9th International Workshop on Climate Informatics
October 02-04, 2019
Hosted by École Normale Supérieure, Paris, France

FORECASTING MAXIMA IN CLIMATE TIME SERIES

Israel Goytom[1,2,3] Kris Sankaran[1,3]

***Abstract*—Climate change is already altering the probabilities of weather hazards. Accurate prediction of climate extremes can inform effective preparation against weather-induced stresses. Accurately forecasting extreme weather events is a task that has attracted interest for many years. Classical and to a lesser extent, machine learning-based approaches have handled this issue; however, such systems are hard to tune or scale. While the prediction of extremes has been the subject of investigation across several communities, including meteorology, machine learning, and statistics, it has been subject to far less scrutiny than the prediction of conditional means. In this work, we offer a systematic comparison of existing approaches on prediction of maximum temperature. Further, motivated by this comparison, we propose a method to forecast maxima temperature in weather time series that unifies deep learning with extreme value theory.**

I. MOTIVATION

Weather has an enormous impact in our daily lives. When weather forecasting is effective, we know that burdensome weather events are not coming tomorrow or the day after, either. Extreme weather events such as hurricanes, tornadoes, heavy downpours, heat waves, and droughts affect all sectors of the economy and the environment, impacting people where they live and work (1). According to EM-DAT (International Disaster Database), more than 60 million people were affected only in the year 2018 alone by extreme events. – Forecasting the occurrence of extreme events in time series has attracted interest of researchers for many years (2; 3; 4). Forecasting maxima in weather time series data is essential for extreme weather events, i.e., anticipating high temperature will help people to prepare in advance, forecasting high precipitation might help with flooding events hazards, high-wind speeds with protecting infrastructure. Forecasting maximum surface temperature will help to foretell extreme rainfall, which is the main factor generating floods, landslides, and soil erosion and thus can cause environmental, societal, and economical damages (5).

Forecasting these sources of stress hinges on being able to forecast extremes accurately, and while this problem has been viewed from several angles in the machine learning community, including quantile regression and extreme value forecasting, there have been no systematic comparisons. This work provides a common evaluation of three alternative approaches – direct prediction using an LSTM, a probabilistic LSTM with a likelihood common in extreme value theory, and a quantile regression technique on daily temperature forecasting problem.

Predicting extreme events such as peak wind (6), traffic, (7) and electricity demand (8) has become a common task in both statistics and machine learning community. In statistics, there is a branch known as extreme value theory (9).

Classical methods for extreme weather events forecasting mostly treat the problem as a full-time series prediction problem (9; 10). Alternatively, methods have been developed to model quantiles specifically, including quantile regression and quantile regression forests (11), though these are rarely applied to extreme values. Classical models require hard tuning for the parameters. Long Short Term Memory (LSTM) (12) based forecasting gained popularity due to its end-to-end modeling, automatic feature extraction abilities, and capacity to learn complex interactions.

A combination of classical time series models and machine learning methods have been used to predicting special events (13; 14). Deep convolutional neural networks based classifiers have been used to detect extreme weather (15). (16) proposed a multichannel spatiotemporal encode-decoder convolutional neural network architecture for semi-supervised bounding box prediction in large climate datasets. Recurrent neural networks (RNNs), especially LSTMs, have been used for precipitation nowcasting (17) – when trained on two-dimensional radar map time series, their system

Corresponding author: Israel Goytom, isrugeek@gmail.com
[1]Mila, Montreal, Canada. [2]Ningbo University, Faculty of Science, Ningbo, 352100, China [3]Université de Montréal, Department of Informatics and Operations Research, Montreal, Canada.

is able to outperform the current state-of-art precipitation nowcasting system on various evaluation metrics. Recently, (18) developed an end-to-end forecast model for multi-step time series forecasting that can handle multivariate inputs for extreme events, applying their system to peak travel prediction.

The questions discussed in this paper are:

- To forecast extreme values of the time series, does it help to account for the heavy-tailed distributions expected to arise according to classical statistics theory, or are modern deep learning or quantile regression methods sufficient as they are?
- Alternatively, is there some way to combine the classical theory with modern machine learning in a way that gets the best of both worlds?

