DAX: Deep Argumentative eXplanation for Neural Networks

Emanuele Albini, Piyawat Lertvittayakumjorn, Antonio Rago and Francesca Toni
Department of Computing, Imperial College London, UK
{emanuele, pl1515, a.rago, ft}@imperial.ac.uk

Abstract

Despite the rapid growth in attention on eXplainable AI (XAI) of late, explanations in the literature provide little insight into the actual functioning of Neural Networks (NNs), significantly limiting their transparency. We propose a methodology for explaining NNs, providing transparency about their inner workings, by utilising computational argumentation (a form of symbolic AI offering reasoning abstractions for a variety of settings where opinions matter) as the scaffolding underpinning *Deep Argumentative eXplanations (DAXs)*. We define three DAX instantiations (for various neural architectures and tasks) and evaluate them empirically in terms of stability, computational cost, and importance of depth. We also conduct human experiments with DAXs for text classification models, indicating that they are comprehensible to humans and align with their judgement, while also being competitive, in terms of user acceptance, with existing approaches to XAI that also have an argumentative spirit.

1 Introduction

A considerable amount of recent efforts in AI are being devoted to the extraction of explanations for outputs, in particular of black-box methods such as deep learning. However, the majority of explanation methods in the literature (e.g., [Ribeiro *et al.*, 2016; Lundberg and Lee, 2017; Schwab and Karlen, 2019]) focus on describing the role of inputs (e.g., features) towards outputs. A consequence of such "flat" structure of explanations is very little transparency on how the outputs are obtained: explanations for the outputs of very different methods, e.g., Convolutional Neural Networks (CNNs) and Bayesian Network Classifiers, may amount to similar (or possibly even identical) pivotal features, when in fact the methods themselves function in vastly disparate manners. This focus on the "flat", functional, input-output behaviour also disregards the human perspective on explanations and the potential benefits for XAI from taking a social science viewpoint [Miller, 2019].

To address these issues when explaining predictions by Neural Networks (NNs), we propose *Deep Argumentative eXplanations (DAXs)*, based on concepts drawn from the field of Computational Argumentation (CA) in symbolic AI (briefly described below, see [Atkinson *et al.*, 2017; Baroni *et al.*, 2018a] for recent thorough overviews). Argumentative models of explanation are advocated in the social sciences, in particular in [Antaki and Leudar, 1992], as being suitable for humans. Indeed, the explanations generated by some existing methods have an argumentative spirit, e.g., LIME [Ribeiro *et al.*, 2016] and SHAP [Lundberg and Lee, 2017] assign to input features importance scores towards outputs, giving a form of weighted pro and con evidence exposing conflicting information underlying predictions.

CA is particularly suitable for underpinning DAXs. It is a mature field with strong foundations, offering a variety of *argumentation frameworks* comprising sets of arguments and *dialectical relations* between them (e.g., of *attack*, as in [Dung, 1995], and, in addition [Cayrol and Lagasquie-Schiex, 2005] or instead [Amgoud and Ben-Naim, 2016], of *support*). In these frameworks, arguments may be chained by dialectical relations forming deep (debate) structures. In some such frameworks, starting from [Dung, 1995], anything may amount to an argument, so long as it is in dialectical relation with other arguments (so, for example, strategies in games can be seen as arguments [Dung, 1995], and, in our DAXs, neurons and groups thereof in NNs may be understood as arguments). Further, in CA, conflicts exposed in debates (i.e., argumentation frameworks) are resolved using so-called semantics, e.g., amounting to definitions of *dialectical strength* of arguments, satisfying desirable *dialectical properties* (such as that attacks against an argument should decrease its dialectical strength). These properties may be studied a-posteriori, for given argumentation frameworks and semantics, e.g., as in [Baroni *et al.*, 2018b; Baroni *et al.*, 2019], or used a-priori, to obtain argumentation frameworks for which a semantics (given up-front) is "dialectically meaningful" according to the properties, e.g., as in [Rago *et al.*, 2018].

DAXs follow the latter approach, using, as gradual semantics, quantitative measures derived from the underlying NNs, extracting (instances of) *Generalised Argumentation Frameworks* (GAFs) with any number of dialectical relations obtained from connections between (groups of) neurons in the underlying NNs which are relevant to the prediction to be explained. The extracted GAFs are governed by desirable dialectical properties identified up-front to fit the setting of deployment for the given NNs/tasks. The GAFs are not themselves explanations, but are intended instead as abstractions of the underlying NNs from which explanations (i.e., DAXs) may be extracted, of potentially very different formats depending on the setting of deployment. Figure 1 (discussed later) gives a simple illustration of the DAX methodology.

In summary, our contribution in this paper is threefold:

- We instantiate the DAX methodology in three settings: with CNNs for text classification, with feed-forward NNs (FFNNs) for prediction with tabular data, and with CNNs for image classification. All three instantiations are of depth 3, namely they explain outputs in terms of intermediate nodes (e.g., features or filters), in turn explained in terms of inputs (but note that in principle, DAXs may be of any depth).
- In all three settings, we show that depth of explanation is important and that DAXs are stable [Sokol and Flach, 2020], i.e., every time the same input-output pair needs to be explained (by a fixed trained NN) the same explanation is generated. Further, after initial set-up, DAXs can be generated efficiently in the proposed instantiations.
- In the text classification setting, we also conduct experiments with humans showing DAXs' comprehensibility to humans and alignment with their judgement, as well as, in comparison with some existing explanation methods (i.e., LIME and SHAP), their increased ease of understanding, insightfulness, capability of inspiring trust and potential for visual appeal.

Although several works exist for argumentation-based explanation, e.g., recently [Rago *et al.*, 2018], to the best of our knowledge DAX are the first for NNs.

2 Background and Related Work

For the purposes of this paper, a (trained) Neural Network (NN) is a directed graph $\langle V, E\rangle$ with a set V of neurons and a set E of directed edges between neurons. Different neural architectures impose different restrictions on the graph. Each neuron in the graph is equipped with an activation value (computed by some activation function) and each link is equipped with a connection weight (learnt during training).

Our DAXs are obtained from Generalised Argumentation Frameworks (GAFs) (e.g., as understood in [Gabbay, 2016; Baroni *et al.*, 2017]), i.e., tuples $\langle \mathcal{A}, \mathcal{R}_1, \ldots, \mathcal{R}_m\rangle$ with $\mathcal{A}$, a set (of *arguments*), $m \geq 1$ and, $\forall i \in \{1,\ldots,m\}$, $\mathcal{R}_i \subseteq \mathcal{A}\times\mathcal{A}$ (binary, directed *dialectical relations*). Intuitively, GAFs are abstractions of debates, with arguments representing opinions and dialectical relations expressing various forms of agreement (support) or disagreement (attack) between opinions. GAFs can also be seen as directed graphs, with arguments as nodes and dialectical relations as labelled edges. Various instances of GAFs have been studied for specific choices of dialectical relations. In this paper, we will use: *support argumentation frameworks* (SAFs) [Amgoud and Ben-Naim, 2016], where $m=1$ and $\mathcal{R}_1$ is support; *bipolar argumentation frameworks* (BAFs) [Cayrol and Lagasquie-Schiex, 2005] where $m=2$, $\mathcal{R}_1$ is attack, $\mathcal{R}_2$ is support; and a form of *tripolar argumentation frameworks* (TAFs), where $m=3$, $\mathcal{R}_1$ is attack and $\mathcal{R}_2, \mathcal{R}_3$ are two different forms of support (potential and critical, to support counterfactual reasoning as in [Albini *et al.*, 2020]). The GAF in Figure 1ii is a BAF, visualised as a graph with nodes labelled by arguments and edges labelled by dialectical relations.

We will also use the notion of gradual semantics [Baroni *et al.*, 2018b; Baroni *et al.*, 2019], given by mappings $\sigma : \mathcal{A} \to \mathcal{V}$ for evaluating arguments' *dialectical strength* over a set of values $\mathcal{V}$ (in this paper $\mathcal{V} = \mathbb{R}$ and σ is extracted from trained NNs). Finally, we will use *dialectical properties* [Baroni *et al.*, 2018b; Baroni *et al.*, 2019] that σ needs to satisfy to ensure that the GAFs have natural features of human debates (e.g., that attacks weaken arguments).

