Joseph Chan

Website: jzchan132.github.io | **Email**: jzchan132@gmail.com | **LinkedIn**: jzchan132 | **GitHub**: jzchan132

Projects

Analyzing Explanations for BERT/RoBERTa Language Models

- Trained BERT/RoBERTa models for sentiment analysis and relation extraction, using PyTorch and Hugging Face.
- Generated extractive explanations for BERT/RoBERTa models' predictions, using attribution algorithms (integrated gradients, gradient*input, occlusion) in Captum.
- Analyzed BERT/RoBERTa model explanations from different attribution algorithms by visualizing explanations as heatmaps and quantitatively comparing them to human-annotated explanations.

Training Neural Networks for Computer Vision

- Trained convolutional neural networks (CNNs) on various computer vision datasets: **(1)** custom CNNs on MNIST and CIFAR-10 image classification datasets, **(2)** U-Net CNN on CamVid semantic segmentation dataset, **(3)** Inception CNN with triplet loss on LFW face recognition dataset, and **(4)** VGG-16 CNN on in-house dataset of classmates' faces.

Extracting Insights from Twitch Streamer Data

- Cleaned and analyzed data about the top 1000 Twitch streamers, including features like watch time and followers gained.
- Plotted Pearson correlation heatmaps and boxplots to visualize pairwise feature relationships, using Pandas and Seaborn. Discovered that the watch time per follower is unusually high for Thai-speaking Twitch streamers.
- Used Scikit-learn to fit linear regression models for the most correlated feature pairs.

Experience

First-Year Innovation & Research Experience (FIRE) College Park, MD
Student Researcher, Capital One Machine Learning (COML) Stream *Aug. 2019 - Dec. 2020*

- Implemented deep learning models for various computer vision and NLP tasks, using Keras and TensorFlow.
- Trained deep learning models, using both cloud (Google Colab, Google Compute Engine) and university GPU servers.
- Extracted insights from real-world datasets, using spreadsheet tools like conditional/lookup functions and pivot tables.

CBCM Gaithersburg, MD
Software Engineering Intern *June 2019 - Aug. 2019*

- *Open Sesame* is an in-house Android app for wirelessly managing building access, especially during off-hours. Users on the building's network can easily control physical door locks via the *Open Sesame* app.
- Independently revamped *Open Sesame*'s codebase (C#, Xamarin) to patch existing login vulnerabilities and make the app compatible with Android 8.1.
- Regularly met with both technical and non-technical collaborators to discuss design specifications, present progress updates, and obtain user experience feedback.

U.S. Army Research Laboratory Adelphi, MD
Research Intern *June 2018 - Aug. 2018*

- Designed an algorithm for automatically reorienting laser communication gimbals in military vehicles, in order to maximize inter-vehicle signal strength in the battlefield. Implemented the gimbal adjustment algorithm using C++.

U.S. Army Research Laboratory Adelphi, MD
Research Intern *June 2017 - Aug. 2017*

- Built a Python+LabOne based laser system to automate the laborious process of locating spin defects in silicon carbide samples. Such spin defects are useful for many applications, such as quantum computing, cryptography, and bioimaging.
- Created an intuitive GUI with Tkinter and Matplotlib to help a team of physicists easily run the laser system and visualize their results with 3D plots, thus greatly speeding up their experiments.

Education

University of Maryland College Park, MD
BS in Computer Science, Minor in Physics *Aug. 2019 - Present*

- **First-Year Innovation & Research Experience (FIRE):** Accelerated undergraduate research program for students interested in machine learning. Students develop hands-on experience with using deep learning to solve problems in computer vision and NLP.
- **Relevant Coursework:** Machine Learning, Data Science, Linear Algebra, Algorithms, Discrete Structures

Skills

Languages: Python, Ruby, Rust, Java, JavaScript, HTML, CSS, C++, C#, Matlab, Bash
ML Tools: PyTorch, PyTorch Lightning, Neptune, Captum, Scikit-learn, TensorFlow, Keras, Google Colab
NLP/CV Tools: Hugging Face, OpenCV, Imgaug
Data Analysis Tools: NumPy, Pandas, Seaborn, Microsoft Excel, Google Sheets
Other Tools: Git, Hydra, Docker, Gatsby, NodeJS, Django, PostgreSQL, SQLite, LabOne, Unity