2023-2024 Student Summership Project List
Computer Science

Verifying the Veracity of Information Exchange Using the Event Calculus

Stephen Cranefield and David Eyers

For a business supply chain, say for the retail of organic wine, or the building and delivery of secure software applications, where are the riskiest points in the interactions between many different stakeholders involved? This summer project involves modelling types of multi-party interactions, while tracking participants' concerns about veracity (e.g., including authenticity and fulfilment of participants' expectations).

We will suggest use of extensions of the logic-programming-based event calculus (EC) to perform this modelling, or other tools developed by our other research collaborators.

That said, we do not assume any prior knowledge about logic programming or the EC. The end result is intended to be an interactive demonstration, e.g., presented within a Jupyter notebook (as commonly used in data science projects) running in the cloud.

Who Said That – Was it ChatGPT?

Andrew Trotman and David Eyers

The advent of large language models (LLMs) and generative AI could lead to a flood of disinformation sent to social media and email – disinformation generated anonymously by AI models under the instruction of nefarious players. LLMs could be, and evidence suggests already are being, used by students around the world to generate their homework answers (i.e. cheat). Producers of AI systems want to train their models on original human-generated information (text, images, etc.), they do not want to train a model on its own output.

In this project we will look at ways to identify whether a piece of text has been generated by a large language model or not. This isn't a simple binary classifier because the output text could be manipulated by a human (or software) in order to disguise the origin. For this reason we are interested in an approach based on plagiarism detection and in this project we will build a pipeline base on this approach.

Deep Trouble

Lech Szymanski and David Eyers

Transfer learning facilitates convenient reuse of machine learning models. Rather than training from scratch, in the context of reuse, partial retraining is performed over an existing source model. For example, training can be applied over a neural network with weights copied from the source network that remain fixed but for within a few final layers. Is there, from a security perspective, a potential risk of original training process embedding malicious data? Consider an unrealistically oversimplified, hypothetical scenario, in which a target neural network is trained, by transfer learning from a source neural network, to recognise faces for the purpose of controlling access to a building. Is it possible to embedded a pattern in the source capable of "surviving" target training and later causing a strong response that may act as a "master key"? This project will involve devising means of embedding source networks with "malicious" patterns. The aim is to determine the extent to which it is possible to have these patterns survive the transfer learning and whether they can affect a target network in any meaningful way.

ML for computationally cheap(er) climate modelling.

Lech Szymanski

Changes in ozone are an important variable used in climate change simulations. Physics-based atmospheric models for predicting the changes in ozone under various potential climate scenarios are computationally expensive. However, traditional machine learning (ML) may be capable of learning to make accurate ozone predictions from other factors with a significantly lesser computational burden. How good can those predictions get? What models are best? Does deep learning offer in this domain anything over the classical regression models? Are there any other aspects of climate modelling, aside from ozone changes, that ML is suitable for? This project will aim to answer some (if not all) of these questions. The work will involve a survey of the literature on current use/proposals of ML for climate simulations as well as implementation, training and evaluation of the models we devise on our own.

Interactive Exploration of 3D Models:

Steven Mills

We have a number of ways to easily make 3D models of objects from photographs or other data, but exploring these in 2D on a screen is somewhat limited. The goal of this project is to create a flexible application where people can use mobile devices to explore models of objects in 3D. The project involves the development of an Augmented Reality (AR) application (likely using Unreal Engine and C++, although Unity and C# are a viable alternative), and an evaluation of its effectiveness. The software developed in this project should take as input a 3D model of an object of interest and produce a suitable image target (e.g. a rendering of the object from a clear viewing angle). When the mobile application is pointed at this target, the user should see a 3D rendering of the model from their current viewpoint. As they move their device around the viewpoint will change, allowing them to explore the object in detail.