A plethora of explanation methods are currently available in the literature (e.g., see [Guidotti *et al.*, 2019] for a recent survey), some model-agnostic (e.g., LIME [Ribeiro *et al.*, 2016], SHAP [Lundberg and Lee, 2017], CXPlain [Schwab and Karlen, 2019]), others tailored to specific methods (e.g., for NNs, Layer-wise Relevance Propagation (LRP) [Bach *et al.*, 2015], contrastive explanations [Dhurandhar *et al.*, 2018], and Grad-CAM [Selvaraju *et al.*, 2020]). The explanations generated by some of these methods have an argumentative spirit, e.g., LIME and SHAP assign to input features importance scores towards outputs, giving a form of weighted pro and con evidence. However, all these methods generate "flat" explanations, explaining outputs in terms of input features without any indication of how they are generated. We will compare DAXs with LIME and SHAP in our human experiments, and will use LRP and Grad-CAM to compute dialectical strength in our proposed DAX instaniations.

Some explanation methods, like our DAXs, make use of symbolic representations: whereas we use GAFs, others use, for example, influences between variables [Albini *et al.*, 2020], ordered decision diagrams [Shih *et al.*, 2018] and logical representations [Ignatiev *et al.*, 2019]. These other methods also focus on input-outputs only and are therefore "flat".

Computational argumentation is used by many for explanation purposes (e.g., recently in [Naveed *et al.*, 2018; Rago *et al.*, 2018; Cyras *et al.*, 2019; Madumal *et al.*, 2019]). Various neural architectures have also been used for argument mining from text [Lippi and Torroni, 2016; Cabrio and Villata, 2018; Lawrence and Reed, 2019], in some cases to extract GAFs [Cocarascu *et al.*, 2020], but not for the purposes of explanation. However, to the best of our knowledge, argumentation has not been used to explain the outputs of NNs until now.

Some recent methods generate "structured" explanations. These include *hierarchical explanations* [Chen *et al.*, 2020], *multilevel explanations* [Ramamurthy *et al.*, 2020] and explanations exploiting dependencies [Aas *et al.*, 2019]. These approaches incorporate interactions between input features, rather than from hidden layers, so they can also be deemed "flat" in our sense.

Some works, like DAXs, move away from "flat" explanations, predominantly with images: in [Wang and Nasconcelos, 2019] regions of an image generating insecurities between two outputs are highlighted, thus unearthing some of the "deliberative" reasoning behind the outputs (but still focusing

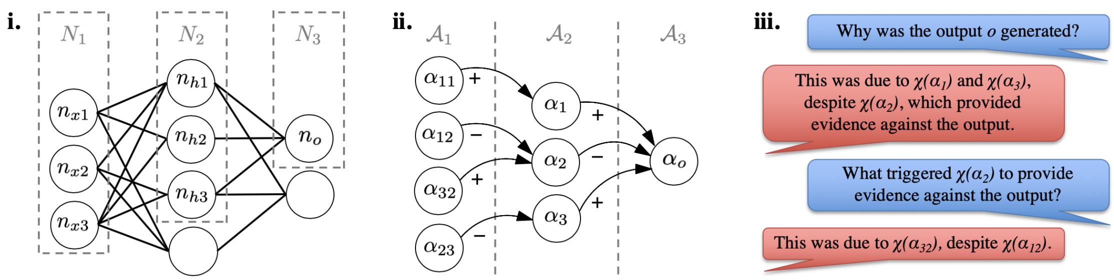

Figure 1: (i) Toy FFNN, with input/hidden/output neurons selected as nodes labelled with subscript $x/h/o$, resp., and layers N_1, N_2, N_3 indicated; (ii) extracted GAF with dialectical relations of attack (indicated with −) and support (indicated with +); and (iii) example DAX generated from the GAF (with a conversational format, comprising user (right) and system (left) statements). Here, $\chi(\alpha)$ (for α any argument in the GAF) indicates a representation for α, amenable to user consumption.

on input features); Network Dissection [Bau *et al.*, 2017] identifies regions of an image activating certain filters in a CNN, giving these filters a semantic meaning; Self-Explaining NNs [Alvarez-Melis and Jaakkola, 2018] use one NN for classification and another for explanation, to obtain explanations based on higher-level features/concepts and their relevance towards a classification; in [Olah *et al.*, 2018] positive/negative connections between concepts are extracted from neurons in a CNN; and in [Mu and Andreas, 2020], input patterns that strongly fire each neuron are represented by compositional logical concepts (considering natural language inference in addition to images). There are promising steps, but focus on visualisation, falling short of a general methodology for explanations showing the dialectical interplay amongst inputs, intermediate layers and outputs.

A user-driven perspective on explanations is advocated by many [Miller, 2019], as is the fact that different explanation styles, as envisaged by DAX, may need to be considered to aid transparency and trust and to support human decision making (e.g. see [Rader *et al.*, 2018; Lertvittayakumjorn and Toni, 2019; Kunkel *et al.*, 2019]).

3 DAX General Methodology

In this section, we outline our general methodology for explaining NN predictions. We will instantiate this methodology in Section 4 for CNNs and FFNNs for classification with text, images and tabular data, but, for the purposes of this section, we will assume as given a generic, trained NN $\mathcal{N}$ for prediction (amounting to a directed graph $\langle V, E\rangle$, as discussed in Section 2). We will also assume that explanations are needed for an output o computed by the given NN for some given input x, and that o is a single classification, e.g., the most probable class in the output layer of $\mathcal{N}$. For the purpose of explaining o for x, $\mathcal{N}$ can also be seen as a function f such that $f(x) = o$. DAXs are constructed in 3 steps, as indicated in Figure 2. Below we describe these steps in general, identifying the choices (hyper-parameters) that need to be instantiated when DAXs are deployed. We use the toy example in Figure 1 throughout the section for illustration.

Step 1 amounts to determining, within $\mathcal{N}{=}\langle V, E\rangle$, an *influence graph* $\langle N, \mathcal{I}\rangle$ with a set N of nodes and a set $\mathcal{I}$ of influences as candidate elements of (binary) dialectical relations of the GAF underlying explanations for why $f(x) = o$. The choice of N is a hyper-parameter dictated by the setting of deployment: we will see that in some DAX instantiations nodes in N are neurons (as is the case in Figure 1i); in others they are groups of neurons.[1] In all instantiations of our methodology in this paper, we choose $N = N_1 \cup \ldots \cup N_k$, for $k > 2$, with $N_k = \{n_o\}$ and $N_1, \ldots, N_k$ disjoint sets of nodes such that for every $i \in \{1, \ldots, k-1\}$ and for every $n_1 \in N_i$, $\exists n_2 \in N_{i+1}$ such that there is a path[2] via E from (neurons in) n_1 to (neurons in) n_2. Thus, $N_1, \ldots, N_k$ amount to *strata*, contributing (in steps 2 and 3) to *deep* explanations, where (arguments representing) nodes in each stratum are explained in terms of (arguments representing) nodes in the previous stratum. For illustration, in Figure 1i nodes are neurons and we have three strata, with neurons without a path to n_o not included in the strata.[3]

Once N has been chosen, we obtain $\mathcal{I}$ as $\{(n_1, n_2) | n_1, n_2 \in N$ and there is a path in (V, E) from (neurons in) n_1 to (neurons in) $n_2\}$.

Step 2 amounts to extracting a GAF $\langle \mathcal{A}, \mathcal{R}_1, \ldots, \mathcal{R}_m\rangle$ from $\langle N, \mathcal{I}\rangle$, with arguments drawn from the nodes in N and dialectical relations drawn from the influences in $\mathcal{I}$ (see Figure 1ii). The extraction of arguments is driven by a (hyper-parameter) mapping $\rho : \mathcal{A} \longrightarrow N$ such that $\mathcal{A} = \mathcal{A}_1 \cup \ldots \cup \mathcal{A}_k$ and for each $\alpha \in \mathcal{A}_i$ there is exactly one $n \in N_i$ with $\rho(\alpha) = n$ (we say that α represents n). The extraction of $\mathcal{R}_j$ is driven by a (hyper-parameter) *relation characterisation* $c_j : \mathcal{I} \longrightarrow \{true, false\}$ for *relation type* t_j, with $j \in \{1, \ldots, m\}$. For example, for Figure 1, dialectical relations of types support (+) and attack (-) may be defined by relation characterisations c_+ and c_- such that $c_+(n_1, n_2)$ is *true*[4] iff $w_{12}a_1 > 0$ (< 0 for $c_-(n_1, n_2)$), where a_1 is n_1's activation and w_{12} is the connection weight between n_1 and n_2 in $\mathcal{N}$. Then, in the extracted GAF, an argument representing

[1]Note that, even when nodes in N are neurons, in principle N may be exactly V but in practice it may be considerably smaller, to accommodate cognitive constraints of users.

[2]There exists a *path* via E (set of edges) from n_a to n_b (from a node to another) iff $\exists n_1, \ldots, n_t$ with $n_1 = n_a$ and $n_t = n_b$ such that $(n_1, n_2), \ldots, (n_{t-1}, n_t) \in E$

[3]Note that, in general, strata may result from non-adjacent layers in $\mathcal{N}$.

