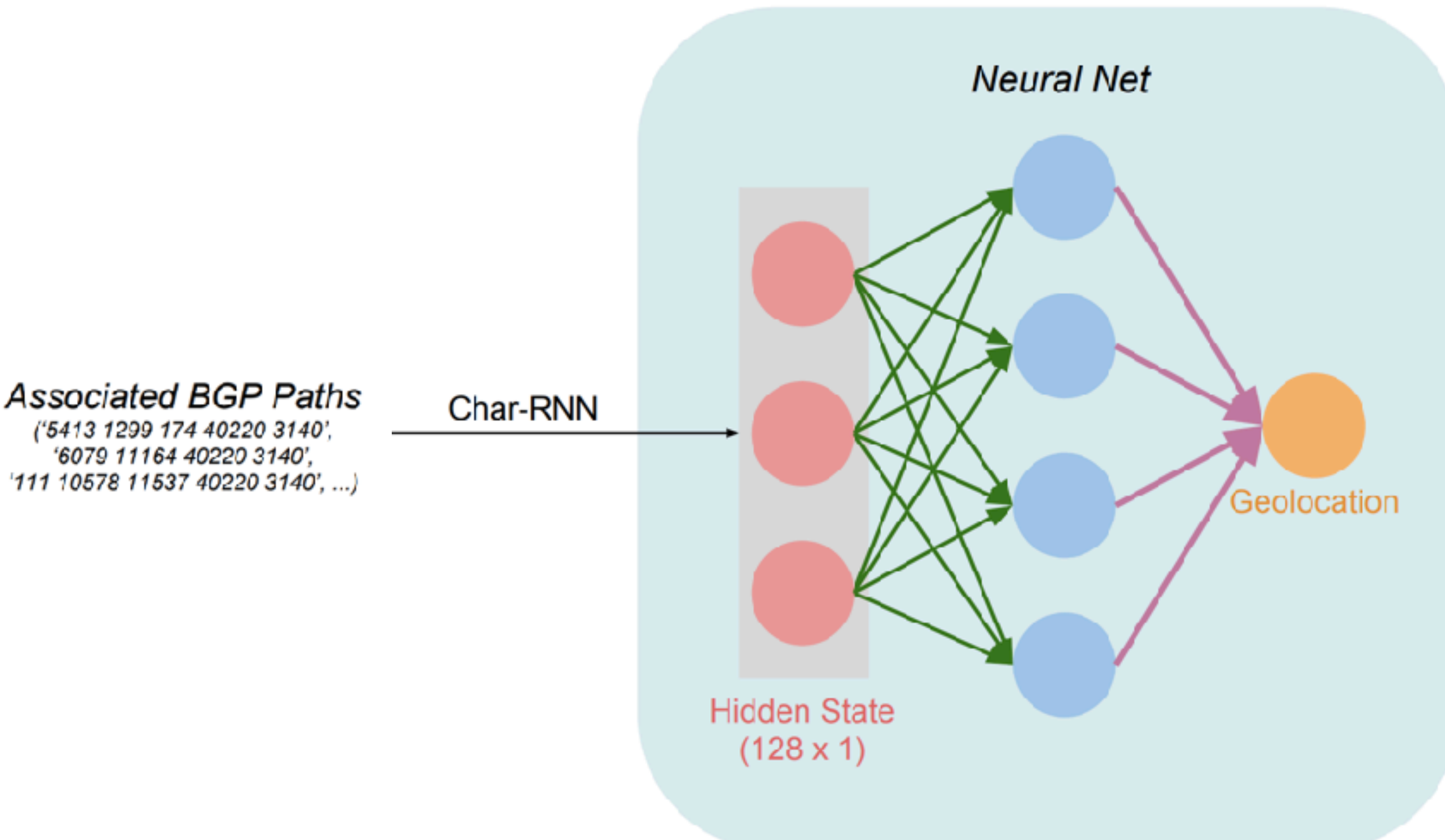


25885 10912 29791 19108 11135 3356
25885 10912 174 1299
25885 10912 29791 6939

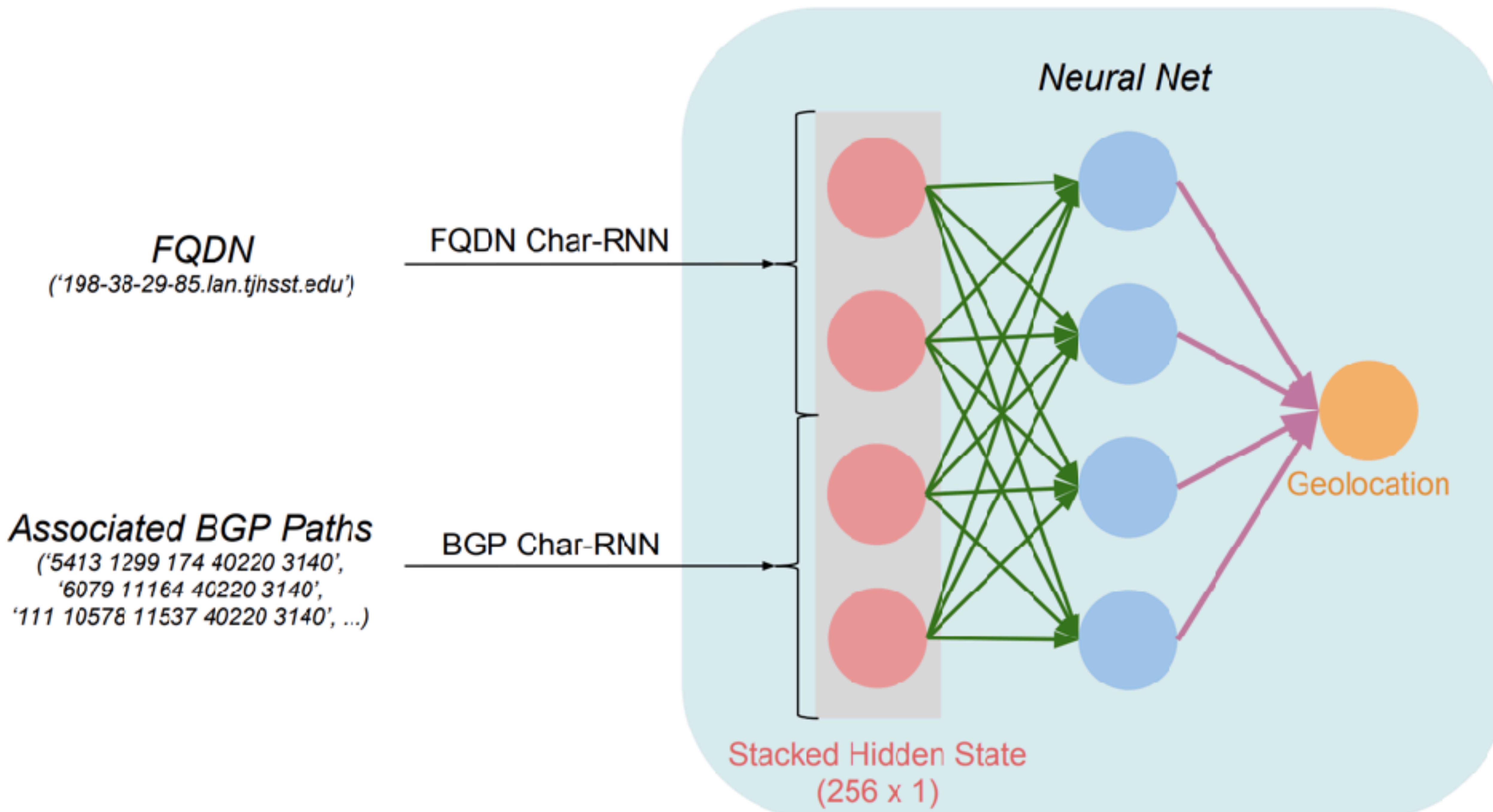
Models: only FQDN



Models: only BGP Paths



Models: Combined Model



Training

FQDN Char-RNN Metrics

Training Samples	100 million
Training Time	2 days
Epochs	1
Number of Unrollings	20

