Aniket Saxena

aniketrec.123@gmail.com +1 (587) 938-9321 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
Division: First Division with Honours

Sep 2023 – Present
University: University of Montreal, Canada
(MILA- Montreal Institute of Learning Algorithms)
Degree: Masters in Computer Science (Machine Learning Specialization)
Courses: Fundamentals of Machine Learning, Data Science, Representation Learning, Natural Language Processing (NLP)

Achievements and Extracurricular Activities

- Submitted two Graph Neural Networks (***GNNs***)-based Patent applications to the US Patent Office, including one **first-inventor Patent application** and hence felicitated by **2023 IBM Research Invention Achievement Award**.
- Awarded **Academic Excellence Award** for exceptional performance during Academic Year 2015-2016.
- **Service Excellence Award** by IBM Systems Development Lab for the year 2022.
- **Diversity Award (Tuition Fees Scholarship/Exemption)** by University of Montreal.

Publications

2023 ***A. Saxena**, A. Bhattacharya, "MASNet: A Simple Neural Network Approach to Solve Multidimensional Array Search," Under Review at IEEE Transactions on Neural Networks and Learning Systems*

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

IBM Global Business Services, India Sept 2018 - July 2020
(Data Scientist)

- Built a machine learning platform from scratch for monitoring and tracking the traversed path of vessels.
- Implemented various regressive prediction algorithms such as feedforward networks, support vector regression, etc. as a part of the platform to predict the movement of vessels in terms of multi-output regressive values (longitudes and latitudes).
- Implemented automated *MS Azure machine learning pipelines* to fulfill various purposes: (i) pulling data (containing 39 features for each of the 120k vessels) from the PostgreSQL database; (ii) training and evaluating the regression algorithm; (iii) to implement efficient batch-inference method to predict regressive outputs; (iv) to deploy the platform using *Azure CI/CD actions* that made the platform accessible to more than 2k customers in real-time.

- Integrated an intuitive data visualization framework, built using the *Dash-Plotly* library, into the platform for creating intuitive as well as user-interactive visualizations to discover daily patterns in moving directions of vessels that could further aid our prediction algorithm.
- Implemented an explainability solution by computing *Shapley Additive Explanation* values for the features present in the dataset using the *Scikit-Learn* library. Features that had higher *Shapley* values were considered more important for the prediction made by the prediction algorithm.

Technical University of Dortmund, Germany Dec 2020 - June 2021
(Research Intern)

- Designed and implemented a unique post-hoc model-agnostic Graph Neural Networks (GNNs) explainability framework. Unlike other traditional GNN explainability methods, the implemented approach does not require retraining to generalize explanations to new instances for node and graph classification, and link prediction tasks.
- The framework takes a trained GNN model along with some meta-data information of the input graph as input, computes layer-wise edge importance weights, and outputs an explanatory subgraph.
- The implemented GNN explanation framework is available on PyTorch Geometric. ***(link)***

IBM Research, India July 2020 - Dec 2022
(Research Associate)

- Worked on creating a framework to transform the domain-specific ontologies into large-scale enterprise heterogeneous knowledge graphs (each comprises billions of nodes and edges connecting them). The framework consumes an ontology as a property graph (unique to different domains), models the various available relations in the knowledge graph using a multi-layered Relational Graph Convolutional Network (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. Work got published in the ***Explainable AI Workshop at ICML 2021***.
- 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.

IBM Research, Zurich Feb 2021 - Dec 2021
(Research Associate)

- Developed a novel unsupervised contrastive learning-based approach for the detection of similar bugs to facilitate faster log analysis. The approach extracts documents from the corpus of bug text, learns a latent representation using a pre-trained BERT language model to form a document embedding for each document, and then measures the degree of similarity between these learned document embeddings of all extracted documents using newly proposed Word Centroids (WC) and Word Mover Distance (WMD) algorithms.
- Upon obtaining the semantic-similarity scores, the approach clusters the similar bugs together based on semantic textual similarity indexing, which is governed by these computed scores.
- Included main measures to evaluate the approach (e.g. novel and repeated). Experiments conducted on the synthetic bugs corpus strongly recommended the complete elimination of manual detection of similar bugs.

IBM Systems Development Lab, India Apr 2022 - June 2023
(Data Scientist)

- Created a multivariate 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.
- Designed and implemented pipelines for various stages of development, starting from extracting high-dimensional

data in the form of multivariate time series (with underlying non-continuous dynamic) from a storage data lake to deploying the whole approach as a fully functional ML-powered product accessible to numerous customers.

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.
- **Graph Representation Learning for Classification** *(GitHub) (Deployed App) (App Demo Video)*
  Libraries/Framework: PyTorch, PyTorch Geometric, DVC, MLFlow, Optuna, and Streamlit
  - Developed an end-to-end Node and Graph classification app, which classifies a node or a graph on different datasets (*Cora* and *ENZYMES*).
  - Utilized an abstract variant of *Binomial Distribution* to compute feature and edge importance weights that result in providing explained feature mask and explanation subgraph.
- **LLM-RAG-powered-QA-App** *(GitHub) (App Demo Video)*
  Libraries/Framework: Transformers, Ray, LangChain, FastAPI, and PyTorch
  - Fine-tuned a 20B parameters Large Language Model (LLM) by leveraging the distributed training paradigm.
  - Developed a production-ready, scalable Retrieval Augmented Generation (RAG)-based context-aware Question Answering (QA) App that first finds contexts relevant to the incoming query by implementing fast vector similarity search within the pre-defined embedding space and then sends these contexts alongside the query to the fine-tuned LLM model to generate the answer.
- **Molecule Graph Generation** *(Link)*
  Libraries/Framework: PyTorch, PyTorch Geometric, Numpy, and NetworkX
  - Implemented a minimal version of Graph Convolutional Networks-based Variational Graph AutoEncoders (VGAE) for generating molecule graphs.
  - More precisely, VGAEs were utilized to generate new molecular graphs that have a similar statistical distribution as that of the learned distribution of the training dataset.
  - Experiments were performed using the ZINC dataset consisting of 12k molecular graphs with 38 heavy atoms.

Open-Source Machine Learning Contributions

Contributed to **PyTorch Geometric** to add new features and operators such as Relational Graph Attention Networks, Clustering for Graph-structured Data using Graph Neural Networks, and Model-agnostic Explainability Framework for Graph Neural Networks.

Skills: Soft and Technical Skills, Certifications, and Courses - Online Learning

Soft Skills	Leadership, Teamwork, Adaptability, Positivity, Interpersonal Skills, Problem-Solving Skills, Creative thinking
Technical Skills	**Interests-** Generative Modelling (Generative AI), Natural Language Processing (NLP), Time Series, Machine Vision, Learning on Graphs, MLOps **Databases-** MySQL (SQL), PostgreSQL (extended SQL). **Programming Languages-** Python, LaTeX. **Framework/Libraries-** Tensorflow, Keras, PyTorch, PyTorch Geometric, DGL (Deep Graph Library), Scikit-learn, Numpy, Pandas, Dash-Plotly, Matplotlib, NetworkX, HuggingFace (*transformers*), FastAPI, Flask, MLFlow, Optuna, LangChain, Ray (distributed machine learning framework) **Other Tools-** Git, Docker, Streamlit, MS Azure, Google Cloud Platform (GCP), GitHub Actions, DVC (for data, model, and code version control)
Certifications and Skill Courses	Machine Learning A-Z with Python Course (Udemy), Machine Learning (Coursera), Machine Learning Crash Course (Google), Columbia University's AI Specialization, Python for Data Science (IBM), Deep learning specialization (Coursera)