[4]With an abuse of notation, for simplicity we write $c_j(a, b)$ instead of $c_j((a, b))$ throughout, for any j.

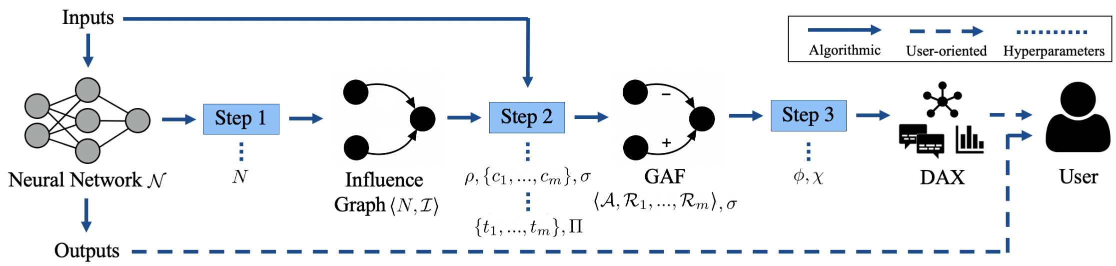

Figure 2: DAX methodology (alongside the typical process of obtaining outputs from a neural model given its inputs) comprising steps: 1. Based on the chosen *nodes* N within $\mathcal{N}$, extract $\langle N, \mathcal{I} \rangle$, a directed graph of *influences* between nodes; 2. Extract a *GAF* from the output of the first step, based on choices of *argument mapping* ρ, *relation characterisations* $\{c_1, ..., c_m\}$ and *dialectical strength* σ. These choices are driven by the types of relations $\{t_1, ..., t_m\}$ to be extracted and the*dialectical properties* Π that σ should satisfy (on the GAF) to form the basis for explanations; 3. Generate a *DAX* from the GAF and σ, for user consumption in a certain *format* ϕ associating arguments with human-interpretable concepts through a *mapping* χ.

neuron n_1 supports an argument representing neuron n_2 iff $c_+(n_1, n_2) = true$ (similarly for attack and c_-). Note that we call the relation characterised by c_+ 'support' (and by c_- 'attack') as it is defined in terms of agreement (disagreement, resp.) between nodes and thus carries positive (negative, resp.) influence; the relations are thus dialectical.

Another choice (hyper-parameter) concerns the notion of (dialectical) strength $\sigma : \mathcal{A} \to \mathbb{R}$ as a quantitative measure derived from $\mathcal{N}$ and giving dialectical meaning to the arguments in $\mathcal{A}$. In the running example, we could choose to set $\sigma(\alpha) = a$ with a being the modulus of the activation of $\rho(\alpha)$.

The two choices of relation characterisations and strength are guided by the further choice of a set of (desirable) dialectical properties Π that σ needs to satisfy in the extracted GAF, determining how "natural" the DAX obtained at step 3 will be. The use of Π to constrain the choice of σ is a distinguishing factor between our DAXs and work on NN visualisation.

For illustration, a natural candidate for inclusion in Π in the running example is *dialectical monotonicity* [Baroni *et al.*, 2018b; Baroni *et al.*, 2019], requiring that the strengthening of any argument results in the weakening (strengthening) of any argument it attacks (supports, resp.). Here, with $\sigma(\alpha)$ given by the modulus of $\rho(\alpha)$'s activation, this property equates to the requirement that increasing the magnitude of a neuron's activation results in a decrease (an increase) in magnitude of the activation of any neuron it attacks (supports, resp.) – and it is thus desirable for explanation of outputs in terms of attacks and supports. In general, dialectical monotonitity can hold with any activation function; in the running example it can be seen by inspection that it holds if *tanh* is used.

Several concrete choices may be possible and natural at this step, depending on the setting of deployment and the underlying choice of NN, as we shall see in Section 4. In general, we require (minimally) that they result in a GAF which (when seen as a graph with edges of m different kinds) is a tree with the explained node's argument at the root (see Figure 1ii for illustration and Appendix A for formal details). In particular, this may give that (conceptually) multiple arguments represent the same node (as in the running example). Note also that, to secure "relevance" of explanations drawn from the GAF, nodes which are not connected by influences do not contribute to dialectical relations in the GAF and nodes dialectically disconnected from the output node are not represented as arguments in the GAF.

Step 3 amounts to extracting, from the GAF and notion of strength from step 2, a suitable DAX for the intended users. While the information content for DAX is harboured in the GAF and accompanying strength, different *formats* ϕ may suit different users' or settings' explanatory requirements. The hyper-parameter ϕ captures both the choice of fragments of the GAF contributing to explanations and the way in which these fragments are shown to users. In the simple running example, we have chosen ϕ amounting to use the full GAF within a *conversational* setting with the user (see Figure 1iii), as advocated, for example, in [Cyras *et al.*, 2019; Cocarascu *et al.*, 2019; Balog *et al.*, 2019], but we can in principle accommodate various other formats, for example choosing the GAF fragment with strongest supporters and attackers only, at every stratum, and a graphical presentation (as in Section 4). The focus on a fragment of the GAF matches other explanation methods, e.g., LIME, where only some features may be included to ensure user comprehensibility. We envisage that the size of the fragment will depend on the needs of users, e.g., expert developers may benefit from larger fragments than lay users. In many cases, the fragment chosen by ϕ will be considerably smaller than the GAF, especially when the size of the GAF is prohibitively large (as in the DAX instances we consider in Section 4, where GAFs include as many as 1,472,224 arguments, see Appendix B).

DAXs also require appropriate *interpretations* of intermediate arguments, returned by a suitably defined (hyper-parameter) mapping χ, left unspecified in Figure 1iii (see various choices for χ in Section 4). The definition of this mapping may benefit from work on explanation-as-visualisation (e.g., as in [Olah *et al.*, 2017; Bau *et al.*, 2017] for images and as in [Lertvittayakumjorn *et al.*, 2020] for text). Please note however that, for DAXs, this interpretation only plays a role in the last step, on top of the GAF obtained from the first two.

4 DAX Instantiations

We applied the general DAX methodology to three specific neural architectures for prediction, trained on specific

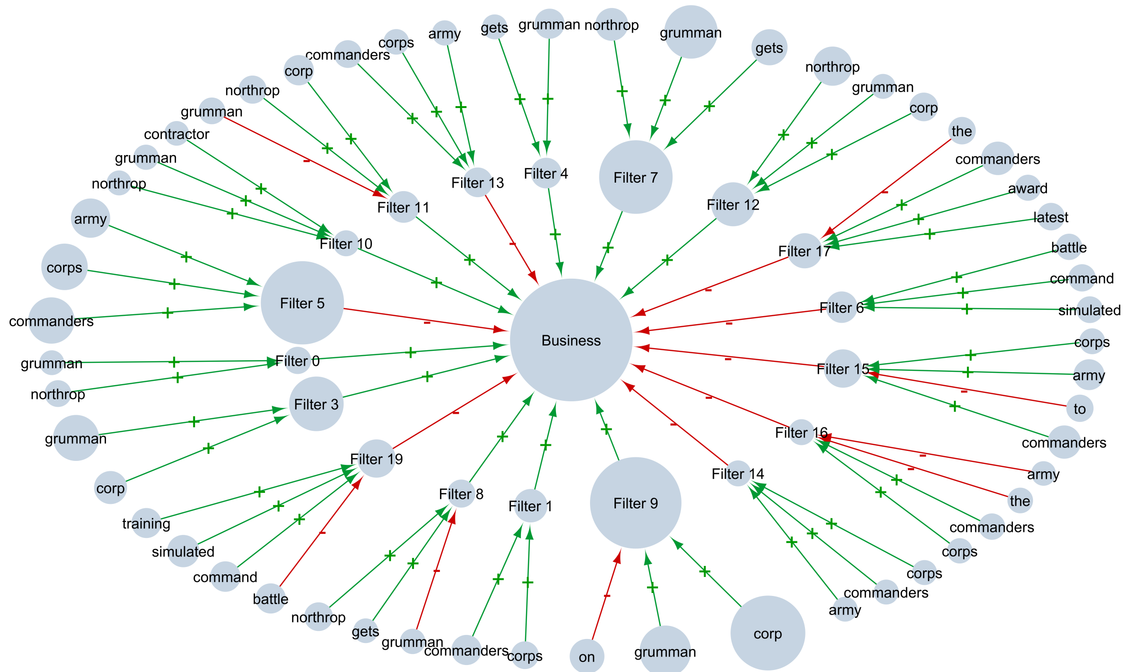

Figure 3: BAF from which the DAX for text CNN in Figure 4 is obtained. The bubbles are arguments $\alpha \in \mathcal{A}$ with sizes corresponding to their dialectical strength $\sigma(\alpha)$ and as label their mapping to nodes $\rho(\alpha)$. Green and red arrows indicate, resp., support and attack.

datasets/tasks as indicated. We give details for the first, and sketch the other two (additional details for the three instances can be found in Appendix B).

