

Sequence Prediction with Recurrent Neural Networks: Predicting the Next Airport a Plane Will Visit



Group 1

Gayathri Chandrasekaran, Jonathan Giguere, Patrick Maus

Outline

- Introduction
- Description of the Dataset
- Description of the Deep Learning Network and Training Algorithm
- Experimental Setup
- Results
- Summary and Conclusions
- References

Information 19:10			
Time	Flight	Status	Gate
18:55	Glasgow BA1486	Flight closing	A11
19:00	Amsterdam BA432	Boarding	B34
19:20	Madrid AA6747	Boarding	C56
Operated by Iberia			
19:25	Delhi AA6663	Flight closing	B46
19:35	Oslo AY5940	Delayed	B90
19:40	Brussels JL7705	Boarding	A8
19:40	Amsterdam BA444	Go to Gate	A19
19:40	Gothenburg AA6625	Go to Gate	A21
19:40	Frankfurt AB5026	Delayed	B84
19:45	New York AA6146	Flight closing	B38
19:45	Zurich AB5072	Go to Gate	C57
19:55	Singapore IB7538	Go to Gate	C65
19:55	Muscat AA6421	Go to Gate	B42
19:55	Boston AA6176	Go to Gate	B44
19:55	Geneva AA6348	Go to Gate	A1
20:05	Edinburgh AB5162	Delayed	B38
20:05	Stockholm BA786	Delayed	B51
20:15	Kuala Lumpur IB4624	Go to Gate	B36
20:15	Venice AA6526	Delayed	B38
20:15	Basel AB5078	Gate shown	19:25
20:20	Luanda IB4729	Go to Gate	B48
20:20	Bologna AA6716	Gate shown	19:37
20:25	Manchester JL7741	Delayed	B38
20:30	Munich BA958	Delayed	B56
20:35	Paris (CdG) AA6303	Gate shown	19:45
20:35	Toulouse AA6273	Delayed	B44
20:35	Milan (Malpensa) BA582	Gate shown	19:45
20:35	Dublin EI8326	Delayed	B22
20:35	Hanover AA6730	Delayed	B57
20:50	Athens AA6326	Gate shown	20:00
20:55	Newcastle BA1338	Delayed	B73
20:55	Glasgow AY5919	Delayed	B73
21:00	Aberdeen BA1318	Delayed	B22
21:10	Tehran BA153	Delayed	B73
21:20	Dubai BA109	Delayed	B73
21:25	Johannesburg AA6753	Gate shown	20:10
21:25	Mumbai AA6659	Delayed	B73
21:35	Sydney IB4745	Delayed	B73
via: Singapore			
21:35	Glasgow BA1496	Delayed	B38
21:45	Hong Kong BA027	Delayed	B73
21:50	Edinburgh AA6646	Gate shown	21:00
21:50	São Paulo IB4741	Delayed	B73
21:55	Kuwait AA6599	Gate shown	20:40
22:10	Doha BA123	Gate shown	20:55
22:20	Dubai EI8805	Gate shown	21:05
22:25	Buenos Aires IB4721	Gate shown	21:10
22:30	Jeddah IB4725	Gate shown	21:15
22:30	Tel Aviv BA163	Gate shown	21:15
22:35	Abuja BA083	Gate shown	21:20
22:35	Moscow 574004	Gate shown	21:20
Flights for Monday, October 23 2017			
06:20	Madrid AA6276	Gate shown	05:30
06:35	Brussels BA388	Gate shown	05:45
06:40	Amsterdam BA428	Gate shown	05:50
06:45	Geneva AA6334	Gate shown	05:55
06:45	Dublin EI8332	Gate shown	05:55
06:55	Athens AA6329	Gate shown	06:05
06:55	Munich AB5036	Gate shown	06:05
07:00	Edinburgh QF3551	Gate shown	06:10
07:00	Rome BA552	Gate shown	06:10
07:00	Istanbul BA678	Gate shown	06:10
07:00	Zurich AB5062	Gate shown	06:10
07:05	Paris (Only) AA6702	Gate shown	06:15
07:05	Copenhagen BA812	Gate shown	06:15
07:05	Frankfurt AB5012	Gate shown	06:15
07:10	Newcastle CX1211	Gate shown	06:20
07:10	Belfast City BA1414	Gate shown	06:20
07:10	Hamburg BA964	Gate shown	06:20

Introduction: Data Overview and Problem Statement

- Focus on use of RNNs to predict next values in a sequence (outside of NLP).
- “Given a sequence of N airports visited by a unique plane, can we predict the next airport ($N+1$)?”
- The best performing model was a single layer LSTM.

