

List of Some Selected Projects in Domain of Intelligent Systems

Open Source Contributions

Google Summer of Code at NRNB (worked with [Prof. Herbert Sauro](#))

- Modified existing plugins in Plugin Loader System such that it works independently from the libRoadRunner(host)
- Simplified development of New Plugins by detaching the need to rebuild the whole system to deploy one plugin
- Removed legacy code of RRPlugin and Ported sample codes for plugins from python 2 to python 3
- Detailed Report and Code [Link](#)

Google Summer of Code at NRNB (worked with [Prof. Herbert Sauro](#))

- Integrated RRPlugin (Plugin Manager) and Libroadrunner, high performance & portable simulation engine to load plugins
- This resulted in the reduction of build time by 25% and eases the overall build process for the Libroadrunner
- Used SWIG to develop a Python Wrapper for plugins provided in C++ to the Python developers for faster computation
- Detailed Report and Code [Link](#)

Existing Project

- **Text to Video Retrieval at Centre Visual Information Technology (working with [Prof. Makarand Tapaswi](#))**
 - Exploring the developments in vision and language scoring to improve alignments between plot synopsis and videos.
 - Enhancing score, by using modalities like speech, body language, facial expressions, and actions, to get better R@k.
 - Working on text to video retrieval system over the Condensed Movie dataset, using Max-Margin Ranking Loss.
- **Temporal Graph Convolutional Network for Urban Traffic Flow Prediction Method**
 - Using GCN to capture spatial structure of the graph and GRU to capture temporal nature of the road using time series data.
 - Experimenting with Temporal GCN to capture more topological structure of the road network.
 - Experimenting with embeddings of spatial and temporal dependencies to use transformers. Research suggests that it will be able to better use seasonal time series average speed of the vehicle on the road.

- **Style Transfer**
 - We tried to transfer the style of one image to another image. This is a classic problem and lots of solutions are already present.
 - Trying to use style and content as an input to transfer and use intermediate layers to generate mixed pictures.

Past Project

- **Explainable Detection of Online Sexism (SemEval Shared Task 2023) at Language Translation and NLP Lab (worked with [Prof. Mansih Shrivastava](#))**
 - Implemented a PyTorch model for Detecting Online sexism by combining DeBERTa & RoBERTa to yield an F1 score of 82.96
 - Improved the model by 2.7%, using Domain Adaptive Pre Training which trains on mask language modeling over 2M sample
 - To explain the cause of sexist text, we performed classification over category of sexism and fine-grained category of sexism
 - Submitted paper at SemEval 2023, ACL, which is under review as of now
- **Wikipedia Search Engine (Indexing and Ranking Application)**
 - Built a scalable and efficient Search Engine on a large Wikipedia Dump of size around 80GB
 - Parsed the XML dump, processed words and created an Inverted Index by recording word counts on each page
 - Facilitated Multi-Field Query Search, which consists of query based on title, categories, external links, body, etc
 - Implemented Page Ranking mechanism using Tf-Idf metric. Optimized for index creation and index size and search time
- **Story Generation given Prompt**
 - Implemented a Two Layered, Fine Tuned GPT2 architecture to write a story based on the prompt provided
 - First layers consist of the prompt and outline generated using Text Rank
 - Second Layer takes prompts and outlines to generate a story. Both the layers are trained on Cross Entropy Loss
- **Summarization using Pointer Generation**
 - Used sequence to sequence architecture with attention to summarize the text
 - To deal with out of word vocabulary, implemented pointer generation on top of seq2seq attention, to give better results
- **Fake News Detection**
 - Fine-tuned BERT and RoBERTa model to train on LIAR Dataset.

- Experimented with fixing and changing parameters of different layers and got an accuracy of 28%.
 - The final model outperformed the hybrid approach of CNN and Text, described in the LIAR paper.
- **Machine Translation (From English to Hindi) and Creating Word Embedding**
 - Created Word2Vec embedding using CBOW (Continuous Bag of Words) and Co-occurrence Matrix Algorithm
 - Implemented Encoder Decoder (Seq2Seq) Architecture with and without Attention for Machine Translation