Answering these questions will help both the machine learning community, by giving insight into where to invest research effort, and the climate modeling community, as it suggests best practices in a problem of practical importance. We are unaware of any deep learning based methods for climate extreme values or maxima forecasting in weather time series.

The main contributions of this paper are:

- We provide benchmark experiments of modern deep learning, the proposed probabilistic LSTM, and quantile random forest, to evaluate their relative merits on shared tasks.
- We propose an LSTM model with Gumbel-distributed errors, as one way to combine classical theory of extreme values with modern deep learning.

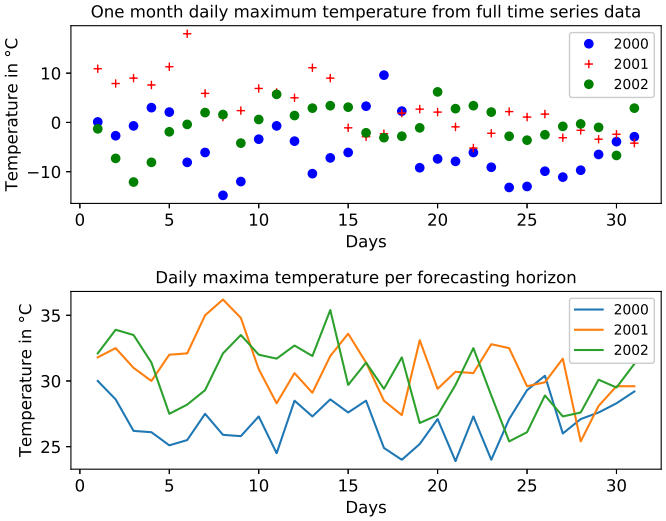


Fig. 1: Extracting maxima values from full time series data. We extract only the maxima from the full time series over the forecasting horizon. This figure is a sample of daily maximum temperature for three years of weather data.

II. METHODS

We consider three models to forecast maxima in weather datasets: LSTM, LSTM with a Gumbel likelihood, and quantile random forest.

A. *Long Short-Term Memory (LSTM)*

Long Short-Term Memory (LSTM) is a type of RNN, capable of learning long-term dependecies that was designed by Hochreiter et al. (12) to address the vanishing and exploding gradient problems of conventional RNNs. The LSTM model discussed in this paper is based on the the original LSTM paper (12) with a hidden layer of LSTM units and an output layer used to make predictions. We provide multivariate data as input and forecast the output maxima. Univariate time-series approaches directly model the temporal domain, they suffer from a frequent retraining requirement (19). Hence, we choose multivariate input, allowing the model to learn from multiple features, not only from the feature being forecasted.

$$
\begin{aligned}
i_t &= \sigma(x_t U^i + h_{t-1} W^i) \\
f_t &= \sigma(x_t U^f + h_{t-1} W^f) \\
o_t &= \sigma(x_t U^p + h_{t-1} W^o) \\
\tilde{C}_t &= \tanh(W x_t U^g + h_{t-1} W^g) \\
C_t &= \sigma(f_t * C_{t-1} + i_t * \tilde{C}_t) \\
h_t &= \tanh(C_t) * o_t.
\end{aligned}
\tag{1}
$$

In Equation 1 i is input gate, f is forget gate and o is output gate. W is the recurrent connection at the previous and current hidden layer while U is the weight matrix connecting the inputs to the current hidden layer. $\tilde{C}$ is a candidate hidden state that is computed based on the current input and the previous hidden state. C is the internal memory of the unit. The output hidden state h_t is computed by multiplying the memory with the output gate as shown in Equation 1.

B. *LSTM + Gumbel-Markov model*

In our next approach we add a Gumbel distribution and Markov stochastic model to the LSTM model. The Markov model is based on (20) and discussed in (21). The cumulative distribution function (CDF) and probability density function (PDF) for the Gumbel

distribution are given in Equation 2 and Equation 3 respectively.

$$F(x;\mu,\beta) = \exp\left(-\exp\left(-\frac{x-\mu}{\beta}\right)\right) \quad (2)$$

$$f(x) = \frac{1}{\beta}\exp\left(\frac{x-\mu}{\beta}\right)\exp\left(-\exp\left(\frac{x-\mu}{\beta}\right)\right) \quad (3)$$

The mode is μ, while the median is $\mu - \beta\ln(\ln 2)$, and the mean is given by : $E(X) = \mu + \gamma\beta$ where $\gamma \approx 0.5772$ is the Euler-Mascheroni constant.

