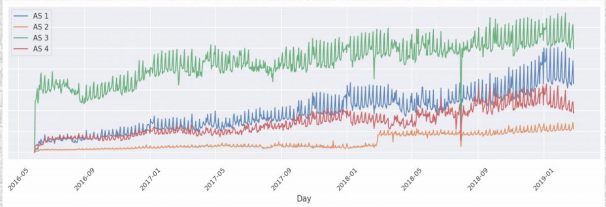
DeepFlow

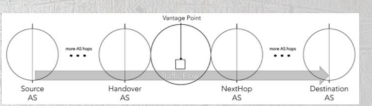
IP flow data prediction using deep learning techniques

Dr. Sebastian Fischer - seb1988@stanford.edu

Predicting network traffic is of high relevance for Internet Service Providers, because it enables them to prepare for upcoming peak network utilizations better. This project illustrates how recurrent neural networks, specifically Long Short-Term Memory models, outperform classical autoregressive models in forecasting IP flow data.

PREDICTING





time	AS 1	AS 2	AS 3	AS 4	Total
2019-02-25 19:00:00	e.g. 1.473	...	...	...	...
2019-02-25 20:00:00	...	...	...	...	...

DATA & FEATURES

ARIMA stands for "autoregressive integrated moving average" and is a statistical model to forecast time series data. I used the Python library PMDARIMA to run the auto_arima function that finds the optimal hyperparameters. This model serves as a baseline.

ARIMA (Baseline)

```
Layer (type)                 Output Shape              Param#
=================================================================
lstm_239 (LSTM)              (None, 200)               161600
dense_367 (Dense)            (None, 100)               20100
dense_368 (Dense)            (None, 7)                 707
```

Basic LSTM

```
Layer (type)                 Output Shape              Param#
=================================================================
lstm_211 (LSTM)              (None, 200)               161600
repeat_vector_84 (RepeatVect (None, 7, 200)            0
lstm_212 (LSTM)              (None, 7, 200)            320800
time_distributed_167 (TimeDi (None, 7, 100)            20100
time_distributed_168 (TimeDi (None, 7, 1)              101
```

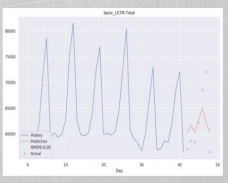
Encoder-Decoder LSTM

```
Layer (type)                 Output Shape              Param#
=================================================================
convld_93 (Convl D)          (None, 22, 64)            256
convld_94 (Convl D)          (None, 20, 64)            12352
max_poolingld_47 (MaxPooling (None, 10, 64)            0
flatten_47 (Flatten)         (None, 640)               0
repeat_vector_96 (RepeatVect (None, 24, 640)           0
lstm_231 (LSTM)              (None, 24, 200)           672800
time_distributed_191 (TimeDi (None, 24, 100)           20100
time_distributed_192 (TimeDi (None, 24, 1)             101
```

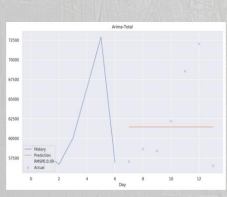
CNN-Encoder-Decoder LSTM

MODELS

AS	Unit	best	Model	History	Forecast
AS 1	Days	9 %	basic_LST	49	7
AS 1	Days	12 %	Arima	7	7
AS 1	Hours	47 %	cnnLSTM	120	24
AS 1	Hours	152 %	Arima	48	24
AS 2	Days	9 %	cnnLSTM	49	7
AS 2	Days	10 %	Arima	28	7
AS 2	Hours	28 %	basic_LST	168	24
AS 2	Hours	47 %	Arima	24	24
AS 3	Days	6 %	e_d_lstm	35	7
AS 3	Days	8 %	Arima	7	7
AS 3	Hours	35 %	basic_LST	168	24
AS 3	Hours	45 %	Arima	48	24
AS 4	Days	8 %	basic_LST	49	7
AS 4	Days	10 %	Arima	7	7
AS 4	Hours	26 %	basic_LST	168	24
AS 4	Hours	55 %	Arima	24	24
Total	Days	8 %	basic_LST	49	7
Total	Days	9 %	Arima	7	7
Total	Hours	35 %	cnnLSTM	168	24
Total	Hours	60 %	Arima	72	24



LSTM



ARIMA

Results

The report has shown that deep learning approaches outperform a statistical model baseline. It became clear that even with a very simple univariate input and without a systematic hyperparameter search nor any data preprocessing or feature engineering reasonably high performance measures were possible. Although the RMSPE values suggest a good prediction performance, one would argue that the prediction performance should still be enhanced further when looking (qualitatively) at the forecasting graphs. This shall be achieved with a more systematic hyperparameter search and the exploration of further model architectures.