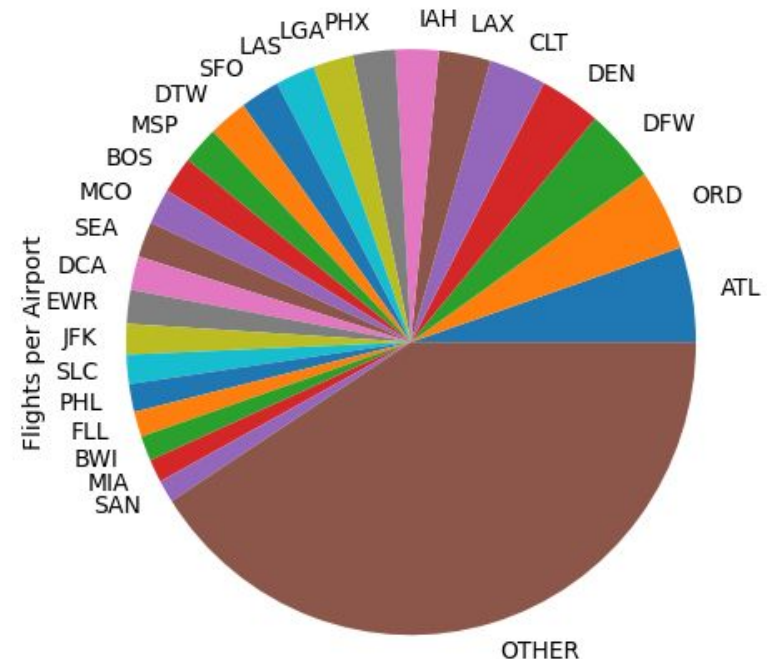


Description of the Dataset

- The data set was downloaded from the Bureau of Transportation Statistics, a US Department of Transportation agency charged to collect, analyze, and store transportation statistics.
- Using six months of data from September 2019 to February 2020, we reviewed about 3.7 million flights from 5,716 unique aircraft. We choose to only use data up to February 2020 to avoid impacts from the Coronavirus pandemic to affect the sequence patterns.
- Within our data, there are 19 different airlines, but the top five represents about 2/3 of the entire data. Several sample rows are included to the right.

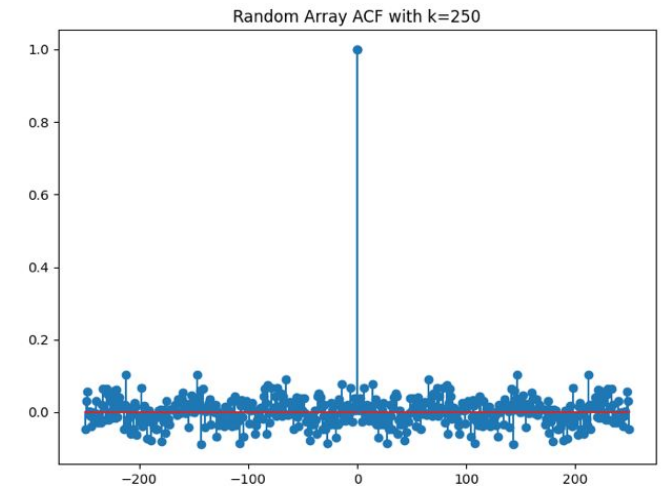
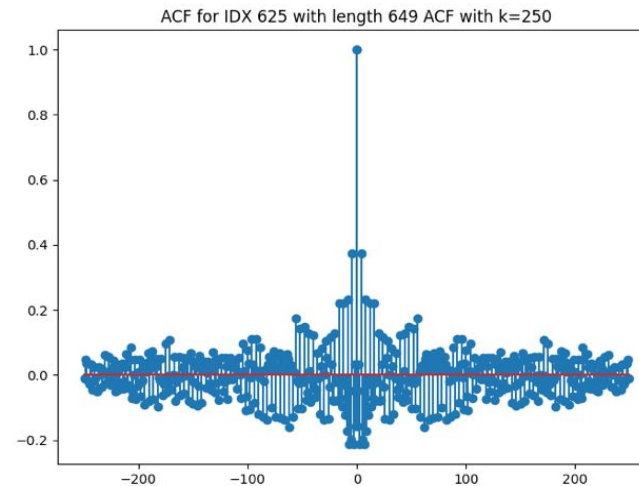
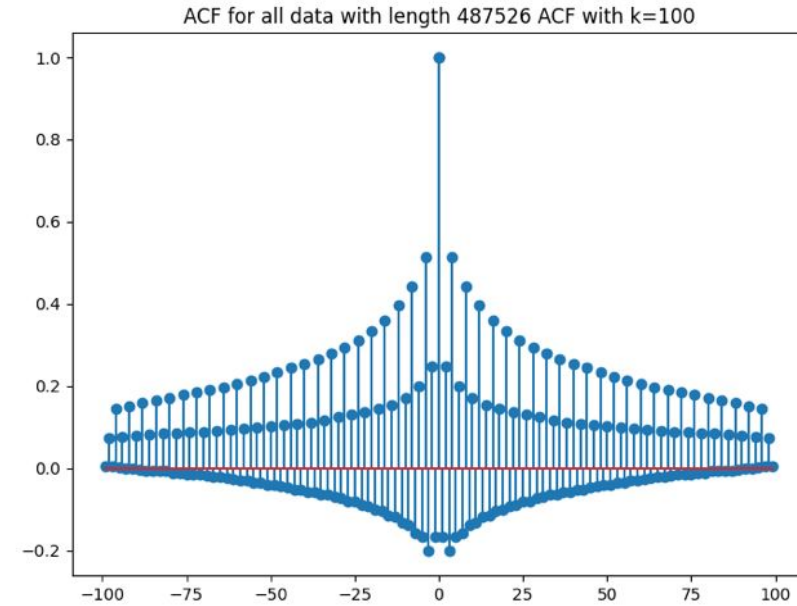
FL_DATE	OP UNIQUE CARRIER	TAIL NUM	OP CARRIER FL NUM	ORIGIN	DEST	DEP TIME	ARR TIME
2020-02-01	MQ	N269NN	3825	ORD	TUL	1646	1820
2020-02-01	MQ	N908AE	3829	JFK	BNA	1336	1458
2020-02-01	MQ	N663AR	3831	GNV	MIA	844	1020
2020-02-01	MQ	N618AE	3833	DFW	SJT	852	955
2020-02-01	MQ	N618AE	3833	SJT	DFW	1024	1132

