Hoang Pham

linkedin.com/in/hoangph3

Email : phamhoanghxh1@gmail.com

Mobile : +84 339 362 666

I like building large-scale ML pipeline!

Experience

- **Machine Learning Engineer (Mid-level)** Jul 2021 - Present
  Viettel Cyber Security *Hanoi, Vietnam*
  - **E2E Pipeline Design**: Architected and implemented scalable machine learning pipelines to automate the data ingestion, processing, model training, evaluation, and deployment processes.
  - **Data Preprocessing**: Setup, schema design and management of Hadoop use Apache Ambari. Built an internal app for access to the data using a web interface. Dataduct integration for daily ETL injection into OneFS.
  - **Model Training and Tuning**: Implemented automated model training workflows using frameworks like TensorFlow, Torch, or Sklearn. Leveraged hyperparameter tuning strategies to optimize model performance.
  - **Monitoring and Maintenance**: Established monitoring solutions for model performance and data drift, ensuring the reliability and accuracy of deployed models. Implemented retraining mechanisms based on new data or model decay.
- **Student Internship** Jul 2020 - Jul 2021
  Viettel Cyber Security *Hanoi, Vietnam*
  - **Detecting Similar Security Alerts**: Developed and implemented machine learning models to automatically cluster and correlate similar security alerts. Leveraged techniques like natural language processing (NLP) and unsupervised learning to group related alerts.
  - **Command Line Anomaly Detection**: Developed a preprocessing pipeline for tokenization and text cleaning, ensuring compatibility with BERT. Implemented BERT to generate contextual embeddings, feeding them into models like isolation forests and one-class SVMs, to identify outliers.
- **Software Engineer AI** Jun 2022 - Present
  Freelance *Remote*
  - **Application of Pre-trained Models**: Specialized in applying pre-trained models for tasks such as NLP, computer vision, utilizing frameworks like ONNX, PyTorch, and Hugging Face.
  - **Solution Architecture**: Developed robust architectures for AI-powered applications, focusing on optimizing performance and ensuring seamless integration with existing software systems.
  - **System Integration**: Successfully integrated AI functionalities into client applications on edge platforms (Jetson), leveraging technologies like ONNX and TensorRT for deployment in resource-constrained environments.

Education

- **Hanoi University of Science and Technology** Hanoi, Vietnam
  Bachelor, Automation Engineer Technology; Grade: 3.37/4.0 *Sep 2017 – Jun 2021*

Certifications

- **TOEIC**: 640
- **Deeplearning.AI**: Machine Learning Engineering for Production (MLOps) Specialization
- **Deeplearning.AI**: Deep Learning Specialization
- **Stanford Online**: Machine Learning

Projects

- **Security Chatbots** Mar 2024 - Present
  Viettel Cyber Security
  - **SOAR Alert Identification**: Developed chatbots to identify and categorize SOAR (Security Orchestration, Automation, and Response) alerts, enabling faster and more accurate threat detection.
  - **Automated Alert Operations**: Implemented automated workflows within the chatbots to handle alerts based on predefined checklists. Utilized AI prompting to execute tasks efficiently and consistently.
  - **Alert Reporting**: Designed the chatbots to automatically summarize alert details and generate comprehensive reports, streamlining the process for end-users to review and respond to security incidents.

- **Tech Stack**: HuggingFace, Airflow, Celery, Docker, SQL, Metabase, Redis, Python.

- **Alert Response Automation** Jan 2023 - Present
  Viettel Cyber Security
  - **Embedding-Based Similarity Search**: Utilized embedding techniques to perform similarity searches on alerts, enabling the automatic identification and closure of false positives, improving overall efficiency and accuracy in alert management.
  - **False Positive Reduction**: Implemented automation for identifying and managing false positives, significantly reducing the manual workload for security teams.
  - **Incident Response Automation**: Developed systems to automate incident response processes, leveraging historical operating data to streamline investigations and reduce response times.
  - **Tech Stack**: Faiss, FastText, SQL, NoSQL, Flask, Kafka, Python, Nomad, Consul.

