ISSUE 42 MAY 2024

GAṆANAM (गणनम्)

HIGH PERFORMANCE *COMPUTING* NEWSLETTER
INSTITUTE FOR PLASMA RESEARCH, INDIA

INSIDE THIS ISSUE (3 Pages)

Hindi Speech Recognition using Attention Mechanism

Gaurav Garg (Scientific Officer - G, MRD, IPR)
Email: gauravg@ipr.res.in

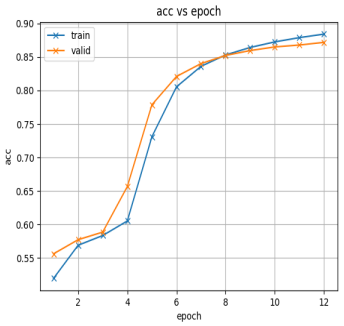


Figure 1: Accuracy for Train and Validation data set

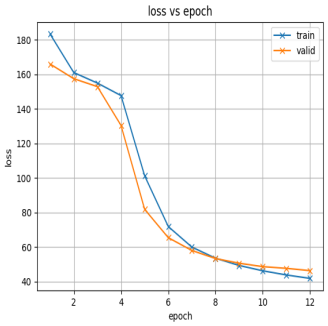


Figure 2: Loss for Train and Validation data set

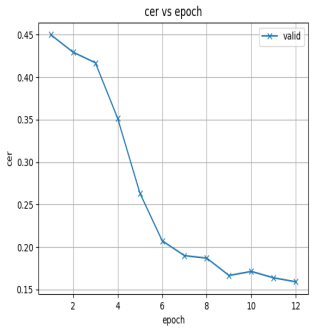


Figure 3: Character Error Rate (CER)

Automatic Speech Recognition (ASR) has been revolutionized by Artificial Intelligence (AI), particularly through the adoption of Deep Learning (DL) techniques. DL architectures, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs). These concepts empower ASR models to directly extract and comprehend complex speech patterns from raw audio, achieving unprecedented levels of transcription accuracy and efficiency. CNNs are used for processing structured grid-like data such as images. RNNs are used for processing sequential data, such as time series or speech recognition

Additionally, encoder-decoder models, a prominent framework in DL, have played a crucial role in ASR advancements. These models consist of an encoder that learns to represent the input audio features and a decoder that generates corresponding text transcriptions. This architecture enables end-to-end sequence -to-sequence learning, simplifying the ASR process by allowing the model to directly map input speech signals to output transcriptions without relying on intermediate representations.

One such model is the Listen, Attend, and Spell (LAS) model . The LAS model combines the strengths of encoder-decoder architectures with an attention mechanism, allowing the model to dynamically focus on relevant parts of the input sequence during transcription. This not only improves the accuracy of transcriptions, especially in the presence of long audio sequences or complex speech patterns but also enhances the model's robustness to noise and variations in speaking styles.

Deep Learning techniques like the LAS model with attention mechanisms have greatly boosted ASR accuracy. Training on large datasets like the 351-hour Hindi dataset using ESPnet on GPUs for 72 hours achieved impressive results.

By leveraging the benefits of the LAS model alongside other deep learning DL techniques, ASR systems have seen significant advancements in performance and usability. The availability of large-scale annotated speech corpora has further facilitated the training of deep learning models for ASR, providing diverse and extensive coverage of spoken language. These datasets enable neural networks to learn robust representations of speech patterns and variations. For instance, we utilized the Hindi dataset provided by SPRING Lab , IIT Madras, containing 351 hours of speech data. To train the model we use ESPnet, a speech processing toolkit developed by the Center for Speech and Language Processing at Johns Hopkins University.

In our model, for encoder, we have used Bidirectional Long Short-Term Memory (BiLSTM), a variation of Long Short-Term Memory (LSTM, a type of RNN is designed to capture long-term dependencies in sequential data by introducing memory cells and gating mechanisms), which processes input sequence in both forward and backward directions. This allows the model to capture contextual information from both past and future states, enhancing its ability to understand and represent the dependencies in sequential data. For decoder, we have used LSTM

After training the model having 5.38 million parameters with wall time of 72 hours on a single node with both the GPUs above figures depicts the performance of the model. 'Figure 1' plot shows how the accuracy of the model changes over training epochs. The accuracy typically increases as the model learns from the training data. However, if the accuracy on the validation set starts decreasing while the training accuracy continues to increase, it could indicate over fitting. 'Figure 2' plot displays how the loss function, which measures the difference between the model's predictions and the actual targets, changes over training epochs (Loss function can vary depending on the task, such as mean squared error for regression tasks or categorical cross-entropy for classification tasks) The loss usually decreases over time as the model improves its predictions. Similar to accuracy plot, and increase in loss on the validation set while the training loss decreases might suggest overfitting.