Percent of Flight From Top 25 Airports



Description of the Dataset: Autocorrelation and Stationarity

- An autocorrelation plot shows how well a signal correlates with itself at equal time lags. If the plot shows an impulse of 1 at time lag 0 but then very little or know correlation at subsequent time lags, the data is likely to be random white noise. On the other hand, an autocorrelation plot with strong correlation for multiple lags indicates that previous events influence the current event and suggest an underlying pattern.
- The Augmented Dickey-Fuller Test can help us determine if we should treat a dataset as stationary. 94% of Delta airlines aircraft were stationary within our dataset.



EDA and Cleaning	ACF and Stationarity Checks	Preprocessing	Modeling	Model Evaluation
<p><u>Input:</u> CSVs downloaded from Bureau of Trans. Stats.</p> <p><u>Actions:</u></p> <ul style="list-style-type: none"> - Concat csvs together - Explore and summarize counts. - Build the history of each unique airplane as a string. <p><u>Output:</u> Aircraft and string of all visited airports (DataFrame)</p>	<p><u>Input:</u> Aircraft and string of all visited airports (DataFrame)</p> <p><u>Actions:</u></p> <ul style="list-style-type: none"> - Plot ACF for all data, across airlines, and each plane. - Use ADF Test for all data, across airlines, and each plane. <p><u>Output:</u> ACF plots and ADF test results (print to screen)</p>	<p><u>Input:</u> Aircraft and string of all visited airports (DataFrame)</p> <p><u>Actions:</u></p> <ul style="list-style-type: none"> - Arrays of airports are broadcast across a window length (L), and then combined. - Airports tokenized. - Target separated and one-hot encoded. <p><u>Output:</u> Tokenizer (dict), X, Y (2D arrays)</p>	<p><u>Input:</u> X, Y (2D arrays) with X.shape=(# arrays, L - 1) and Y.shape=(# arrays, 1).</p> <p><u>Actions:</u></p> <ul style="list-style-type: none"> - Set model hyperparameters. - Build, Compile, and Fit model. <p><u>Output:</u> Trained model (hdf5), history (DataFrame)</p>	<p><u>Input:</u> Tokenizer (dict), X, Y (2D arrays), Trained model (hdf5), history (DataFrame)</p> <p><u>Actions:</u></p> <ul style="list-style-type: none"> - Plot training history. - Determine accuracy and Cohen's Kappa. - Generate confusion matrix and report. - Plot scatterplot <p><u>Output:</u> Reports and plots (print to screen)</p>

Experimental Setup: Data Processing

Training Algorithm

Framework – Keras Sequential Model

Model – Recurrent Neural Networks (RNN)

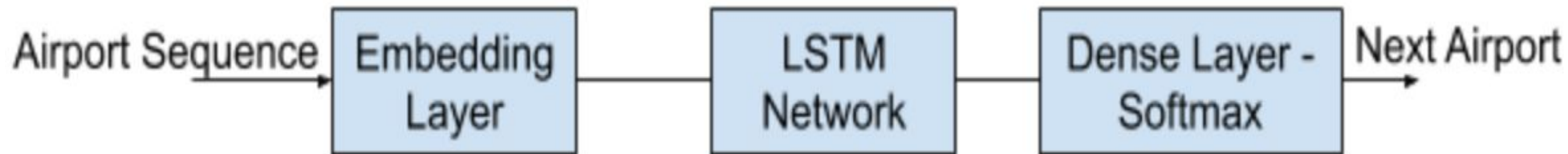
- LSTM Networks
- Stacked LSTM
- GRU
- Sequence-to-Sequence