DISCUSSION & FUTURE

Bates, T., Gerich, E., Joncheray, L., J-M. Jouanigot, Karrenberg, D., Terpstra, M., & Yu, J. (1995). Representation of IP Routing Policies in a Routing Registry (ripe-81++) (No. RFC1786). RFC Editor. https://doi.org/10.17487/rfc1786; Brownlee, J. (2016, December 18). How to Backtest Machine Learning Models for Time Series Forecasting. Retrieved March 17, 2019, from https://machinelearningmastery.com/backtest-machine-learning-models-time-series-forecasting/ ; Brownlee, J. (2018a, August 5). 11 Classical Time Series Forecasting Methods in Python (Cheat Sheet). Retrieved March 17, 2019, from https://machinelearningmastery.com/time-series-forecasting-methods-in-python-cheat-sheet/; Brownlee, J. (2018b, October 10). How to Develop LSTM Models for Multi-Step Time Series Forecasting of Household Power Consumption. (n.d.). Retrieved March 17, 2019, from https://machinelearningmastery.com/how-to-develop-lstm-models-for-multi-step-time-series-forecasting-of-household-power-consumption/ ; Gers, F. A., Eck, D., & Schmidhuber, J. (2001). Applying LSTM to Time Series Predictable through Time-Window Approaches. In G. Dorffner, H. Bischof, & K. Hornik (Eds.), Artificial Neural Networks — ICANN 2001 (pp. 669–676). Springer Berlin Heidelberg. ; Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. Neural Computation, 9(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735 ; Lakhina, A., Papagiannaki, K., Crovella, M., Diot, C., Kolaczyk, E. D., & Taft, N. (2004). Structural analysis of network traffic flows. In Proceedings of the joint international conference on Measurement and modeling of computer systems - SIGMETRICS 2004/PERFORMANCE 2004 (p. 61). New York, NY, USA: ACM Press. https://doi.org/10.1145/1005686.1005697 ; Nau, Robert. (n.d.). Introduction to ARIMA. Retrieved from https://people.duke.edu/~rnau/411arim.htm ; Papagiannaki, K., Taft, N., Zhang, Z.-L., & Diot, C. (2005). Long-Term Forecasting of Internet Backbone Traffic. IEEE ; Transactions on Neural Networks, 16(5), 1110–1124. https://doi.org/10.1109/TNN.2005.853437 ; Perera, S. (2016, June 3). Rolling Window Regression: a Simple Approach for Time Series Next value Predictions. Retrieved March 17, 2019, from https://medium.com/making-sense-of-data/time-series-next-value-prediction-using-regression-over-a-rolling-window-228f0acae363 ; Portilla, J. M. (2018, March 26). Using Python and Auto ARIMA to Forecast Seasonal Time Series. Retrieved March 17, 2019, from https://medium.com/@josemarcialportilla/using-python-and-auto-arima-to-forecast-seasonal-time-series-90877adff03c ; Schmidhuber, J. (2015). Deep learning in neural networks: An overview. Neural Networks, 61, 85–117. https://doi.org/10.1016/j.neunet.2014.09.003 ;Sperotto, A., Schaffrath, G., Sadre, R., Morariu, C., Pras, A., & Stiller, B. (2010). An Overview of IP Flow-Based Intrusion Detection. IEEE Communications Surveys & Tutorials, 12(3), 343–356. https://doi.org/10.1109/SURV.2010.032210.00054 ;Zseby, T., Quittek, J., Claise, B., & Zander, S. (2004). Requirements for IP Flow Information Export (IPFIX). Retrieved March 17, 2019, from https://tools.ietf.org/html/rfc3917; Used libraries,

KERAS, MATPLOTLIB, NUMPY, PANDAS, SCIKIT-LEARN, SCIPY, SEABORN, PMDARIMA, STATSMODELS, TENSORFLOW

REFERENCES