- **Manufacturing Chatbots** Oct 2023 – Mar 2024
  DENSO Vietnam
  - **Production Line Information Retrieval**: Developed chatbots to provide detailed production line information, such as equipment downtime and causes, by querying SQL Server databases.
  - **LLM Localization**: Customized large language models (LLMs) to understand and process manufacturing-specific queries, enhancing chatbot accuracy and relevance.
  - **Entity Extraction with RAG**: Implemented Retrieval-Augmented Generation (RAG) to extract named entities from user queries, improving the precision of information retrieval.
  - **Keyword Matching with Embeddings**: Used word embeddings for keyword matching, ensuring that the chatbot could accurately interpret and respond to user queries based on semantic similarity.
  - **Data Management**: Stored historical chat data and interaction logs using MongoDB, enabling easy access to past queries and responses.
  - **Telegram Integration**: Integrated the chatbot with Telegram, adding features like feedback buttons, daily reports, and command-based job execution to enhance user interaction and efficiency.
  - **Tech Stack**: HuggingFace, Telegram Bot, WebHook, FastAPI, Milvus, Celery, NoSQL, SQL, Python, Kubernetes.

- **Eyebrow Pose Estimation** Dec 2023 - Feb 2024
  NAL Vietnam
  - **Face Detection and Alignment**: Developed algorithms to detect and align faces, ensuring accurate positioning for subsequent eyebrow pose estimation.
  - **Eyebrow Key-Point Detection**: Implemented key-point detection models to identify critical points on the eyebrows, enabling precise pose estimation.
  - **IOU Calculation**: Computed the Intersection over Union (IOU) between predicted eyebrow poses and ground-truth data provided by stylists. Used affine transformations to scale, translate, and rotate coordinates for accurate comparison.
  - **Tech Stack**: YOLO, FastAPI, Python.

- **People Counting** Sep 2023 - Jan 2024
  Indochina Plaza Hanoi
  - **Multi-Camera Tracking**: Developed a system for tracking individuals using both single and multi-camera setups, with integration via RTSP streams from cameras deployed in shopping malls.
  - **Event-Driven Architecture**: Implemented an event-driven system using Redis and Kafka for message queuing, enabling real-time processing and scalability across distributed devices.
  - **Person Re-Identification (ReID)**: Employed embedding methods for person re-identification, utilizing similarity search and re-ranking techniques to accurately track and count individuals across different camera feeds.
  - **Distributed Edge Processing**: Deployed the system on multiple NVIDIA JetsonNX devices, optimizing the pipeline for real-time performance on edge hardware.
  - **Tech Stack**: NVIDIA Jetson, Pytorch, ONNX, Kafka, MongoDB, Clustering, Python, Docker.

- **Automatic Number-Plate Recognition (ANPR)** Feb 2023 - Jun 2023
  KOTORA
  - **Number-Plate Detection**: Developed algorithms for detecting vehicle number plates from camera feeds in an apartment complex, ensuring high accuracy in various lighting conditions.
  - **Number-Plate Processing**: Implemented key-point alignment and image enhancement techniques to preprocess number plates, improving recognition accuracy.
  - **Number-Plate Recognition**: Created a recognition system that accurately reads and records number plates, integrating with existing security and access control systems.
  - **RTSP Integration and Stream Processing**: Integrated the system with camera RTSP streams, utilizing Kafka for real-time stream processing, enabling efficient and scalable operation.
  - **Tech Stack**: NVIDIA Jetson, ONNX, OCR, Kafka, Flask, Python, Docker.

- **Face KYC** Sep 2022 - Feb 2023
  Telehouse Vietnam
  - **Camera Management**: Developed a system for capturing images from tablets or IP cameras, ensuring high-quality input for facial recognition processes.
  - **Model Serving**: Hosted ONNX-based face detection and recognition models on NVIDIA Jetson Xavier devices, leveraging CUDA for accelerated performance.
  - **Door Control System**: Implemented a door control mechanism using ESP32 and GPIO with UART protocol, allowing for secure access control through USB port communication.
  - **Licensing Management**: Designed a licensing system using JWT with expiration time to manage and secure API access, ensuring compliance with security protocols.
  - **Backend Development**: Managed backend processes including user management, CR handling, and integration with the CASDM Broadcom system, providing a robust and scalable infrastructure for the solution.
  - **Tech Stack**: NVIDIA Jetson, ESP32, Facial Recognition, Flask, NodeJS, Python, Docker.