ASR is evaluated using the Character error rate (CER) shown in 'Figure 3'. CER is a metric used to evaluate the performance of an ASR system. It measures the rate at which characters in the recognized text output differ from the characters

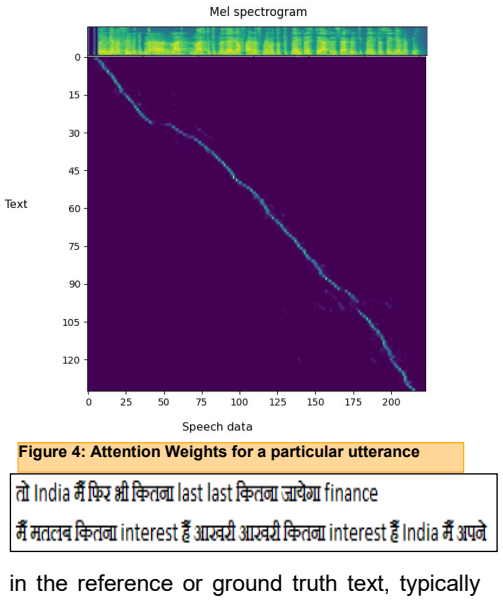


Figure 4: Attention Weights for a particular utterance

in the reference or ground truth text, typically expressed as a percentage.

'Figure 4' visualizes the alignment between the input audio features and the recognized output text. It shows how different parts of the input audio correspond to different parts of the recognized text, indicating where the model focuses its attention during the transcription process.

References:

1. *Chan,W., Jaitly, N., Le, Q. V., and Vinyals, O. (2015). Listen, attend and spell. arXiv:1508.01211 [cs, stat].*
2. *Nithya R, Malavika S, Jordan F, Arjun Gangwar, Metilda N J, S Umesh, Rithik Sarab, Akhilesh Kumar Dubey, Govind Divakaran, Samudra Vijaya K, Suryakanth V Gangashetty. (2023). "SPRING-INX: A Multilingual Indian Language Speech Corpus by SPRING Lab, IIT Madras."*
3. *Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, Adithya Renduchintala, and Tsubasa Ochiai, "ESPnet: End-to-end speech processing toolkit," in Interspeech, 2018, pp. 2207–2211*

Part 3: Simplifying TensorFlow Work: Run Jupyter Notebooks Inside Pre-built Containers

In the third part of this series, this article delve into the practical steps of leveraging jupyter notebook from Tensorflow container pulled from Docker Hub (can be accessed from here) with Singularity in HPC settings. The pre-built Tensorflow container must include jupyter-notebook or jupyter-lab while building the Docker image. User may find and use the appropriate tags and pull the image file and use jupyter notebook embedded within the container.

```
# Load singularity module available in ANTYA
[user@login1 ~]$ module load  singularity/3.5.3/3.5.3
```

Step 1: Downloading container from Docker Hub which has jupyter notebook enabled while building docker image.

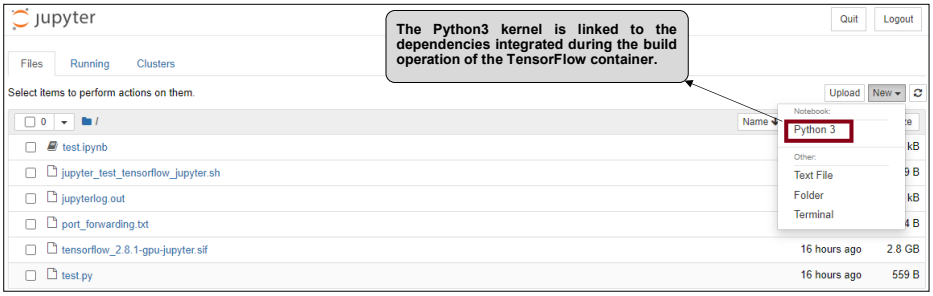
```
# Pull Tensorflow container from Docker Hub
[user@login1 ~]$ singularity pull docker://tensorflow/tensorflow:2.8.1-gpu-jupyter
```