Base Model – Random Sequences

```
Final accuracy on validations set: 24.448  
Cohen kappa score 0.0  
Precision: 0.059769239279687005  
Recall: 0.24447748215262485  
F1: 0.09605515589772128
```

Training Algorithm - Single Layer LSTM Network

Single Layer Lstm Network:



Hyper Parameters:

- Sequence Length - 25 and 50
- Number of Neurons - 400
- optimizer - Adam
- Epoch - 10
- Embedding Size - 200
- Learning Rate - $1e-3$

Model Results - Single Layer LSTM Network

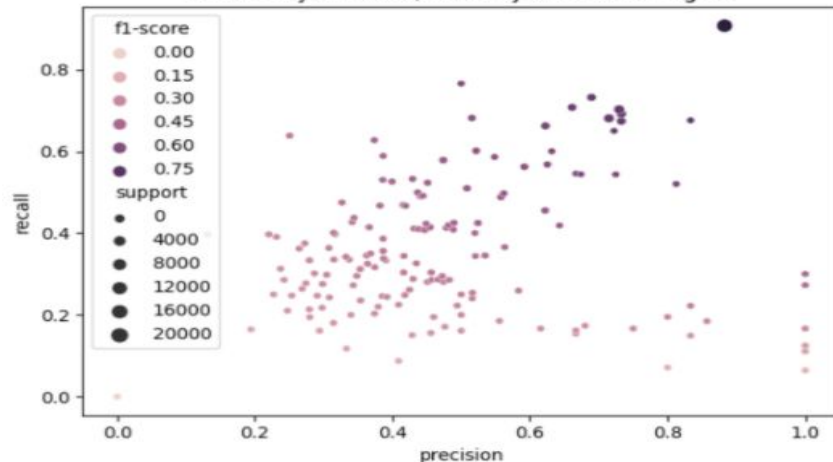
Single Layer Lstm Network:

Sequence Length - 25

Modal Evalution Results

```
Accuracy: 62.422
Cohen kappa score 0.5919228700612527
Precision: 0.6325509588477811
Recall: 0.6242243679737651
F1: 0.622350160687898
```

Scatterplot of Model's Precision and Recall,
Colored by F1 Score, Sized by Number of Flights

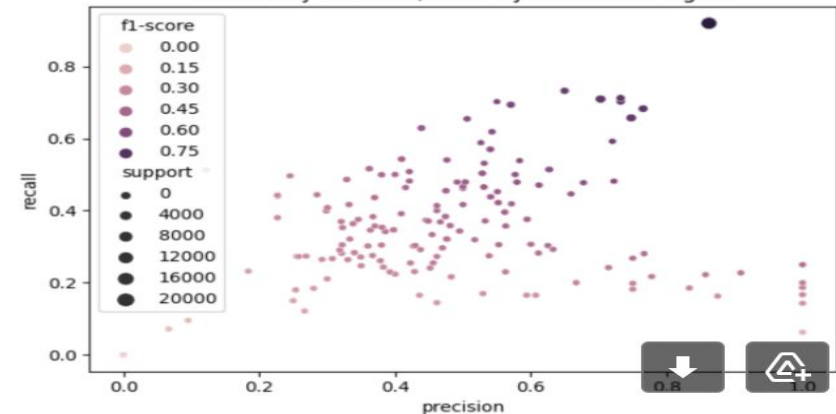


Sequence Length - 50

Modal Evalution Results

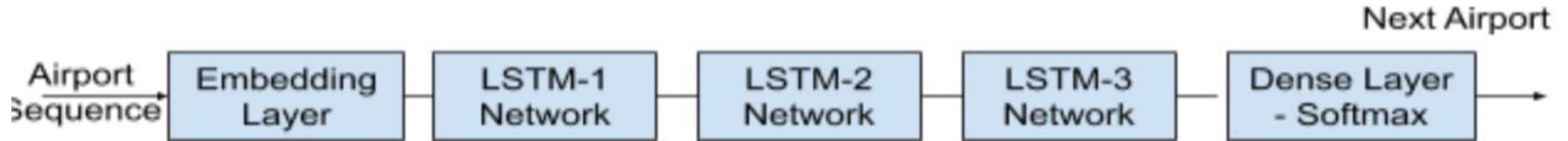
```
Accuracy: 62.646
Cohen kappa score 0.5933283419217596
Precision: 0.6360756942665542
Recall: 0.626464252053319
F1: 0.623502422167004
```

Scatterplot of Model's Precision and Recall,
Colored by F1 Score, Sized by Number of Flights



Training Algorithm - Stacked LSTM Network

Stacked Lstm Network:



Hyper Parameters:

- Sequence Length - 25 and 50
- Number of Neurons - 400 - 300 - 200
- optimizer - Adam
- Epoch - 10
- Embedding Size - 200
- Learning Rate - $1e-3$