4.1 DAX for CNNs for Text Classification

We targeted a CNN architecture [Kim, 2014] with an input layer of 150 words followed by: an embeddings layer (6B GloVe [Pennington *et al.*, 2014] embeddings of size 300); a hidden layer of 20 1D ReLU convolutional filters of which 10 are of filter size 3, 5 are of size 2 and 5 are of size 4, followed by a max pooling layer; and finally a dense softmax layer. We trained this architecture with two datasets: a pre-processed version of *AG-News* [Gulli, 2005] (without HTML characters and punctuation) and *IMDB* [Maas *et al.*, 2011]. For both datasets, sentences were tokenized using the *spaCy* tokenizer. In the remainder of Section 4.1 we assume as given $\mathcal{N}$ for either dataset.

Step 1. We chose 3 strata: an input stratum N_1 with nodes corresponding to the input words ($|N_1| = 150$), an intermediate stratum N_2 with nodes corresponding to the neurons of the max-pooling layer ($|N_2| = 20$) and an output stratum $N_3 = \{n_o\}$ where n_o is the neuron of the most probable class. Influences were then obtained as defined in Section 3.

Step 2. We extracted a BAF $\langle \mathcal{A}, \mathcal{R}^-, \mathcal{R}^+ \rangle$ with dialectical relations $\mathcal{R}^-$ and $\mathcal{R}^+$ of types attack (i.e., negative influence) and support (i.e., positive influence, resp.). $\mathcal{A}$ is divided in 3 disjoint sets, matching the strata: $\mathcal{A}_3 = \{\alpha_o\}$, with α_o the *output argument*; $\mathcal{A}_2$ is the set of all *intermediate arguments* α_j representing filters n_j in the second stratum; and $\mathcal{A}_1$ is the set of all *input arguments* α_{ij}, each representing a word n_i in the first stratum that influences the filter n_j in the second stratum. The mapping from arguments to nodes is thus defined, trivially, as $\rho(\alpha) = n_i \in N_1$ if $\alpha = \alpha_{ij} \in \mathcal{A}_1$, $\rho(\alpha) = n_j \in N_2$ if $\alpha = \alpha_j \in \mathcal{A}_2$ and $\rho(\alpha) = n_o$ if $\alpha = \alpha_o \in \mathcal{A}_3$. Even though in this DAX instantiation we chose to use the same types as in the toy illustration in Section 3, in this setting we based the relation characterisations c_+, c_- on LRP [Bach *et al.*, 2015]. Let $R(j,i)$ stand for LRP *relevance* back-propagated from n_j back to n_i[5]. Then $c_-(n_i, n_j) = true$ iff $R(j,i) < 0$ and $c_+(n_i, n_j) = true$ iff $R(j,i) > 0$. The dialectical strength σ is also defined in terms of LRP in this instance: for a_p the activation of neuron n_p, $\sigma(\alpha) = a_o$ if $\alpha = \alpha_o \in \mathcal{A}_3$, $\sigma(\alpha) = |R(o,j)|$ if $\alpha = \alpha_j \in \mathcal{A}_2$ and $\sigma(\alpha) = \left| R(j,i) \cdot \frac{R(o,j)}{a_j} \right|$ if $\alpha = \alpha_{ij} \in \mathcal{A}_1$. In the choice of c_-, c_+, and σ, we were driven by the properties of *dialectical monotonicity* from Section 3 and of *additive monotonicity*, sanctioning that the strength of an argument amounts to the sum of the strengths of its supporting arguments and the negations of the strengths of its attacking arguments. Formally, σ satisfies this property iff, for any $\alpha \in \mathcal{A}$: $\sigma(\alpha) = \sum_{(\beta,\alpha) \in \mathcal{R}^+} \sigma(\beta) - \sum_{(\beta,\alpha) \in \mathcal{R}^-} \sigma(\beta)$. These properties led to a BAF aligning with human judgement, as we will show in Section 6. The BAFs extracted at this step may be quite large. For example, Figure 3 shows an example of such BAF, with 73 arguments. Note that, here, the filters, without the presentation by χ, are black-boxes

[5] If n_i and/or n_j are groups of neurons, the relevance back-propagated from node n_j to node n_i is the sum of the relevances back-propagated from every neuron in n_j to every neuron in n_i.

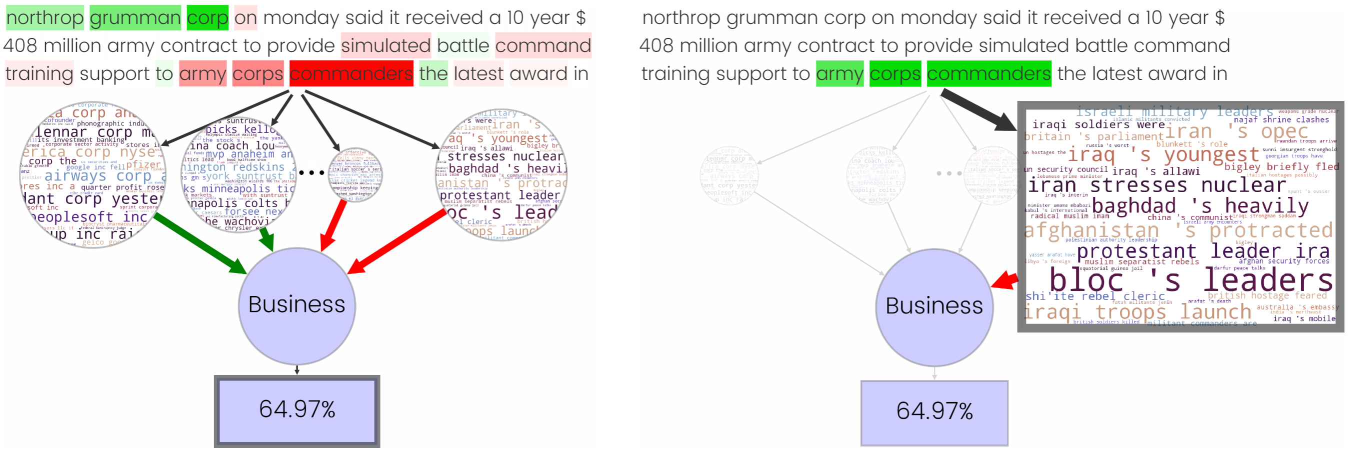

Figure 4: (Left) Graphical DAX for the (correct) output class "Business" (determined with 64.97% probability) by the CNN for text classification with AG-News from Section 4.1, for the given input article from AG-News at the top. (Right) The view of the DAX after clicking on the rightmost word cloud. For this DAX, arguments are on 3 levels, visualised (by χ) as follows: *input words* with, on the left, highlighting of a word given by the sum of the (dialectical) strengths of all arguments representing that word and, on the right, highlighting of a word given by its corresponding argument's dialectical strength; *convolutional filters/word clouds* with sizes representing the corresponding arguments' strengths; and the most probable *class* and its probability. The green/red colours indicate, resp., support/attack (in both images, colours are used on edges/relations from intermediate to output arguments, and on words from input to intermediate arguments).

unlikely to appeal to users; indeed, as mentioned in Section 3, the GAFs from step 2 are not meant to be shown to users.

Step 3. We extracted a DAX of *graphical* format (ϕ) from the BAF, as shown (for a specific input text and relative CNN-computed output, with $\mathcal{N}$ trained on AG-News) in Figure 4. This DAX has 3 levels, aligning with the BAF. Since convolutional filters/intermediate arguments are not intrinsically human-comprehensible, we opted (in the choice of mapping χ) to pair them with word clouds showing n-grams from the training set that activate the most the corresponding filter, as in [Lertvittayakumjorn *et al.*, 2020]. We also chose to visualise attacks and supports from the intermediate arguments with differently coloured arrows, and the attacks and supports from the input to the intermediate arguments as arrows originating from differently coloured words. We chose the format of this DAX to be *interactive*, allowing users to focus on particular parts, e.g., when a user clicks on a word cloud, it is magnified and the single words (arguments) supporting or attacking the filter are highlighted. Interactivity is particularly useful with deep explanations, to help users control the amount of information they receive.