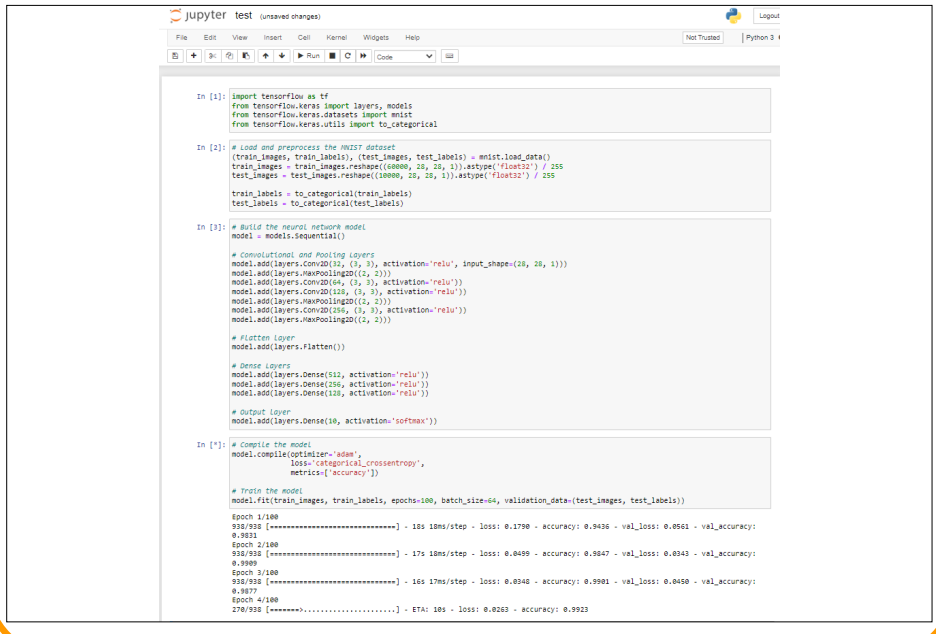
Step 2: Running Jupyter Notebook from container in your local machine browser using port forwarding.

To run notebook on a user local machine, user may try following the steps mentioned in HPC Newsletter Issue 3 here. The mentioned steps will open jupyter notebook in default anaconda environment. To load environment from singularity container, a command to run jupyter notebook from singularity container must be incorporated in script file.

```
module load singularity/3.5.3/3.5.3
# launch the Jupyter Notebook run from singularity container
singularity run --nv tensorflow_2.8.1-gpu-jupyter.sif jupyter notebook --no-browser --ip=${node} --port=${port} > ${NOTEBOOK_LOGFILE} 2>&1
```

The above script will assign compute resources to run jupyter notebook and will port forward to local machine via ssh. User may then proceed with further steps as mentioned in HPC Newsletter Issue 3. The kernel associated with the notebook encompasses all dependencies installed while building Tensorflow container which can be downloaded from DockerHub.



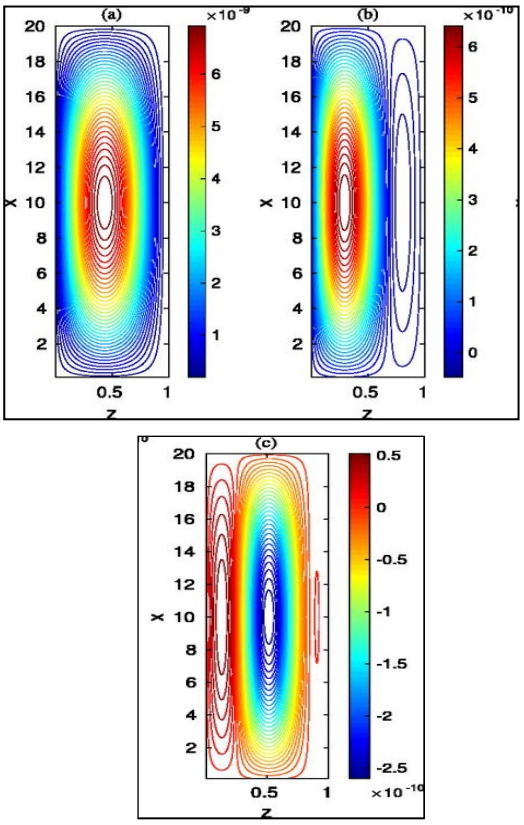


ANTYA UPDATES AND NEWS

1. New Packages/Applications Installed

To check the list of available modules
$ module avail –l

HPC PICTURE OF THE MONTH



Pic Credit: Prince Kumar

Streamlines for the dust fluid flow are driven by the background weakly magnetized plasmas. The streamlines are plotted with difference values of the driver mode number (a) m=1, (b) m=2, and (c) m =3, using parameter α = 1 L^{-1} and B_0 = 6.4×10^{-10} $m_d U_A / q_d L$. The parameter B_0 and α represent the magnetic field and its gradient, where normalized parameters are acoustic velocity $U_A = 10^5$ cm/sec, charge on dust $q_d = 10^{-16}$ C, mass of dust $m_d = 10^{-14}$ Kg, and simulation box length L = 10 cm .