BGP Char-RNN Metrics

Training Samples	30.3 million
Training Time	1 day
Epochs	1
Number of Unrollings	20

Neural Net Training Metrics

Total Samples	2.8 million
Training Samples	2.75 million
Validation Samples	50,000
FQDN Embedding Time	1 day
BGP Path Embedding Time	a long time*
Training Time	10 min
Epochs	10

*2 days with six processes running in parallel

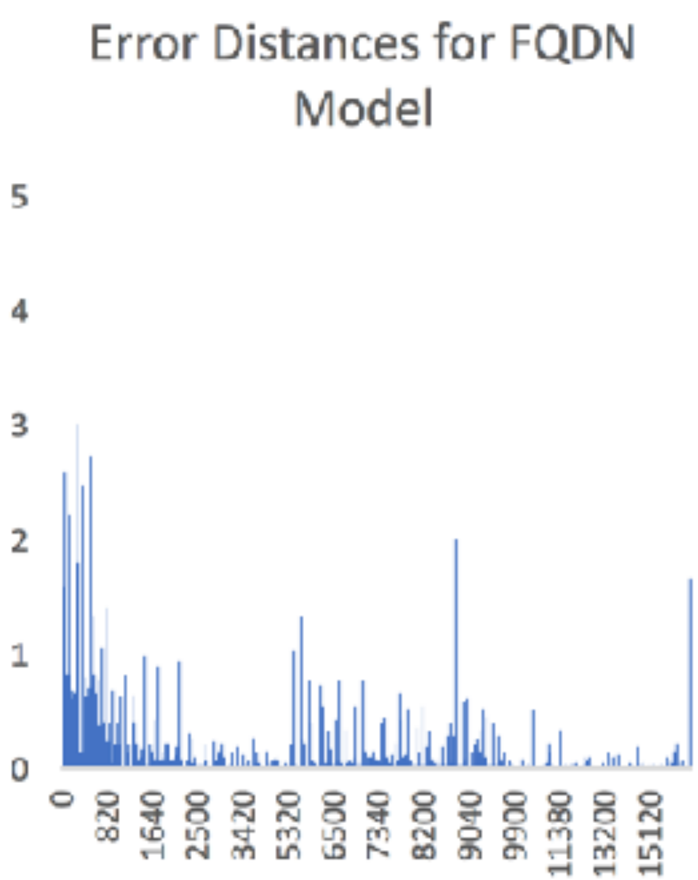
Results

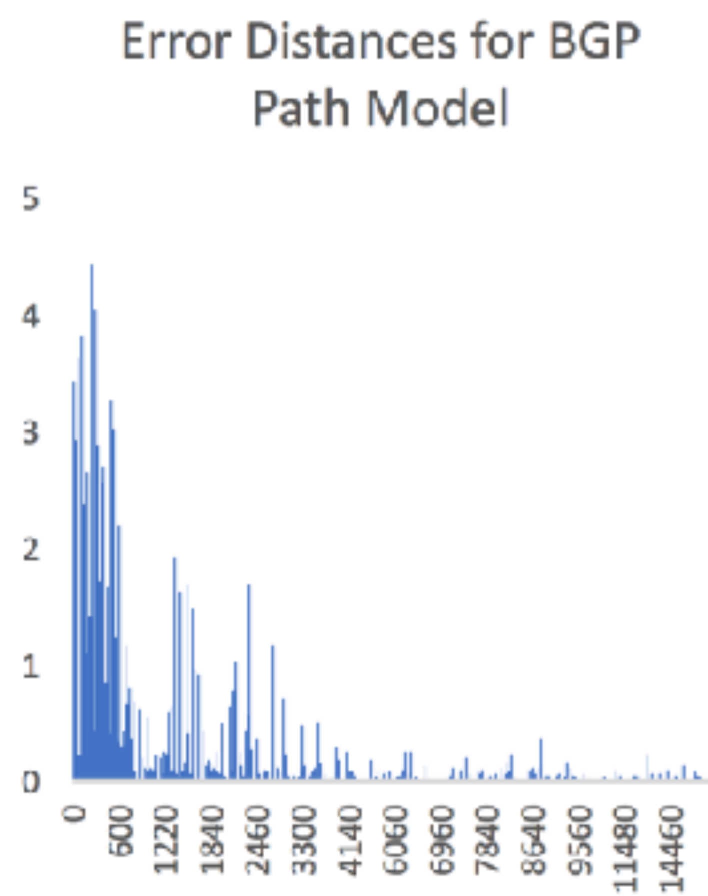
Accuracy of FQDN, BGP, and Combined Models

