Aniket Saxena

aniket.saxena756@gmail.com Aniket Saxena Aniket Saxena

Education

Aug 2014 – Jul 2018 | **University: Dr. A.P.J. Abdul Kalam Technical University, AKTU**
College: Meerut Institute of Technology, MIET Group, Meerut.
Degree: B.Tech in Computer Science and Engineering
Percentage: 75.8/100 (First Division with Honours)

Publications

2021 | *V. BK, B. Ganesan,* ***A. Saxena****, D. Sharma, A. Agarwal, "Towards Automated Evaluation of Explanations in Graph Neural Networks," 2021 International Conference on Machine Learning (ICML) Workshop on Theoretic Foundation, Criticism, and Application Trend of Explainable AI (XAI).* ***(Link)***

2019 | ***A. Saxena*** *and A. Saxena, "Optimal Partition Search," 2019 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT), pp. 1-8, doi: 10.1109/ICECCT.2019.8869459.* ***(Link)***

2017 | *A. Saxena,* ***A. Saxena****, J. Patel, "DeepCoder: An Approach to Write Programs," 2017 International Conference on Advanced Research and Innovation in Engineering (ICARIE), International Journal of Engineering and Manufacturing Science (IJEMS), pp. 9-13, Vol. 7, No. 1, Research India Publications.* ***(Link)***

Work Experience

Indian Institute of Technology, Kanpur Jan 2022 - June 2022
Advisor: Arnab Bhattacharya (Research Assistant)

- Proposed a novel neural network-in-network architecture-based approach to search down multidimensional arrays. Invariant to the size, the number of dimensions, and the scale of elements that any multidimensional array possesses, the proposed approach considerably surpasses the human-level performance and thus makes the proposed framework state-of-the-art for the multidimensional array search task.
- Performed various ablation studies to come up with a more promising network architectural design in order to produce an interpretable, yet a scalable solution to the proposed task.
- Proposed the very first large-scale benchmark dataset for multidimensional array search. This work is submitted to the IEEE Transactions on Neural Networks and Learning Systems.

IBM Global Business Services, India Sept 2018 - July 2020
Collaborators: Gomathy Shankar Ratnasabapathy and Rohit Singh (Data Scientist)

- Built a data analytics platform from scratch for monitoring and tracking the traversed path of the ship.
- Implemented various regressive prediction algorithms such as feedforward networks, support vector regression, etc. as a part of the platform to predict the movement of the ship in terms of longitudes and latitudes.
- Implemented an automated pipeline for pulling data from the PostgreSQL database and preprocessing it before feeding it into the prediction algorithms.
- Integrated an intuitive data visualization framework, built using the Dash-Plotly library, into the data analytics platform for creating intuitive visualizations for the ship strategy data obtained from various tables of the PostgreSQL database and discovered daily patterns in moving directions of the ship that could aid our prediction algorithms.
- Implemented an explainability solution by computing Shapley values for the features present in the dataset using the Scikit-Learn library. Features that had higher Shapley values had been considered more important for the prediction made by the prediction algorithms.

IBM Research, India July 2020 - Dec 2022
Collaborators: Balaji Ganesan, Muhammed Ameen, Avirup Saha, and Sameep Mehta (Research Associate)

- Worked on creating a framework to transform the domain-specific tabular data into heterogeneous knowledge graphs. The framework consumes tabular data as a property graph (unique to different domains), models the

various available relations in the knowledge graph using multi-layered RGCN, and generates a domain-level importance weight for each relation present in the input relational graph as output.
- To make the overall approach interpretable, introduced an explainability component in the framework to produce human-intelligible explanations for every single prediction being made for the input knowledge graph.
- Developed and analyzed an automated approach to further evaluate GNN explanations to boost their applicability to end users, thereby promoting the creation of more complex as well as trusted AI systems for the Graph-structured domain.
- The developed framework was also employed to create a novel large-scale graph representation learning dataset called the TACRED people dataset, which is being used by other teams in the lab to accelerate graph ML research.
- Developed and implemented an unsupervised and computationally light framework that facilitates clustering to partition each heterogeneous graph associated with every single entity present in Master Data to generate the smallest possible graph sub-structures for each entity having only monumental similar nodes clustered together. Moreover, the framework takes into account the primitive structure as well as node embeddings of each entity's heterogeneous graphs to cluster the nodes according to the graph topology (nodes in the same cluster should be strongly connected) and to the node features (nodes in the same cluster should have similar features).