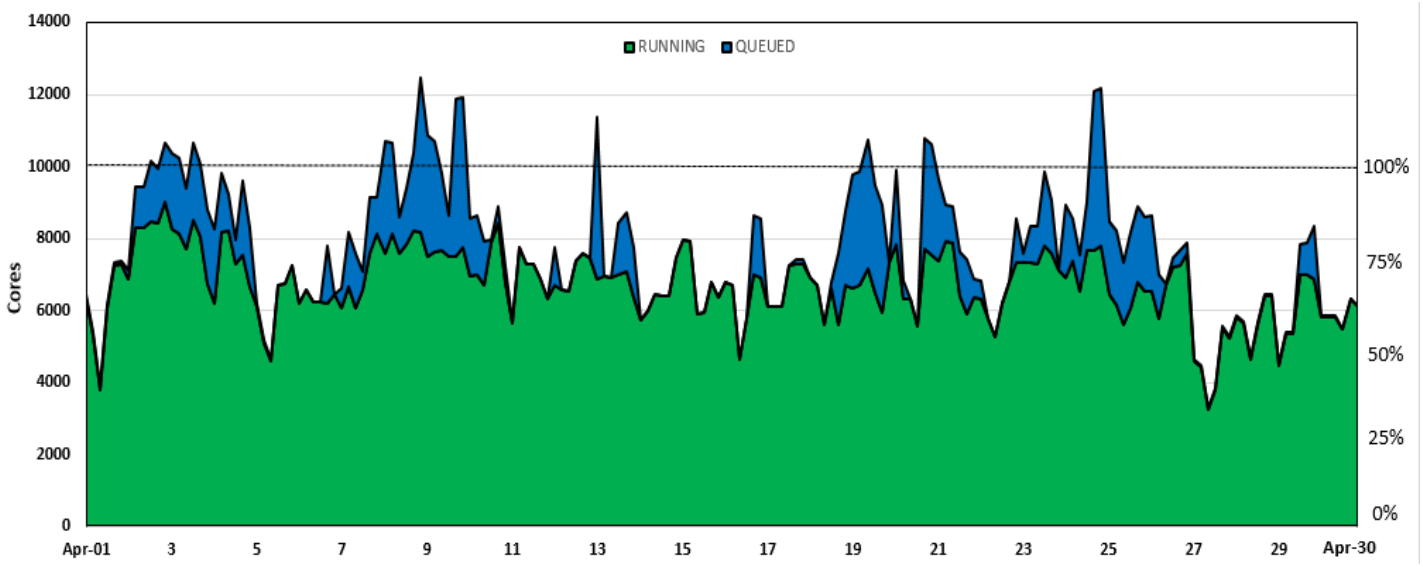
The figures are generated in MATLAB with data obtained from in-house developed FORTRAN code on ANTYA.

Reference:

P. Kumar and D. Sharma, Phys. Plasmas 27, 063703 (2020). https://doi.org/10.1063/5.0010850

ANTYA Utilization: APRIL 2024

ANTYA Daily Observed Workload



Other Recent Work on HPC (Available in IPR Library)

Laser-cluster interaction in an external magnetic field: The effect of laser polarization	Kalyani Swain
ADITYA and ADITYA-U Tokamak – An Epitome of Fusion Research in India	Joydeep Ghosh
Development of wideband 10 kW Solid State Power Amplifier - Challenges, Remedies and Test results	Manojkumar Arvindbhai Patel
Minimisation of Phase Error of Antenna-Plasma Coupling Impedance Using Least Square Technique for Ion Cyclotron Range of Frequencies	Dr. Vangalla Veera Babu
A mini-review on plasma blob formation mechanism	Nirmal K. Bisai
Understanding toroidal non-neutral plasmas	Rajaraman Ganesh
Experiments on the formation and melting of dusty plasma crystals in a DC glow discharge plasma	Pintu Bandyopadhyay

ANTYA HPC USERS' STATISTICS—

APRIL 2024

Total Successful Jobs~ 1806

♦ **Top Users** (Cumulative Resources)

- **CPU Cores** Amit Singh
- **GPU Cards** Shishir Biswas
- **Walltime** Shishir Biswas
- **Jobs** Sagar Choudhary

Acknowledgement

The HPC Team, Computer Division IPR, would like to thank all Contributors for the current issue of *GANANAM*.

On Demand Online Tutorial Session on HPC Environment for New Users Available
Please send your request to ***hpcteam@ipr.res.in.***

Join the HPC Users Community
hpcusers@ipr.res.in
If you wish to contribute an article in GANANAM, please write to us.

Contact us
HPC Team
Computer Division, IPR
Email: ***hpcteam@ipr.res.in***