4.2 DAX for CNN for Image Classification

To assess DAX's applicability to deeper NNs, we considered a version of *VGG16* [Simonyan and Zisserman, 2015] pre-trained on *ImageNet* [Deng *et al.*, 2009], with 1000 classes. This NN comprises 5 blocks of convolutional layers, a max pooling layer, and 3 fully connected layers.

Step 1. To avoid explanations that may overwhelm users with information, we chose to 'promote' only 3 of the NN's layers to strata: an input stratum N_1 such that $n_{xy} \in N_1$ represents the x, y pixel in the input image[6] ($|N_1| = 224 \cdot 224$), an intermediate stratum N_2 such that node $n_j \in N_2$ corresponds to a filter in the last convolutional layer[7] ($|N_2| = 512$), and an output stratum $N_3 = \{n_o\}$ ($|N_3| = 1$) where n_o is the neuron of the most probable (output) class.

Step 2. We chose a GAF with a single relation of support (SAF, see Section 2), and used Grad-CAM weighted forward activation maps [Selvaraju *et al.*, 2020] to define relation characterisations and dialectical strength. The focus on the support dialectical relation only (and the exclusion of attack) is in line with Grad-CAM, whose focus is on features *positively* influencing predicted classes.

Step 3. We obtained (from the SAF) a *graphical* DAX on 3 levels: the input image, the intermediate strata's filters and the predicted class node. In the spirit of the text CNN, to give a human-comprehensible meaning to the intermediate arguments, we defined χ in terms of feature visualisation, based here on activation maximisation [Olah *et al.*, 2017; Kotikalapudi, 2017; Mahendran and Vedaldi, 2015; Mordvintsev *et al.*, 2015].

4.3 DAX for FFNN for binary classification with tabular data

To explore a broader spectrum of dialectical relations (including a third relation of *strict support* of a counterfactual nature), we deployed an FFNN for classification with the categorical tabular data in the COMPAS dataset [ProRepublica, 2016], understanding *two_year_recid* as binary prediction (see Appendix B.3 for details). The FFNN has 3 layers: an input layer that takes the one-hot encoding of the inputs, a dense hidden layer of size 8 with *tanh* activation followed by *ReLU* and an output layer with *sigmoid* activation.

[6] Here each node is a collection of the 3 neurons corresponding to the 3 RGB channels of the pixel.

[7] Here each node is a collection of 14x14 neurons corresponding to a filter in the last convolutional layer.

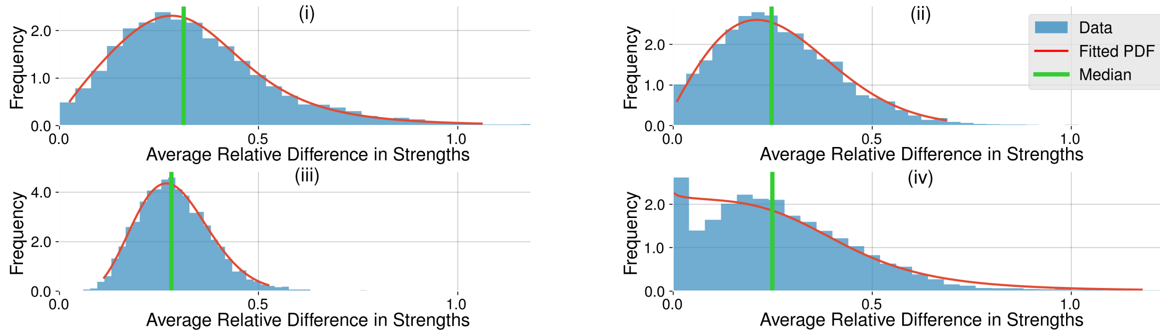

Figure 5: Distributions of difference of the intermediate arguments' strengths, i.e., $\frac{2\cdot||\boldsymbol{\sigma}'-\boldsymbol{\sigma}||_1}{||\boldsymbol{\sigma}'||_1+||\boldsymbol{\sigma}||_1}$ where $\boldsymbol{\sigma}$ and $\boldsymbol{\sigma'}$ are resp. the strengths' vectors of the arguments extracted from the original input and the similar one. We show results for the (NNs for) IMDB (i), AG-News (ii), a sample of ImageNet (iii), and COMPAS (iv).

Step 1. Each layer coincides with a stratum, giving N_1 ($|N_1| = 58$) with $n_f \in N_1$ corresponding to input neuron f, N_2 such that $n_j \in N_2$ is the node corresponding to the hidden neuron j after (tanh and) ReLU activation ($|N_2| = 8$), and $N_3 = \{n_o\}$ with n_o the most probable class node.

Step 2. The GAF is a TAF $\langle \mathcal{A}, \mathcal{R}^-, \mathcal{R}^+, \mathcal{R}^{!+} \rangle$ with dialectical relations, resp., of attack, support and *critical support* (i.e., an essential positive influence). Here, the drivers of the extraction of the TAF (dialectical strength and relations) are the activations. Intuitively, the relation characterisation $c_{!+}$ defines a critical support between arguments representing n_i and n_j iff deactivating n_i causes the deactivation of n_j (i.e., neuron n_i is not only positively influencing n_j, but its contribution is essential for its activation). The dialectical property chosen here is of *binary counter-factuality*, requiring that the removal of the critical support from an argument to another with positive dialectical strength will result in the strength of the latter becoming zero or lower.

Step 3. We chose a graphical DAX (obtained from the TAF) with 3 levels reflecting the strata/argument structure: (1) the input features (2) the hidden features (3) the prediction and its probability. To visualise the intermediate arguments (via χ), we generated a pie chart showing the features activating the most for each of the hidden neurons/arguments.

5 Empirical Evaluation

In this section, we conduct an empirical evaluation of our three proposed DAX instances, along two dimensions for evaluation of explanations (*stability* and *computational cost*) that are standard in the literature (e.g., see the overview in [Sokol and Flach, 2020]) as well as a third dimension (importance of depth) which is tailored to deep explanations such as DAXs.

Stability. All instantiations use deterministic methods (LRP, GradCAM, activations) which, given the same input/output pairs, return the same information. So, all GAFs are unique for each input-output pair, and DAXs are *stable* [Sokol and Flach, 2020].

Computational cost. Firstly note that, although the choice of $\langle N, \mathcal{I} \rangle$ is tailored to explaining $f(x)=o$, for specific input-output pairs (x,o), the choice of candidate nodes in N from the hidden layers can be made a-priori, independently of any pair. Also, the skeleton underpinning the construction of DAX relies exclusively on the given neural architecture, prior to training. Thus, there is a single one-off cost for constructing DAXs that can be shared across pairs. Secondly, the time complexity to calculate DAXs is relatively small: there is a one time cost for the mapping χ (from arguments), but the cost to generate a single DAX is comparable to the cost of a single prediction and single (for the FFNN) or multiple (for LRP and Grad-CAM) back-propagation steps. The time cost depends *linearly* on the size of N. In our experiments, the setup took up to 2.35 hours (for VGG16 activation maximization) while the time to generate a single DAX ranged from under $1ms$ (for the FFNN) to $4.35s$ (for VGG16) (see Appendix C for details).

Importance of depth. We analysed how the intermediate arguments' strengths change when providing the various NNs for our instantiated DAXs with samples with "similar" inputs that lead to "similar" outputs, where similarity is defined appropriately for the three settings. For text classification, we generated similar input pairs back-translating from French [Ribeiro *et al.*, 2018] using Google Translate; for images we paired each image with its noisy counterpart (with Gaussian noise of $std = 10$); for tabular data we paired each sample with another sample with one categorical feature changed. In all settings, we considered outputs to be similar when the predicted probability changed less than 5%. Figure 5 shows, for the considered datasets, the distribution of the average relative difference of the intermediate arguments' strengths. Note the significant average relative difference with the median of the difference for all the considered scenarios between 24.7% and 31.2%. This shows the importance of intermediate arguments given that, despite only minor changes in the input and output arguments, the intermediate arguments changed significantly.

6 Experimental Results

We conducted experiments with 72 human participants on Amazon Mechanical Turk, to assess: (A) whether DAXs are comprehensible to humans; (B) whether DAXs align with human judgement and (C) how desirable DAXs are when compared to other explanation methods with an argumentative spirit. We focused on text classification (see Section

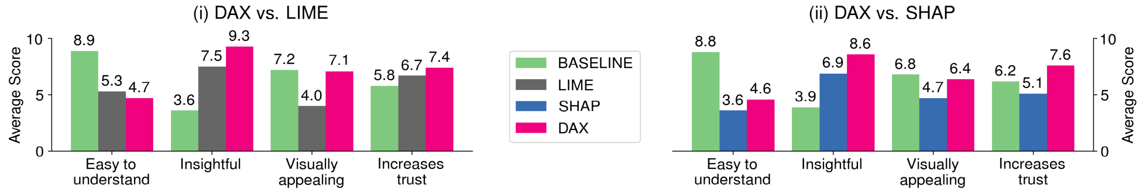

Figure 6: Participants average score amongst different criteria of acceptability for an explanation when comparing DAX with a baseline and LIME (i) or SHAP (ii). Scores were converted from a discrete 3-value scale ('No', 'Somewhat', 'Yes') to numerical scores (0, 5, 10) for ease of comprehension.