IBM Systems Development Lab, India Apr 2022 - Mar 2023
Collaborators: Thanh T Pham, Paul Muench, Sandeep R Patil, and Binayak Dutta (Data Scientist)

- Created a time series-based neural learning framework for predicting the capacity of storage systems.
- Utilized generic machine learning algorithms (LSTM, Linear Regression, and RNN) to impeccably predict run-out memory times of storage systems in the resources pool.
- Developed a novel approach to segment storage systems based on their respective predicted behaviors in terms of average growth in usable capacity over a specific forecasting window period, which in turn helped fill the missing data present in the time series.
- Implemented a data extraction pipeline to extract data in the form of multivariate time series from a storage data lake. Work done in close collaboration with IBM Research, Almaden.

Projects

- **Contrastive Learning-based Few-shot Image Classification** ***(Link)***
  Libraries/Framework: Numpy, Matplotlib, PyTorch, and TorchVision
  - Defined a custom contrastive loss and trained a few-shot version of Siamese Networks to do n-way k-shot image classification by mapping the image similarity task into a fully-supervised classification learning task.
- **Abstractive Text Summarization** ***(Link)***
  Libraries/Framework: NLTK, Keras, Tensorflow, Numpy, and Pandas
  - Implemented an Attention-based Stacked LSTM Encoder-Decoder architecture of the Seq2Seq Model to summarize the long sequences of words into their respective short versions of relevant sentences based on their semantic proximity.
  - In essence, the implemented model is able to not only capture the long-term dependencies in long-paragraph summarization but also generate a legible summary based on the contextual information present in the given textual paragraphs.
- **Text Generation** ***(Link)***
  Libraries/Framework: Numpy and Tensorflow
  - Trained a multi-layer LSTM network for word-level language modeling. Used this trained model to generate text, which shares similar statistical properties with those used for training the model. Moreover, also performed transfer learning using pre-trained GloVe word embeddings to improve the overall accuracy of the model being trained.
- **Zero-shot Question Answering with Large Language Models** ***(Link)***
  Libraries/Framework: Transformers, Numpy, and Tensorflow
  - Implemented a zero-shot question-answering system that, for each question "q" with available answer options "a", "b", and "c", computes each option's score as the negative log-likelihood under the language model conditioned on the question and then returns the option with the highest score as the most probable answer

to the question “q”.

Open-Source Machine Learning Contributions for PyTorch Geometric Library

Relational Graph Attention Networks *(Link)*
Libraries/Framework: torch-scatter, torch-sparse, PyTorch, PyTorch Geometric, and PyTest

- This contribution relates to bringing attention to heterogeneous (relational) graphs and incorporating this relation-aware attention operator into PyTorch Geometric Library.
- Offers two different mechanisms to compute attention for relational graphs, i.e. **within-relation** and **across-relation**.
- This implementation also provides four different cardinality preservation options (**additive**, **scaled**, **f-additive**, and **f-scaled**) to further improve attention computation for the heterogeneous graphs.
- Wrote custom unit tests to verify the technique’s accuracy.

Clustering for Graph-structured Data using Graph Neural Networks *(Link)*
Libraries/Framework: PyTorch, PyTorch Geometric, and PyTest

- This contribution relates to adding up a fast, yet effective graph clustering technique to PyTorch Geometric library based on **Spectral Modularity Maximization**.
- A multi-layer graph convolutional network (GCN) is used to learn the hidden representations of each of the nodes in the graph.
- Softmax activation is then applied to the output of GCN to obtain **soft, yet differentiable cluster assignments** being made for the input graph.
- Designed and implemented unit tests to confirm the correctness of the clustering technique.

Technical Skills

- Programming Languages- C, C++, Python, LaTeX
- Tools/IDE- PyCharm, Spyder, Jupyter Notebooks, Google Colab
- Framework/Libraries- Tensorflow, Keras, PyTorch, PyTorch Geometric, DGL (Deep Graph Library), Scikit-learn, Numpy, Pandas, Dash-Plotly, Matplotlib, NetworkX, Transformers

Achievements and Extracurricular Activities

- Submitted a Patent application and hence felicitated by **2023 IBM Invention Achievement Award**.
- Secured **First Division with Honors** at undergraduate level (Bachelor of Technology (B.Tech)) in Computer Science and Engineering.