- **Security Datamining** May 2022 - Dec 2022
  Viettel Cyber Security
  - **Hadoop Cluster Deployment**: Successfully deployed a Hadoop cluster on-premise to support large-scale security data mining operations.
  - **Cluster Integration**: Integrated the Hadoop cluster with a OneFS Isilon cluster, utilizing Kerberos and LDAP for secure authentication and access control.
  - **Data Workflows**: Developed and executed Spark jobs using Oozie workflows, enabling efficient data processing and analysis.
  - **System Hardening and Monitoring**: Implemented system hardening measures, performed debugging, and set up monitoring tools to ensure the stability and security of the entire infrastructure.
  - **Tech Stack**: Hadoop Administration, Apache Ambari, HDFS, OneFS, Kafka, Spark, Scala.

- **Typing biometrics** Jan 2022 - Mar 2022
  Viettel Cyber Security
  - **Typing Behavior Authentication**: Developed a system to authenticate users based on their typing patterns during login and continuous work sessions, enhancing security beyond traditional methods.
  - **Metric Learning Implementation**: Applied metric learning techniques to create an encoder that transforms typing behavior into distinct user features, enabling accurate identification.
  - **User Profile Management**: Implemented a system to save typing feature embeddings into user profiles during registration, with OTP verification for added security.
  - **Behavior Comparison**: Designed algorithms to compare new typing behaviors with stored user features, ensuring consistent and reliable authentication.
  - **Tech Stack**: Tensorflow, Flask, SQL, Clustering, Python.

- **Transaction Fraud Detection** Sep 2021 - Dec 2021
  Viettel Cyber Security

- **Pattern Extraction and Time Series Conversion**: Extracted patterns from abnormal transaction behaviors and converted them into multiple time series for detailed analysis.
- **Real-Time Behavior Comparison**: Implemented algorithms to compare real-time transaction behaviors with learned abnormal patterns, using techniques such as dynamic time warping and longest common subsequence for accurate detection.
- **Time Series Approximation**: Implemented a system to save typing feature embeddings into user profiles during registration, with OTP verification for added security.
- **Behavior Comparison**: Applied methods like Discrete Fourier Transform (DFT) and Piecewise Aggregate Approximation (PAA) to approximate time series data, enhancing the efficiency of pattern recognition and fraud detection.
- **Tech Stack**: Statsmodels, Matching Algorithms, Kafka, Pandas, Numpy, Python.

- **Text to Speech Synthesis** Feb 2021 - May 2021
CMC ATI
  - **Noise Handling and Dataset Cleaning**: Applied speech diarization techniques to manage and clean noisy datasets, improving the quality and accuracy of the speech synthesis process.
  - **Model Fine-Tuning**: Fine-tuned the WavLM model with a Vietnamese dataset to enhance its performance and adaptability for Vietnamese text-to-speech applications.
  - **Tech Stack**: Librosa, VAD, Pytorch.

- **Keyword Spotting** Sep 2020 - Nov 2020
RF3i Lab
  - **Audio Recognition Model**: Developed a simple model for recognizing keywords from a speech commands dataset, enabling voice control functionality.
  - **Infrared Signal Reading**: Utilized an IR-1838 sensor with a Raspberry Pi and LIRC (Linux Infrared Remote Control) to read infrared signals from a television.
  - **Signal Mapping and Emission**: Mapped recognized keywords to corresponding infrared signals and used an IR-T940 emitter connected to the Raspberry Pi to transmit signals to the television, allowing for device control through voice commands.
  - **Tech Stack**: Raspberry Pi, Tensorflow, TensorFlow Lite, Signal Processing, ASR, Python.