4.1) with AG-News. We chose this setting because topic classification (AG-News) is multiclass (with four classes) – thus more complex than (binary) sentiment analysis (IMDB); also, it requires no domain expertise (unlike COMPAS) and has not been widely studied (unlike ImageNet). The participants were non-expert (see Appendix D), showing that DAX' target audience may not be limited to experts.

In the experiments, we used only samples predicted with high confidence (i.e., samples in the test set whose prediction had a probability over 0.95); we did so as, for samples predicted with lower confidence, if the explanations are implausible to users, we cannot know whether this is due to the poor underlying model or the explanations themselves. Given the activations of the convolutional filters of 1000 random training samples, we considered a filter strongly activated if its activation was greater than the 90th percentile of its sampled activations, and weakly activated if its activation was lower than the 1st percentile. We considered a class to be strongly supported by an intermediate argument if the latter's strength ranked in the top 20% of the arguments associated with the class.

(A) Comprehensibility. The following research questions aim to ensure that humans can understand the interplay between intermediate arguments (represented by word clouds, see Figure 4) and other arguments (i.e., input words and output classes). **RQ1**: "Can humans understand the *roles* of individual intermediate arguments towards the output by examining their word clouds?" To answer RQ1, we showed participants a word cloud interpreting an intermediate argument and asked them to select the class they best associated with it. ***Results:*** Over 97% of answers ($p << 0.001$) picked a class strongly supported by the presented argument, confirming human understanding. **RQ2**: "Can humans understand the *patterns* of intermediate arguments by examining their word clouds?" To answer RQ2, we asked participants to select from 4 n-grams that which best matched the pattern in the word cloud of a strongly activated filter. The 4 n-grams were extracted from input texts correctly predicted with high confidence by the CNN, and the CNN's best-match was selected by the max pooling, while the others weakly activated the filter. ***Results:*** Participants answered correctly in 65.5% of the cases. We posit this result as positive as random guesses can lead to 25% correct answers and the task is not trivial without NLP experience, as was the case for 96% of participants (see Figure 15 in Appendix D). **RQ3**: "Can humans understand the NN from the dialectical relations originating from the intermediate arguments only, without the input?" To answer RQ3, incrementally, we first showed only the strongest supporter and attacker of the predicted class, and then added the next strongest supporter and attacker, and so on until 4 of each were shown. At each step we asked the participants to select the class they would predict. ***Results:*** The strongest supporter and attacker sufficed for participants to correctly classify samples in over 80% of the cases. Thus, even partial information about the intermediate arguments gives strong insight into the CNN model.

(B) Alignment with Human Judgement. We assessed whether DAXs (especially support) align with human judgement when the CNN confidently and correctly predicts a certain class. **RQ4**: "Do supports *between intermediate arguments and predicted class* in DAX align with human judgement?" To answer RQ4, we showed n-grams extracted from input texts correctly predicted with high confidence and selected by a strongly activated filter (but without showing the corresponding word cloud) and asked participants to select the class with which this n-gram is best associated. ***Results:*** In over 96% of the answers ($p << 0.001$), participants selected a class that the n-gram strongly supports. **RQ5**: "Do supports *between individual words and intermediate arguments* in DAXs align with human judgement?" To answer RQ5, we showed n-grams extracted from input texts correctly predicted with high confidence of a strongly activated filter together with its word cloud and asked participants to select a word (in the n-grams) fitting best with phrases in the word cloud amongst 2 words: the strongest supporter and attacker of the intermediate argument representing the filter. ***Results:*** In 85% of the answers to RQ5, participants chose the correct answer ($p << 0.001$). This assured us that our notion of support aligns with human judgement.

(C) User Acceptance. To assess whether *deep* explanations, in the form of DAXs, are amenable to humans in comparison with "flat" explanations, we chose to compare DAXs and existing methods with a somewhat similar argumentative spirit as the BAF-based DAXs we have deployed for text classification (in particular, we chose the popular LIME and SHAP forms of explanation, presented in their standard formats within the available LIME and SHAP libraries). To compare DAXs and LIME/SHAP in terms of user acceptance, we asked participants to rate explanations according to (desirable) criteria: (1) ease of understanding; (2) ability to provide insight into the internal functioning of the CNN; (3) capability of inspiring trust in users; and (4) visual appeal. We showed each participant an example consisting of the input text, the predicted class with its confidence, and 3 explanations (varying the order in different experiments): a baseline (a graph showing the probability of each class), a DAX, and the explanation computed by LIME or SHAP. ***Results:*** (See Figure 6) As expected, DAXs give more insight on the CNN internal working when compared with LIME ($p < 0.004$) and SHAP ($p < 0.006$). DAXs are also significantly more visually appealing to users than LIME ($p << 0.001$) and SHAP ($p < 0.02$). This may be in part due to DAXs' use, in this instantiation, of world clouds to show filters, possible for DAXs showing the inner workings of the CNN but not for model-agnostic LIME and SHAP, focusing on the CNN's input-output behaviour only. Another possible reason for DAXs' visual appeal may be the interactivity of DAXs in the instantiation considered, which allows users to better explore how the underlying model predicts. We believe that, even if we could make LIME and SHAP interactive, they may not provide additional insight on the model due to their flat structure. Finally, DAXs significantly inspire more trust when compared with SHAP ($p < 0.002$) although the increase in trust is not significant when compared with LIME ($p = 0.32$). We posit that this is due to the lack of expertise of participants (less than 12% of participants stated that they had expertise in machine learning, NNs or explanation methods). We also posit that these benefits may be more significant with expert participants.

7 Conclusions

We presented novel Deep Argumentative Explanations (DAXs), resulting from a marriage between symbolic AI (in the form of computational argumentation) and neural approaches, and leveraging on the amenability of argumentation for explanation to humans and on several advances in the field of computational argumentation as well as in interpretability and visualisation of NNs. DAXs allow *deep* explanations for NN predictions to be generated, casting light on the inner working of the NN beyond visualisation and adaptable to fit a variety of explanatory settings. Open questions for future work include whether other forms of symbolic AI, e.g., rule-based, can equally well support the argumentative nature of explanations for NN predictions, the depth and the systematic versatility naturally supported by DAXs. We presented three DAX instantiations, for: CNNs for (two forms of) text classification, CNNs for image classification, and FFNNs for tabular data classification. These instantiations amount to novel stand-alone contributions per se that can be readily deployed in other tasks supported by the same neural architectures: future work includes exploring this further deployment. We conducted experiments, with lay users, to show that DAXs (for CNNs for text classification) exhibit desirable properties from the user perspective, in a stand-alone manner and in comparison with some existing explanation methods which have an argumentative spirit but are flat. As future work, it would be interesting to conduct further experiments with expert users (which may greatly benefit from DAXs, e.g. during model development) and/or with different DAX instances and tasks/data. Furthermore, we plan to explore how DAXs can support downstream tasks and for debugging models, e.g., along the lines of [Lertvittayakumjorn *et al.*, 2020] but using the varied spectrum of DAXs that our methodology affords.

Acknowledgements

This research was funded in part by J.P. Morgan and by the Royal Academy of Engineering under the Research Chairs and Senior Research Fellowships scheme. Any views or opinions expressed herein are solely those of the authors listed, and may differ from the views and opinions expressed by J.P. Morgan or its affiliates. This material is not a product of the Research Department of J.P. Morgan Securities LLC. This material should not be construed as an individual recommendation for any particular client and is not intended as a recommendation of particular securities, financial instruments or strategies for a particular client. This material does not constitute a solicitation or offer in any jurisdiction. We thank Kristijonas Cyras for helpful comments on earlier versions of this paper.

References

[Aas *et al.*, 2019] Kjersti Aas, Martin Jullum, and Anders Løland. Explaining individual predictions when features are dependent: More accurate approximations to shapley values. *CoRR*, abs/1903.10464, 2019.

[Albini *et al.*, 2020] Emanuele Albini, Antonio Rago, Pietro Baroni, and Francesca Toni. Relation-based counterfactual explanations for bayesian network classifiers. In *29th Int. J. Conf. on Artif. Intel., IJCAI 2020*, pages 451–457, 2020.

