**Argflow: A Toolkit for Deep Argumentative Explanations for Neural Networks**

Adam Dejl, Peter He, Pranav Mangal, Hasan Mohsin, Bogdan Surdu, Eduard Voinea, Emanuele Albinì, Piyawat Lertvittayakumjorn, Antonio Rago, Francesca Toni

Department of Computing, Imperial College London

---

**Summary**

Neural network (NN) models are often difficult for human users to understand. We present Argflow, a toolkit enabling the generation of a variety of explanations for NN outputs in a classification setting using an argumentation-based approach called Deep Argumentative Explanation (DAXs).

**Deep Argumentative Explanations**

Given a neural network $\mathcal{N}$, we can generate a directed graph $(\mathcal{N}, I)$ which corresponds to how each neuron in $\mathcal{N}$ affects the output of other neurons in $\mathcal{N}$. In practice, $I$ may be a subset of neurons from $\mathcal{N}$. From this, we can derive a Generalised Argumentation Framework (GAF) by mapping each node to an argument with some strength and each edge to a type of argumentative relation (e.g. attack). Finally, we map different arguments to human-friendly visualisations using a function $\chi$ and present our GAF to users in some modality $\phi$. We refer to explanations created with this method as DAXs. A fuller treatment of DAXs can be found in [1].

---

**Example Explanations for VGG16**

**Baby or Diaper?**

We visualise the filters in the final convolutional layer of the network using the Grad-CAM algorithm [2] as our $\chi$ and a graph visualisation as our $\phi$. We can see the filters picking up parts of the baby when coming to the predicted classification ‘diaper’.

**Explanation Chatbots**

Using the same layer and $\chi$ as before, we generate an explanation for the network’s predicted classification of ‘tiger’, though this time picking a conversational interface as our $\phi$.

**Going Deeper**

We produce a deeper visualisation of the network by visualising filters from the tenth and final convolutional layers. We use activation maximisation as our $\chi$ and a graph visualisation as our $\phi$.

---

**Library**

Argflow comes in the form of a Python library and a web portal implemented using Python and React. The library computes explanations and sends them to a locally-running instance of the portal for display to end users. Both the library and portal are modular and highly extensible.

**Open Questions**

- What are the best choices of $\chi$, $\phi$ and the other mappings?
- How might we best integrate these DAXs into real-world applications?
- How might we extend DAXs to recurrent neural networks or transformers?

**Links**

Code: [https://gitlab.com/argflow](https://gitlab.com/argflow)

Demo video: [https://youtu.be/LPz4QbmLaxs](https://youtu.be/LPz4QbmLaxs)

Correspondence: ph1718@imperial.ac.uk

---