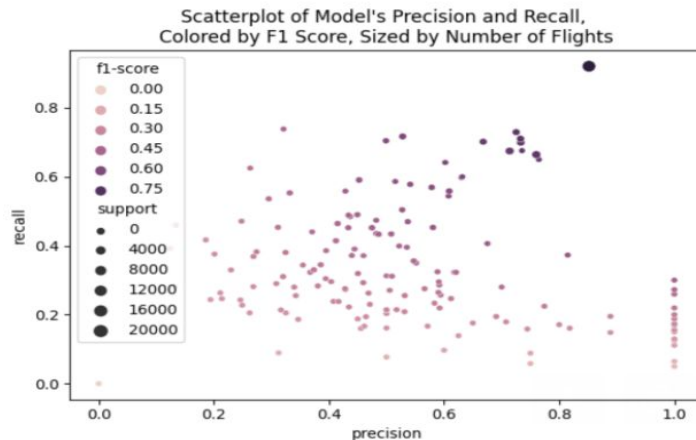
Model Results – Stacked LSTM Network

Stacked Lstm Network:

Sequence Length – 25

Modal Evalution Results

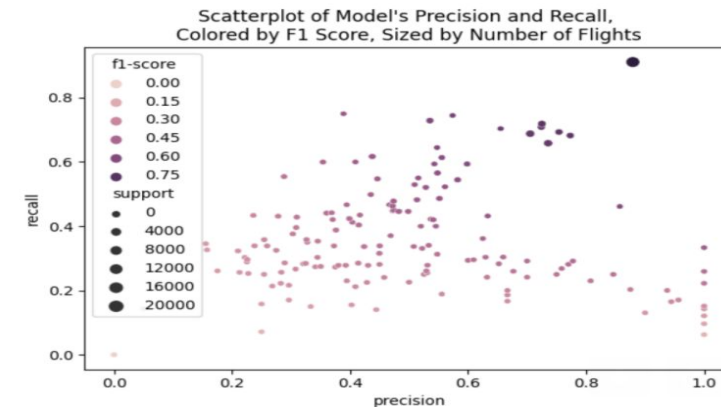
```
Accuracy: 61.911  
Cohen kappa score 0.5850593497951324  
Precision: 0.6364303404942205  
Recall: 0.619112430474435  
F1: 0.6149982165744742
```



Sequence Length – 50

Modal Evalution Results

```
Accuracy: 62.227  
Cohen kappa score 0.5895667642018765  
Precision: 0.636641565032489  
Recall: 0.622267851532696  
F1: 0.6209776412113642
```