[Alvarez-Melis and Jaakkola, 2018] David Alvarez-Melis and Tommi S. Jaakkola. Towards robust interpretability with self-explaining neural networks. In *31st Conf. on Neural Information Processing Systems, NeurIPS*, pages 7786–7795, 2018.

[Amgoud and Ben-Naim, 2016] Leila Amgoud and Jonathan Ben-Naim. Evaluation of arguments from support relations: Axioms and semantics. In *25th Int. J. Conf. on Artif. Intel., IJCAI*, pages 900–906, 2016.

[Antaki and Leudar, 1992] Charles Antaki and Ivan Leudar. Explaining in conversation: Towards an argument model. *Europ. J. of Social Psychology*, 22:181—-194, 1992.

[Atkinson *et al.*, 2017] Katie Atkinson, Pietro Baroni, Massimiliano Giacomin, Anthony Hunter, Henry Prakken, Chris Reed, Guillermo Ricardo Simari, Matthias Thimm, and Serena Villata. Towards artif. argumentation. *AI Magazine*, 38(3):25–36, 2017.

[Bach *et al.*, 2015] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLOS ONE*, 10(7):1–46, 07 2015.

[Balog *et al.*, 2019] Krisztian Balog, Filip Radlinski, and Shushan Arakelyan. Transparent, scrutable and explainable user models for personalized recommendation. In *42nd Int. ACM SIGIR Conf. on Research and Development in Information Retrieval, SIGIR*, pages 265–274, 2019.

[Baroni *et al.*, 2017] Pietro Baroni, Giulia Comini, Antonio Rago, and Francesca Toni. Abstract games of argumentation strategy and game-theoretical argument strength. In *20th Int. Conf. on Principles and Practice of Multi-Agent Systems, PRIMA*, pages 403–419, 2017.

[Baroni *et al.*, 2018a] P. Baroni, D. Gabbay, and M. Giacomin. *Handbook of Formal Argumentation*. College Publications, 2018.

[Baroni *et al.*, 2018b] Pietro Baroni, Antonio Rago, and Francesca Toni. How many properties do we need for gradual argumentation? In *32nd AAAI Conf. on Artif. Intel.*, 2018.

[Baroni *et al.*, 2019] Pietro Baroni, Antonio Rago, and Francesca Toni. From fine-grained properties to broad principles for gradual argumentation: A principled spectrum. *Int. J. Approx. Reason.*, 105:252–286, 2019.

[Bau *et al.*, 2017] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *IEEE Conf. on Computer Vision and Pattern Recognition, CVPR*, pages 3319–3327, 2017.

[Cabrio and Villata, 2018] Elena Cabrio and Serena Villata. Five years of argument mining: a data-driven analysis. In *27th Int. J. Conf. on Artif. Intel., IJCAI*, pages 5427–5433, 2018.

[Cayrol and Lagasquie-Schiex, 2005] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. On the acceptability of arguments in bipolar argumentation frameworks. In *8th Europ. Conf. on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, pages 378–389, 2005.

[Chen *et al.*, 2020] Hanjie Chen, Guangtao Zheng, and Yangfeng Ji. Generating hierarchical explanations on text classification via feature interaction detection. *CoRR*, abs/2004.02015, 2020.

[Cocarascu *et al.*, 2019] Oana Cocarascu, Antonio Rago, and Francesca Toni. Extracting dialogical explanations for review aggregations with argumentative dialogical agents. In *18th Int. Conf. on Autonomous Agents and MultiAgent Systems, AAMAS*, page 1261–1269, 2019.

[Cocarascu *et al.*, 2020] Oana Cocarascu, Elena Cabrio, Serena Villata, and Francesca Toni. Dataset independent baselines for relation prediction in argument mining. In Henry Prakken, Stefano Bistarelli, Francesco Santini, and Carlo Taticchi, editors, *Computational Models of Argument - Proceedings of COMMA 2020*, volume 326 of *Frontiers in Artificial Intelligence and Applications*, pages 45–52. IOS Press, 2020.

[Cyras *et al.*, 2019] Kristijonas Cyras, Dimitrios Letsios, Ruth Misener, and Francesca Toni. Argumentation for explainable scheduling. In *The 33rd AAAI Conf. on Artif. Intel.*, pages 2752–2759, 2019.

[Deng *et al.*, 2009] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition CVPR*, pages 248–255, 2009.

[Dhurandhar *et al.*, 2018] Amit Dhurandhar, Pin-Yu Chen, Ronny Luss, Chun-Chen Tu, Pai-Shun Ting, Karthikeyan Shanmugam, and Payel Das. Explanations based on the missing: Towards contrastive explanations with pertinent negatives. In *31st Conf. on Neural Information Processing Systems, NeurIPS*, pages 590–601, 2018.

[Dung, 1995] Phan M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artif. Intel.*, 77(2):321 – 357, 1995.

[Gabbay, 2016] Dov M. Gabbay. Logical foundations for bipolar and tripolar argumentation networks: preliminary results. *J. Log. Comput.*, 26(1):247–292, 2016.

[Guidotti *et al.*, 2019] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. A survey of methods for explaining black box models. *ACM Comput. Surv.*, 51(5):93:1–93:42, 2019.

[Gulli, 2005] Antonio Gulli. AG-News Corpus, 2005.

[Ignatiev *et al.*, 2019] Alexey Ignatiev, Nina Narodytska, and João Marques-Silva. On relating explanations and adversarial examples. In *32nd Conf. on Neural Information Processing Systems, NeurIPS*, pages 15857–15867, 2019.

[Keras, 2020] Keras. Keras applications, 2020.

[Kim, 2014] Yoon Kim. Convolutional neural networks for sentence classification. In *2014 Conf. on Empirical Methods in Natural Language Processing, EMNLP*, pages 1746–1751. Association for Computational Linguistics, October 2014.

[Kotikalapudi, 2017] Raghavendra Kotikalapudi. keras-vis, 2017. GitHub.

[Kunkel *et al.*, 2019] Johannes Kunkel, Tim Donkers, Lisa Michael, Catalin-Mihai Barbu, and Jürgen Ziegler. Let me explain: Impact of personal and impersonal explanations on trust in recommender systems. In *2019 Conf. on Human Factors in Computing Systems, CHI*, page 487, 2019.

[Lawrence and Reed, 2019] John Lawrence and Chris Reed. Argument mining: A survey. *Comput. Linguistics*, 45(4):765–818, 2019.

[Lertvittayakumjorn and Toni, 2019] Piyawat Lertvittayakumjorn and Francesca Toni. Human-grounded evaluations of explanation methods for text classification. In *2019 Conf. on Empirical Methods in Natural Language Processing - 9th Int. J. Conf. on Natural Language Processing, EMNLP-IJCNLP*, pages 5194–5204, 2019.

[Lertvittayakumjorn *et al.*, 2020] Piyawat Lertvittayakumjorn, Lucia Specia, and Francesca Toni. FIND: human-in-the-loop debugging deep text classifiers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 332–348, 2020.

[Lippi and Torroni, 2016] Marco Lippi and Paolo Torroni. Argumentation mining: State of the art and emerging trends. *ACM Trans. Internet Techn.*, 16(2):10:1–10:25, 2016.

[Lundberg and Lee, 2017] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions.

In *30th Conf. on Neural Information Processing Systems, NeurIPS*, pages 4765–4774, 2017.

[Maas *et al.*, 2011] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150. Association for Computational Linguistics, June 2011.

[Madumal *et al.*, 2019] Prashan Madumal, Tim Miller, Liz Sonenberg, and Frank Vetere. A grounded interaction protocol for explainable artif. intel. In *18th Int. Conf. on Autonomous Agents and MultiAgent Systems, AAMAS*, pages 1033–1041, 2019.

[Mahendran and Vedaldi, 2015] Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. In *2015 IEEE Conf. on Computer Vision and Pattern Recognition, CVPR*, pages 5188–5196, 2015.

[Miller, 2019] Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artif. Intel.*, 267:1–38, 2019.

[Mordvintsev *et al.*, 2015] Alexander Mordvintsev, Christopher Olah, and Mike Tyka. Inceptionism: Going Deeper into Neural Networks. *GoogleAIBlog*, 2015.

[Mu and Andreas, 2020] Jesse Mu and Jacob Andreas. Compositional explanations of neurons. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[Naveed *et al.*, 2018] Sidra Naveed, Tim Donkers, and Jürgen Ziegler. Argumentation-based explanations in recommender systems: Conceptual framework and empirical results. In *Adjunct Publication of the 26th Conf. on User Modeling, Adaptation and Personalization, UMAP*, pages 293–298, 2018.