At the mode, where $x = \mu$, the value of $F(x;\mu,\beta)$ becomes $e^{-1} \approx 0.37$ for whatever the value of β.

We use a Markov model to describe the sequence of possible extremes, requiring the distribution of the extreme value to depend only on the state attained at the previous time point,

$$\Pr(X_{n+1} = x \mid X_n = y) = \Pr(X_n = x \mid X_{n-1} = y)$$

For the standard Gumbel distribution where $\mu = 0$ and $\beta = 1$, then CDF states in Equation 2 will be $F(x) = e^{-e^{(-x)}}$ and the PDF states in Equation 3 will be $f(x) = e^{-x}e^{-e^{x}}$.

We optimize the Gumbel likelihood over the features learned by LSTM. The negative log-pdf of a Gumbel distribution (parameterized by μ and β) is

$$-\log p(x_t;\mu,\beta) = \log\beta - \frac{x_t-\mu}{\beta} + \exp\left(-\frac{x_t-\mu}{\beta}\right)$$

Here, we parameterize the mode μ by learned features h_t from the LSTM at the same timepoint. We consider them a linear function of those features, i.e. $\mu(h_t) = w^T h_t$. If we force $\beta = 1$, then this expression becomes

$$-\log p(x_t|h_t;w) = -x_t + w^T h_t + \exp\left(-\left(x_t - w^T h_t\right)\right)$$

For an LSTM the representations $h_t = f_\theta(x_{t-\Delta},\ldots,x_t)$, parameterized by θ, must be learned, along with the Gumbel parameter w. We approach this using maximum likelihood. Specifically, if $x_i := \left(x_{i(t-\Delta)},\ldots,x_{it}\right)$ is the i^{th} window, we minimize

$$-\sum_{i,t}\log p(x_{it}|h_{it};w)$$
$$=\sum_{i,t} -x_{it} + w^T h_{it} + \exp\left(-\left(x_{it} - w^T h_{it}\right)\right).$$

We take a minibatch of x_i and backpropagate through this loss, updating our estimates for θ and w based on the gradient.

C. Quantile Random Forest

Quantile random forests are a variant of random forests that maintain the empirical distribution of all points at leaves in every component tree, as opposed to taking the mean in every leaf, as in standard random forests. This allows the model to provide estimates of arbitrary quantiles at any input x. This is in contrast with standard quantile regression methods – including those based on deep learning – which learn to target specific quantiles by optimizing an asymmetric absolute error loss.

Specifically, to estimate the α-quantile at a position x, the method proceeds as follows. First, grow a collection of trees according to the split criterion in standard random forests. For the t^{th} tree, define the weight, $w_i(x) = \frac{1}{|L_t(x_i)|}$ if x is in the same leaf as x_i in the t^{th} tree, and 0 otherwise, where $|L_t(x_i)|$ is the number of observations in that leaf of the t^{th} tree. That is, observations x_i far from x shouldn't get any weight, and large leaves should be downweighted. Finally, average the weights across trees into a single $w_i(x)$, and use them to estimate the distribution function, $\hat{F}(y|x) = \sum w_i(x)\,\mathbf{1}\{y_i \le y\}$. From this distribution function, any quantile can be extracted.

D. Data

The dataset[1] considered in this work is based on Environment and Climate Change Canada data, the dataset has 148 years of recorded data with 68 features.

Maximum temperature is extracted from the daily temperature during the forecast period Figure 1. We use pushforwards imputation to fill missing values and interpolate values onto evenly spaced timepoints. We prepared the data as maxima for the extreme value which needs to be forecasted and their representative multivariate inputs. An example of a raw dataset is shown in Figure 2 (top). We prepared the training dataset by splitting the raw data into sliding windows (Figure 2, bottom). The input x_i includes the 30 most recent observations, and y_i are the maxima over the next two weeks. We used temporal cross-validation to evaluate in sliding windows.