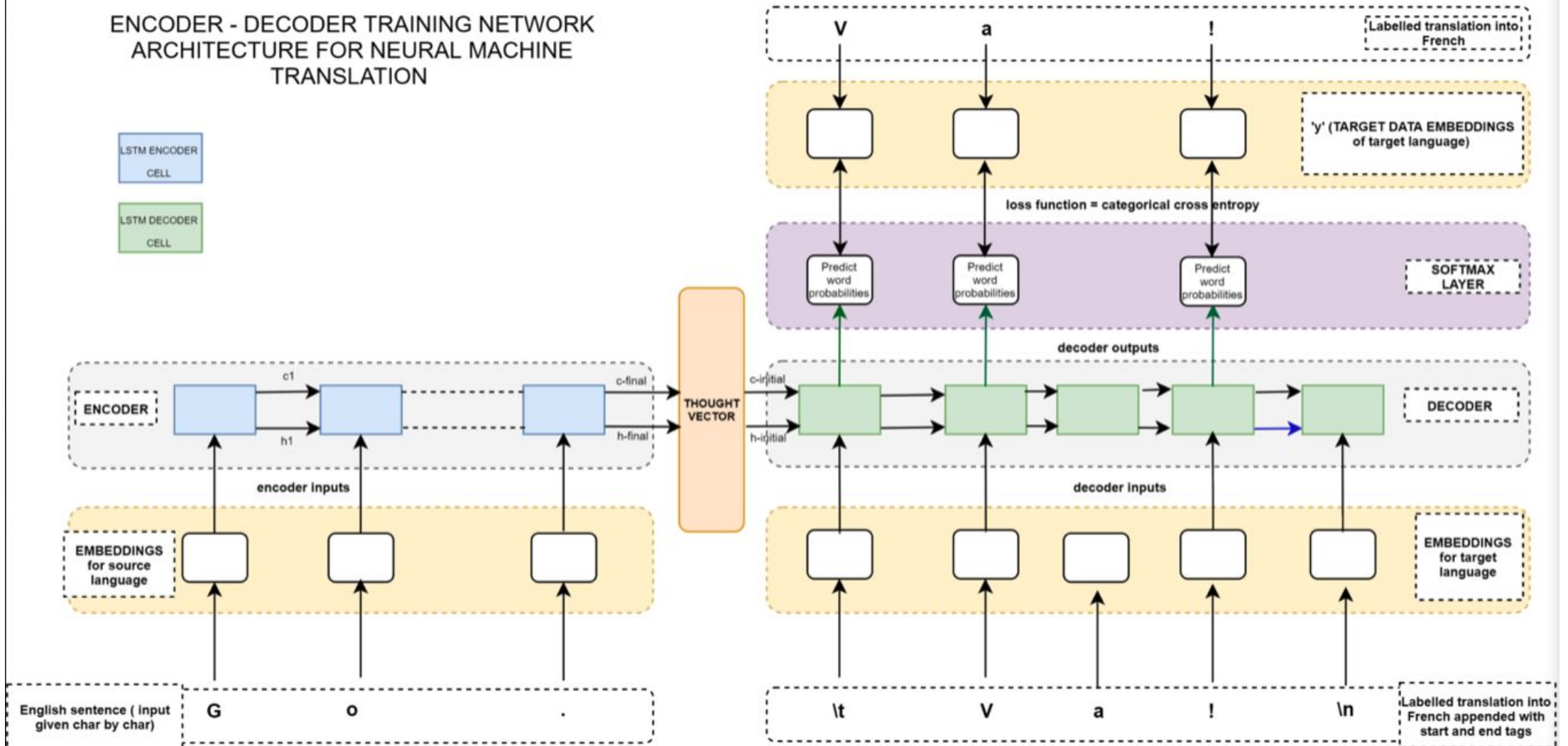


Training Algorithm – Sequence2sequence

ENCODER - DECODER TRAINING NETWORK
ARCHITECTURE FOR NEURAL MACHINE
TRANSLATION

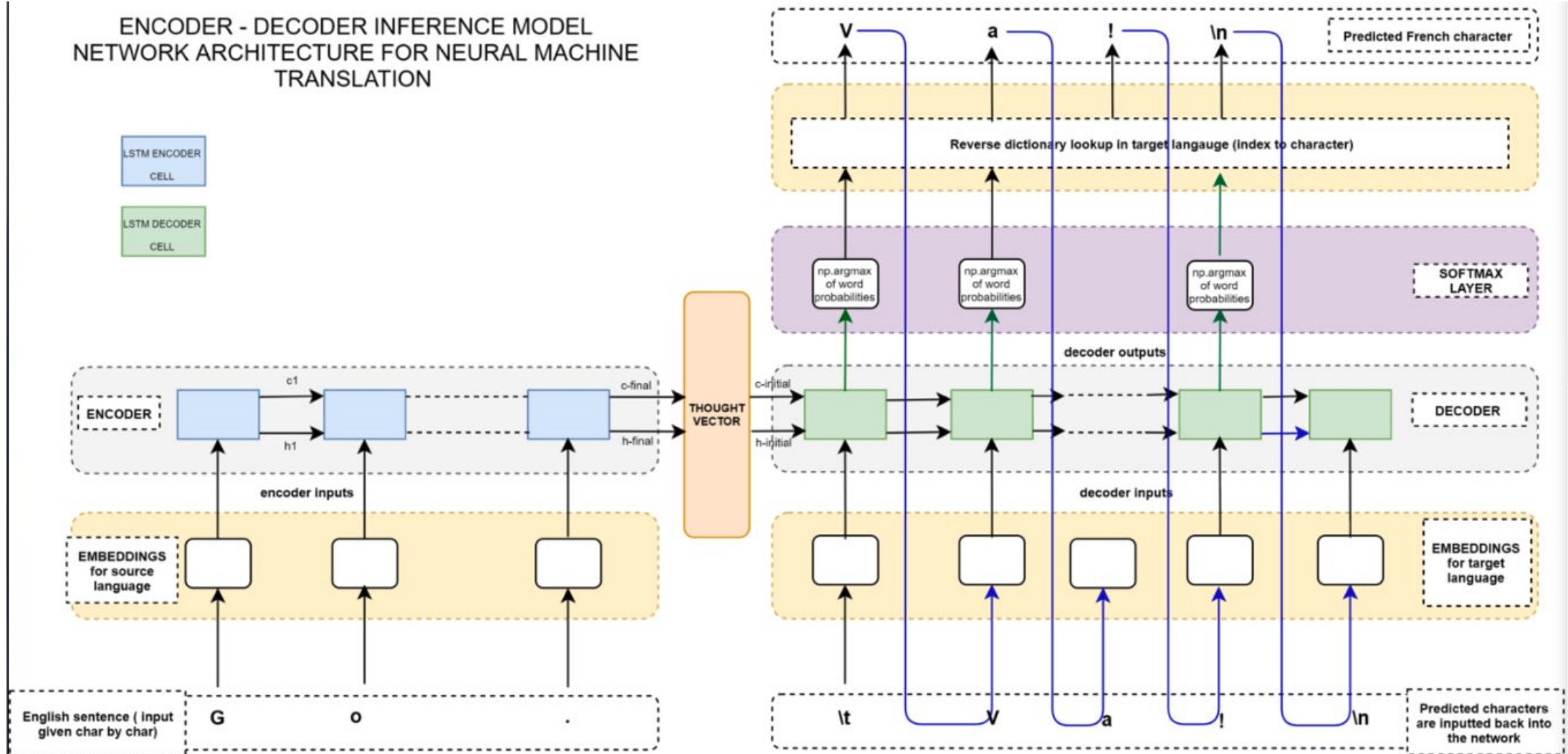
LSTM ENCODER
CELL

LSTM DECODER
CELL



Training Algorithm – Sequence2sequence

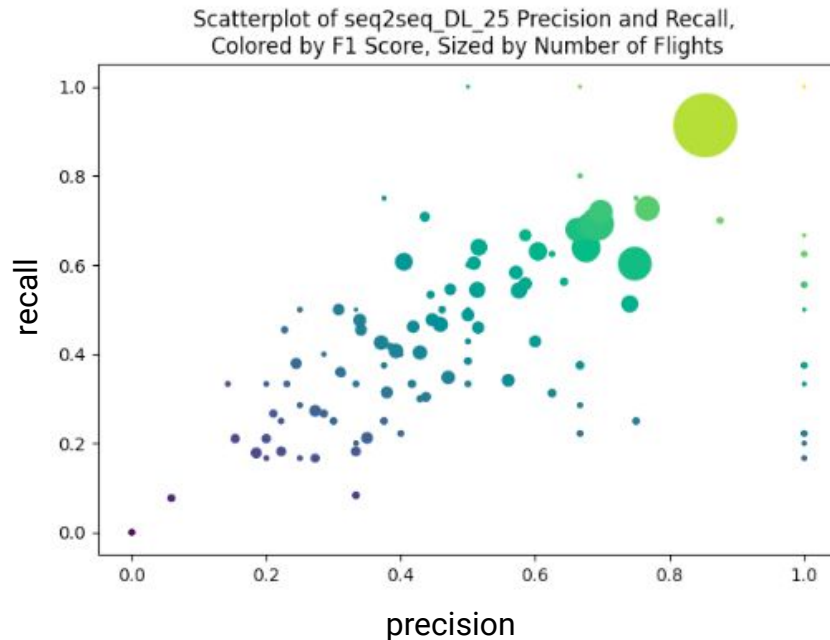
ENCODER - DECODER INFERENCE MODEL
NETWORK ARCHITECTURE FOR NEURAL MACHINE
TRANSLATION



Model Results - Sequence2sequence

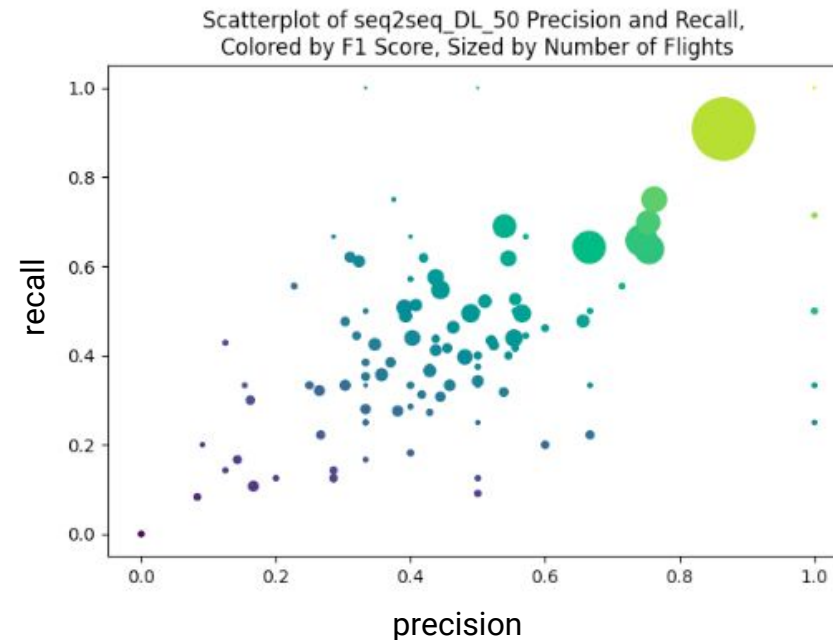
Sequence Length - 25

```
Accuracy: 0.6262  
Cohens Kappa: 0.5920248164166245  
Precision: 0.6318722285441556  
Recall: 0.6262  
F1: 0.6221704140365036
```



Sequence Length - 50

```
Accuracy: 0.613  
Cohens Kappa: 0.5793893933873405  
Precision: 0.6229796477216917  
Recall: 0.613  
F1: 0.611046099536135
```



Results

Model	Accuracy	Cohen's Kappa	F1	Sequence length
LSTM network	0.6264	0.5933	0.6235	50
seq2seq	0.6262	0.5920	0.6222	25
LSTM network	0.6242	0.5919	0.6223	25
Stacked LSTM	0.6222	0.5895	0.6209	50
seq2seq	0.6130	0.5794	0.6110	50
Baseline (pure random)	0.2444	0.0	0.09	50

Summary and Conclusions

- Best performing model was arguably the simplest (one-layer LSTM).
- Good data pipelining is an investment that pays off in complex projects. But it's ok to diverge!
- Our models may have performed best on Delta because Atlanta (ATL) is such a central hub.
- Thorough EDA is important to understand your data, but be prepared to dig deeper if needed.
- Longer sequences performed better (overall), but they also would be less responsive in a real-world environment that changed often.

