Improving medium-range ensemble forecasts with transformers

Jonathan Weyn (Microsoft, ECMWF)
Zied Ben Bouallegue, Matthew Chantry, Peter Deuben, Jesper Dramsch (ECMWF)

Thanks: Tobias Finn (Univ. of Hamburg, MPI)
Can deep learning provide *global, high-resolution, full-ensemble* calibrated medium-range weather forecasts?

- Ensemble numerical weather forecasts still benefit from statistical post-processing
- Methods such as Ensemble Model Output Statistics (EMOS) are rigid, and generally restricted to predicting distribution parameters
- Machine learning methods have successfully been applied to improve predictions of ensemble distribution parameters (usually mean and std)
- Recent introduction of a self-attentive ensemble transformer offers to provide full ensemble distributions of calibrated forecasts
Relevant work

• Rasp and Lerch, “Neural Networks for Postprocessing Ensemble Weather Forecasts” (2018)
  o Neural networks trained to predict mean and std of an ensemble forecast
  o Non-linear NN is the best model at most validation stations across Germany
Relevant work

• Gronqvist et al., “Deep Learning for Post-Processing Ensemble Weather Forecasts”, 2020
  o U-net residual NN architecture applied on global 0.5-degree fields from ECMWF hindcasts
  o Models for bias correction (mean) and uncertainty quantification (std)
  o Able to exceed performance of 10-member hindcast with 5-member input
The ensemble transformer

- Finn, “Self-Attentive Ensemble Transformer”, 2021
  - member-by-member approach that applies corrections based on interactions with all other members
  - maintains spatial correlations
  - results for 3-day forecasts of 2-m temperature show 20% improvement in CRPS and spread-skill ratio near 1
Transformer 101

- Developed for natural language processing (NLP) tasks, such as translation
- Unlike recurrent neural networks, computes “attention,” or relevance, of each sequence element to every other element
Transformer 101: Self-attention

- Compute a *query*, *key*, and *value* by multiplying each sequential input by arrays of trainable weights
  - key and query initialized as 0 in training
- Compute the similarity (dot-product) between the *query* of each input to the *key* of all inputs
- Add to the *value* of each input the sum of all inputs’ *values* weighted by the normalized similarity
The ensemble transformer

- Apply the transformer along the ensemble member dimension
- **Values** represent an embedding of the original ensemble member
- **Keys** and **queries** are used to compute similarity between ensemble members on a global scale
PoET: a transformer U-net
Training framework

- Training data consists of 11-member (10 perturbed + 1 control) hindcasts from 2000-2016
- Methodology is agnostic to the number of ensemble members, so testing is done on the full 51-member (50 perturbed + 1 control) operational IFS twice weekly in 2021
- Selection of either 3 or 6 variables, both including prescribed fields
  - 2-m temp, Z500, T850, land-sea mask, topography, TOA insolation, $U_{700}$, $V_{700}$, total cloud cover
- Each training example contains the full 11-member hindcast for a single lead time, with lead times every 6 h up to 96 h
- Target analysis data is ERA5 reanalysis 2-m temperature; loss function is CRPS
Benchmark

• Member-by-member (MBM) sophisticated linear post-processing approach (Schaeybroeck and Vannitsem, 2015)

• Maximum likelihood estimation of 3 parameters
  o ensemble mean scaling & nudging
  o ensemble spread scaling

• Constrained to preserve weak ensemble reliability (spread = skill) and climatological reliability (forecast variability = observation variability)
## PoET vs MBM

<table>
<thead>
<tr>
<th></th>
<th>PoET</th>
<th>MBM benchmark</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Inputs</strong></td>
<td>multiple variables</td>
<td>only target variable</td>
</tr>
<tr>
<td><strong>Training period</strong></td>
<td>2000-2016</td>
<td>2000-2019</td>
</tr>
<tr>
<td><strong>Training lead times</strong></td>
<td>all from 6 to 96</td>
<td>one lead time per model</td>
</tr>
<tr>
<td><strong>Training window</strong></td>
<td>all</td>
<td>+/- 15 days from init</td>
</tr>
<tr>
<td><strong>Training ensemble members</strong></td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td><strong>Constraints</strong></td>
<td>none</td>
<td>reliability</td>
</tr>
<tr>
<td><strong>Spatial context</strong></td>
<td>aware</td>
<td>per-grid-point (rank retained)</td>
</tr>
</tbody>
</table>
Scores

**CRPS**

2m temperature
Continuous ranked probability score
20210101 00z to 20211201 12z
Global

<table>
<thead>
<tr>
<th>Better</th>
<th>0.5</th>
<th>1.0</th>
<th>1.5</th>
<th>2.0</th>
<th>2.5</th>
<th>3.0</th>
<th>3.5</th>
<th>4.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>raw ENS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>benchmark</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3vlead96</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5var+tcc</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**CRPS SCORE**

2m temperature
Continuous ranked probability score
20210101 00z to 20211201 12z
Global

<table>
<thead>
<tr>
<th>Better</th>
<th>0.00</th>
<th>0.05</th>
<th>0.10</th>
<th>0.15</th>
<th>0.20</th>
</tr>
</thead>
<tbody>
<tr>
<td>raw ENS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>benchmark</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3vlead96</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5var+tcc</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Scores

**SPREAD/SKILL**

![Graph showing spread/skill for 2m temperature spread and skill from 20210101 00z to 20211201 12z, comparing different models like raw ENS, benchmark, 3vlead96, and 5var+tcc.](image1)

**RELIABILITY (SPREAD/SKILL)**

![Graph showing reliability for 2m temperature spread and skill from 20210101 00z to 20211201 12z, comparing different models like raw ENS, benchmark, 3vlead96, and 5var+tcc.](image2)
Difference matrix: Raw vs PoET ensemble members

Inner panels are the difference between forecast from PoET-calibrated ensemble member and raw IFS ensemble member. Dominant feature is a mean bias correction applied in all members.
Difference matrix: Raw vs PoET ensemble members

Inner panels are the difference between forecast anomalies relative to ensemble mean from PoET-calibrated ensemble member and raw IFS ensemble member.
PoET preserves inter-member calibration

This figure computes the MSE between each pair of raw and PoET-calibrated ensemble members.

The diagonal elements are lower, indicating that PoET members still resemble their original member more closely.

The control member is also more similar to all other members than other members are to each other.
Further work

• Expand input data to include more variables and lead times, and predict more output variables of interest, such as precipitation
  o preliminary results on precipitation show modest improvement with PoET

• Further optimize architecture to allow for inference on full-resolution IFS fields
PoET (Post-processing of Ensembles with Transformers) effectively improves performance of the operational IFS ensemble while improving spread/skill ratio.

The ensemble transformer can run on the full operational IFS ensemble while only being trained on an 11-member hindcast, yet still preserving inter-member calibration.

The full distribution of the PoET-calibrated ensemble can be used in further ensemble analysis.

Take-aways