E. Experiments

The network was trained and tested using NVIDIA Tesla K80 GPU, leveraging with NVIDIA CUDA Toolkit (22). We use the pytorch (23) library for implementation. The code is publicly available[2].

[1] https://montreal.weatherstats.ca/
[2] https://github.com/isrugeek/climate_extreme_values

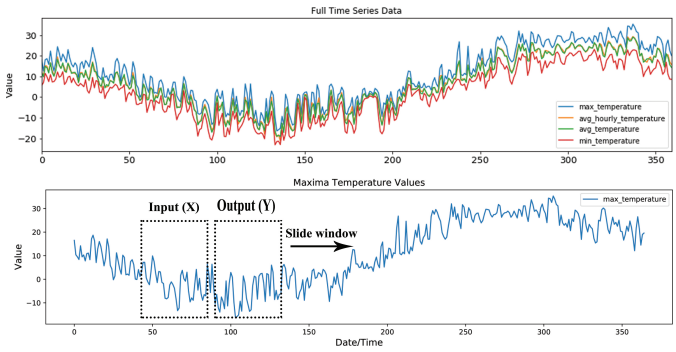
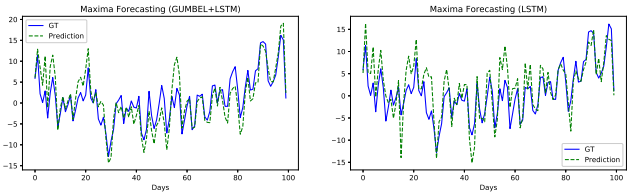


Fig. 2: This figure displays sample maxima from the based on Environment and Climate Change Canada data. Top: Sample input to our model. Bottom: Description of sample creation. We create two sliding windows, one for x_{it} and another for y_{it}.

III. EVALUATION

In this section, we present results from the methods we discussed and benchmark them relative to the ground truth. We understood that that LSTM method discussed at the page subsection II-A with more datapoints has improved accuracy, missing points between datapoints will strongly affect the performance of subsection II-A. We notice that interpolation as pre-processing and quantiles in the sample improves the performance. Results from all methods are shown in Table I, and the forecasting results are shown in Figure 3.



(a) (b)

Fig. 3: Ground truth (GT) and prediction comparison of two models on the Canada weather dataset.

We calculate mean absolute error (MAE) see Equation 4 to measure the errors in a set of predictions, when n is number of samples, y actual value (GT) and $\hat{y}$ is the predicted value.

$$MAE = \frac{1}{n}\sum_{t=1}^{n} |y_j - \hat{y}_j| \tag{4}$$

TABLE I: Forecasting error for LSTM, LSTM + Gumbel Markov and quantile random forest model on two different datasets.

METHOD	MAE	RMSE
LSTM	0.3828± 0.0664	0.3075± 0.004
LSTM + GM	**0.3252± 0.0087**	**0.3072± 0.009**
QRF	0.50±0.04	0.411±0.06

IV. DISCUSSION AND FUTURE WORK

In this paper, we have contrasted existing machine learning and statistical approaches to extreme value modeling, and then we proposed a way to combine the perspectives. We have also evaluated the three models performance in forecasting maximal values on public weather dataset. From our experience (a) LSTM has more trouble on heavy-tailed distributions than the Gumbel-Markov model, (b) directly predicting the maximum of distribution does better than producing a forecast and extracting the maximum from that forecast, (c) a combination of the LSTM + Gumbel-Markov model outperforms LSTM or quantile random forest methods with respect to sample complexity and MAE, and the improvement can be traced to the LSTM's high flexibility and the Gumbel model's ability to deal with heavy tails. In the future, we hope to extend these ideas to the classification of weather extremes, and we will study the effectiveness of our approach on other quantiles and at different time horizons in an extended version of this paper.

REFERENCES

[1] R. R. Heim, "An overview of weather and climate extremes products and trends," vol. 10, pp. 1–9.