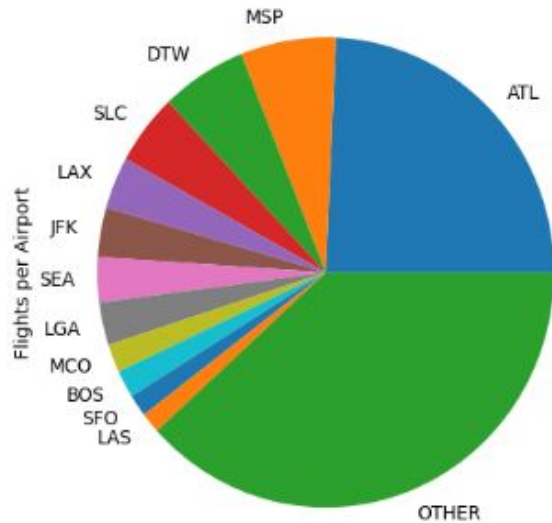
References

- https://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy_score.html?highlight=accuracy#sklearn.metrics.accuracy_score
- https://scikit-learn.org/stable/modules/generated/sklearn.metrics.cohen_kappa_score.html
- <https://towardsdatascience.com/neural-machine-translation-using-seq2seq-with-keras-c23540453c74>
- [Understanding LSTM Networks -- colah's blog](#)
- [Illustrated Guide to LSTM's and GRU's: A step by step explanation | by Michael Phi | Towards Data Science](#)

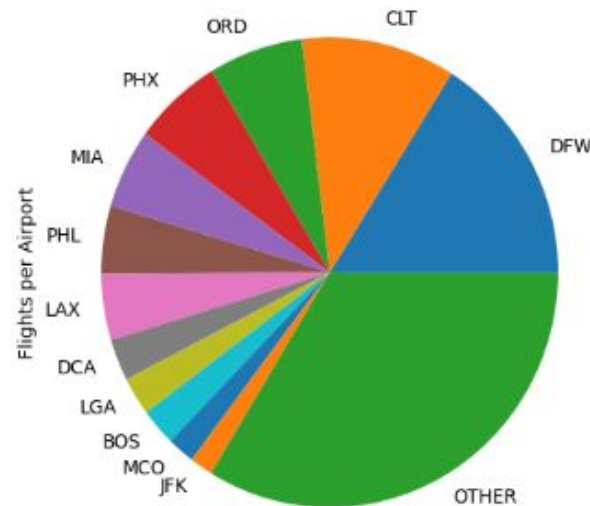
Questions?

Comparison of Airlines With Best Model

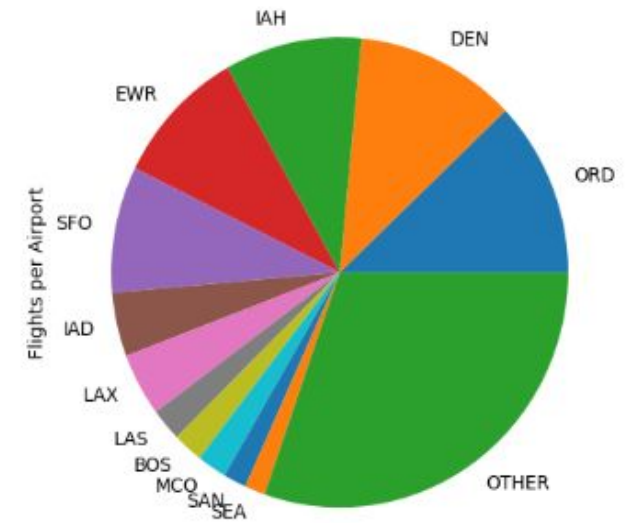
Percent of Flight From Top 12 Airports for DL



Percent of Flight From Top 12 Airports for AA



Percent of Flight From Top 12 Airports for UA



Delta Airlines with
Sequence Length = 25

Accuracy Score: 0.6264
Cohen's Kappa Score: 0.5933

American Airlines with
Sequence Length = 25

Accuracy Score: 0.57898
Cohen's Kappa Score: 0.5513

United Airlines with
Sequence Length = 25

Accuracy Score: 0.52076
Cohen's Kappa Score: 0.48737