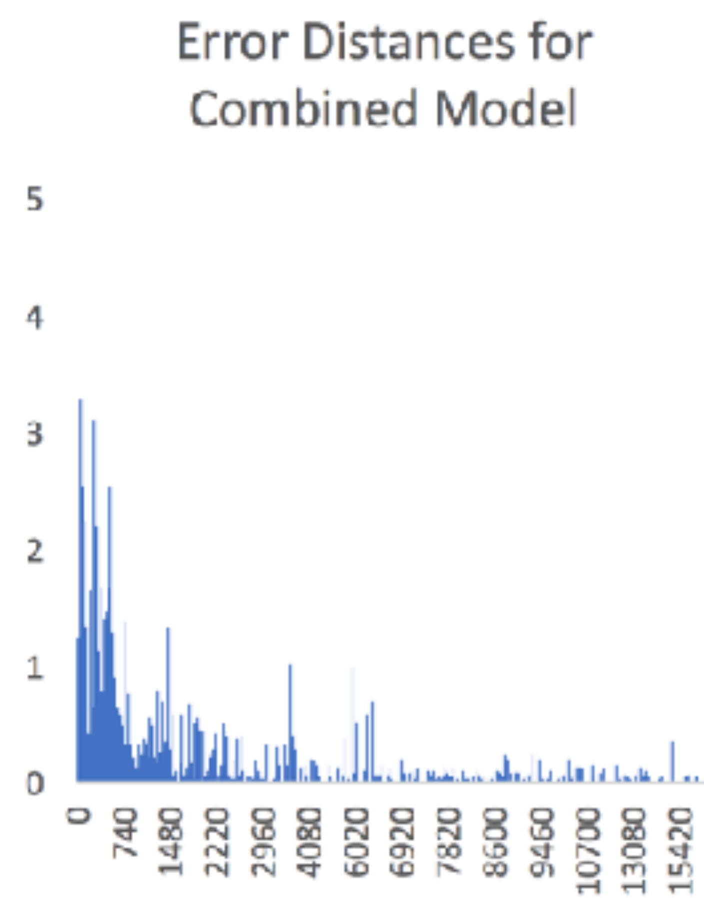
Model	Top 1	Top 3	Top 10
FQDN	67.0% (69.9%)	80.1% (82.8%)	87.5% (90.2%)
BGP	57.9% (57.0%)	78.7% (76.5%)	**94.2%** (94.1%)
Combined	**67.3%** (66.7%)	**85.2%** (85.1%)	92.8% (94.5%)

Key: Validation Acc% (Training Acc%)

Error Distances







So what?

- With improvements (more training data), our neural net could be able to predict the geolocations of rare or unknown FQDNs.
- Our ability to embed variable-length text information into a fixed-length numeric vector is transferable to other data.

What's next?

- Improving testing accuracy on new data. Our model is limited to predicting geolocation only for data it has seen before.
- Eventually hope to replace manual rule-based approach with an automated data-driven approach.
- Predicting AS paths.
- Latency distributions.
- Predicting traceroute hops from collector and target IP.
- More machine learning!

Thank you.