[2] E. J. Kendon, N. M. Roberts, H. J. Fowler, M. J. Roberts, S. C. Chan, and C. A. Senior, "Heavier summer downpours with climate change revealed by weather forecast resolution model," vol. 4, no. 7, pp. 570–576.

[3] A. G. Barnston and S. J. Mason, "Evaluation of iris seasonal climate forecasts for the extreme 15% tails," *Weather and Forecasting*, vol. 26, no. 4, pp. 545–554, 2011.

[4] R. Schnur, "The investment forecast," *Nature*, vol. 415, no. 6871, pp. 483–484, 2002.

[5] G. Panthou, A. Mailhot, E. Laurence, and G. Talbot, "Relationship between surface temperature and extreme rainfalls: A multi-time-scale and

event-based analysis," *Journal of Hydrometeorology*, vol. 15, no. 5, pp. 1999–2011, 2014.

[6] P. Friederichs and T. L. Thorarinsdottir, "Forecast verification for extreme value distributions with an application to probabilistic peak wind prediction," *Environmetrics*, vol. 23, no. 7, pp. 579–594, 2012.

[7] N. Polson and V. Sokolov, "Deep learning for short-term traffic flow prediction," vol. 79, pp. 1–17.

[8] J. Ringwood, D. Bofelli, and F. T. Murray, "Forecasting electricity demand on short, medium and long time scales using neural networks," *Journal of Intelligent and Robotic Systems*, vol. 31, pp. 129–147, 05 2001.

[9] L. d. Haan and A. Ferreira, *Extreme value theory: an introduction*. Springer series in operations research, Springer. OCLC: ocm70173287.

[10] R. W. Katz and B. G. Brown, "Extreme events in a changing climate: Variability is more important than averages," vol. 21, no. 3, pp. 289–302.

[11] N. Meinshausen, "Quantile regression forests," *J. Mach. Learn. Res.*, vol. 7, pp. 983–999, Dec. 2006.

[12] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, pp. 1735–1780, Nov. 1997.

[13] R. J. Hyndman and Y. Khandakar, "Automatic time series forecasting: the forecast package for R," *Journal of Statistical Software*, vol. 26, no. 3, pp. 1–22, 2008.

[14] T. Opitz, "Modeling asymptotically independent spatial extremes based on Laplace random fields," *arXiv e-prints*, p. arXiv:1507.02537, Jul 2015.

[15] Y. Liu, E. Racah, Prabhat, J. Correa, A. Khosrowshahi, D. Lavers, K. Kunkel, M. Wehner, and W. Collins, "Application of Deep Convolutional Neural Networks for Detecting Extreme Weather in Climate Datasets," *arXiv e-prints*, p. arXiv:1605.01156, May 2016.

[16] E. Racah, C. Beckham, T. Maharaj, Prabhat, and C. J. Pal, "Semi-supervised detection of extreme weather events in large climate datasets," *CoRR*, vol. abs/1612.02095, 2017.

[17] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W. chun Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," in *NIPS*, 2015.

[18] N. Laptev, J. Yosinski, L. E. Li, and S. Smyl, "Time-series extreme event forecasting with neural networks at uber," p. 5.

[19] L. Ye and E. Keogh, "Time series shapelets: A new primitive for data mining," in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, (New York, NY, USA), pp. 947–956, ACM, 2009.

[20] R. G. Krishnan, U. Shalit, and D. Sontag, "Structured inference networks for nonlinear state space models," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, pp. 2101–2109, AAAI Press, 2017.

[21] E. Bingham, J. P. Chen, M. Jankowiak, F. Obermeyer, N. Pradhan, T. Karaletsos, R. Singh, P. Szerlip, P. Horsfall, and N. D. Goodman, "Pyro: Deep Universal Probabilistic Programming," *arXiv preprint arXiv:1810.09538*, 2018.

[22] J. Nickolls, I. Buck, M. Garland, and K. Skadron, "Scalable parallel programming with cuda," *Queue*, vol. 6, pp. 40–53, Mar. 2008.

[23] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017.