[Olah *et al.*, 2017] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. *Distill*, 2017.

[Olah *et al.*, 2018] Chris Olah, Arvind Satyanarayan, Ian Johnson, Shan Carter, Ludwig Schubert, Katherine Ye, and Alexander Mordvintsev. The building blocks of interpretability. *Distill*, 2018.

[Pennington *et al.*, 2014] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *2014 conference on empirical methods in natural language processing, EMNLP*, pages 1532–1543, 2014.

[ProRepublica, 2016] ProRepublica. COMPAS Recidivism Risk Score Data and Analysis, 2016.

[Rader *et al.*, 2018] Emilee J. Rader, Kelley Cotter, and Janghee Cho. Explanations as mechanisms for supporting algorithmic transparency. In *2018 Conf. on Human Factors in Computing Sys., CHI*, page 103, 2018.

[Rago *et al.*, 2018] Antonio Rago, Oana Cocarascu, and Francesca Toni. Argumentation-based recommendations: Fantastic explanations and how to find them. In *27th Int. J. Conf. on Artif. Intel., IJCAI*, pages 1949–1955, 2018.

[Ramamurthy *et al.*, 2020] Karthikeyan Natesan Ramamurthy, Bhanukiran Vinzamuri, Yunfeng Zhang, and Amit Dhurandhar. Model agnostic multilevel explanations. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[Ribeiro *et al.*, 2016] Marco Túlio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should I trust you?": Explaining the predictions of any classifier. In *22nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, pages 1135–1144, 2016.

[Ribeiro *et al.*, 2018] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Semantically equivalent adversarial rules for debugging NLP models. In *56th Annual Meeting of the Association for Computational Linguistics, Volume 1*, pages 856–865, 2018.

[Schwab and Karlen, 2019] Patrick Schwab and Walter Karlen. Cxplain: Causal explanations for model interpretation under uncertainty. In *32nd Conf. on Neural Information Processing Systems, NeurIPS*, pages 10220–10230, 2019.

[Selvaraju *et al.*, 2020] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. *Int. J. Comput. Vis.*, 128(2):336–359, 2020.

[Shih *et al.*, 2018] Andy Shih, Arthur Choi, and Adnan Darwiche. A symbolic approach to explaining bayesian network classifiers. In *27th Int. J. Conf. on Artif. Intel., IJCAI*, pages 5103–5111, 2018.

[Simonyan and Zisserman, 2015] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *3rd Int. Conf. on Learning Representations, ICLR*, 2015.

[Sokol and Flach, 2020] Kacper Sokol and Peter A. Flach. Explainability fact sheets: a framework for systematic assessment of explainable approaches. In *FAT* '20: Conf. on Fairness, Accountability, and Transparency*, pages 56–67, 2020.

[Wang and Nasconcelos, 2019] Pei Wang and Nuno Nasconcelos. Deliberative explanations: visualizing network insecurities. In *32nd Conf. on Neural Information Processing Systems, NeurIPS*, pages 1372–1383, 2019.

Appendices

A DAX General Methodology Details

In Step 2, we mentioned that the various choices "result in a GAF which (when seen as a graph with edges of m different kinds) is a tree with the explained node's argument at the root". Formally, σ and $\langle \mathcal{A}, \mathcal{R}_1, \ldots, \mathcal{R}_m \rangle$, with accompanying ρ and $c_1, \ldots, c_m$, need to be such that:

- $\mathcal{A} = \mathcal{A}_1 \cup \ldots \cup \mathcal{A}_k$ where $\mathcal{A}_i$ is the set of arguments representing the nodes at stratum N_i.

  Intuitively, arguments represent nodes and are partitioned so as to mirror the strata.
- $\mathcal{A}_k = \{\alpha_o\}$ such that $\rho(\alpha_o) = n_o$.

  Intuitively, the single argument from the last stratum represents the output node.
- $\forall i \in \{2, \ldots, k\}, \forall \alpha_h \in \mathcal{A}_i$ and $\forall n_g \in N_{i-1}$:

  $\exists \alpha_g \in \mathcal{A}_{i-1}$ such that $\rho(\alpha_g) = n_g$ iff

  $\exists (n_g, n_h) \in \mathcal{I}$ and $\exists l \in \{1, \ldots, m\}$ such that $c_l(n_g, n_h) = true$;

  if $n_h = \rho(\alpha_h)$ then $(\alpha_g, \alpha_h) \in \mathcal{R}_l$.

  Intuitively, nodes in (a stratum in our) $\mathcal{N}$ are represented by arguments in the GAF iff they influence nodes (in the next stratum) represented by arguments themselves (because ultimately they lead to influences towards the output node) that contribute to one of the dialectical relations.
- $\forall (\alpha_p, \alpha_q), (\alpha_r, \alpha_s) \in \mathcal{R}_1, \ldots, \mathcal{R}_m$ if $\rho(\alpha_p) = \rho(\alpha_r)$ and $\rho(\alpha_q) \neq \rho(\alpha_s)$ then $\alpha_p \neq \alpha_r$.

  Intuitively, a node influencing several nodes may need to correspond to several arguments (one for each influence that makes it into the dialectical relations, e.g. see in Figure 1 in the paper, with two arguments α_{11} and α_{12} representing n_{h1} to distinguish between its influences on n_{h1} and n_{h2}.).

B DAX Instantations Details

For all the models we used Keras with the Adam optimizer and we set the random seeds of Python (`random`), NumPy and TensorFlow/Keras to 0.

B.1 CNN for text classification

As mentioned in the paper we targeted a CNN architecture with an input layer of 150 words followed by: an embeddings layer (6B GloVe embeddings of size 300); a hidden layer of 20 1D ReLU convolutional filters of which 5 are of size 2, 10 are of filter size 3 and 5 are of size 4; followed by a max pooling layer; a dropout layer with probability 0.3; and finally a dense softmax layer. We trained the network in batches of 128 samples for a maximum of 100 epochs with an early stopping with patience 3. We split the IMDB dataset [Maas *et al.*, 2011] in training (20000 samples), validation (5000) and test (25000) sets. The CNN trained on this dataset has accuracy of 81.81% on the test set. We split the AG-News dataset [Gulli, 2005] in training (96000 samples), validation (24000) and test (7600) sets. We pre-processed the AG-News dataset removing HTML characters, punctuation and lowering all the text. The model trained on this dataset has accuracy of 90.14% on the test set. To run the empirical experiments in Section 5 for this instance, we used the test set. To generate the word clouds, we set the Pandas' random seed set to 2020 and randomly sampled 1000 articles from the train set (as in [Lertvittayakumjorn *et al.*, 2020]).

In **step 3**, we showed only two kinds of interactions in Figure 4. Figure 7 shows the interactive explanation after clicking on a word of the input text (representing an argument), a third kind of interaction with the explanation afforded by our DAX in this instantiation.

B.2 CNN for Image Classification

We used a model of VGG-16 pre-trained on the ImageNet dataset [Deng *et al.*, 2009] available in the Keras distribution we used. The model has a top-1 accuracy of 71.3% and a top-5 accuracy of 90.1% on the ImageNet validation set [Keras, 2020]. To run the empirical experiments in Section 5, we downloaded a randomized sample of correctly predicted images from ImageNet. To generate the activation maximization of each filter we used Keras-vis library and ran it multiple times for filters for which the algorithm did not converge (by manual inspection). The first time we used a jitter of 0.05 and 200 iterations, the second a jitter of 0.02 and 500 iterations, then a jitter of 0.04 and 1000 iterations, then a jitter of 0.05 and 1500 iterations, then a jitter of 0.05 and 3000 iterations, and finally a jitter of 0.01 and 1000 iterations, after which all filters' activation maximizations converged.

In **step 2**, we chose to extract a SAF $\langle \mathcal{A}, \mathcal{R}^+ \rangle$ with a single dialectical relation of support $\mathcal{R}^+$. The set of arguments $\mathcal{A}$ is divided in 3 disjoint sets corresponding to the strata: $\mathcal{A}_3 = \{\alpha_o\}$, with α_o the *output argument*; $\mathcal{A}_2$ is the set of all *intermediate arguments* α_j representing convolutional filter n_j in the the second stratum; $\mathcal{A}_1$ is the set of all the *input arguments* α_{xyj} representing a pixel n_{xy} in the input image influencing a convolutional filter n_j in the second stratum. Figure 8 shows an excerpt of the argumentation framework generated in this step; it is even more evident than in the previous instantiation how the argumentation framework can hardly be used as an explanation itself because of its sheer size (1,472,224 arguments for the example in the figure).

In this instantiation, the relation characterisation and the argument strength are based on the Grad-CAM weighted forward activation maps [Selvaraju *et al.*, 2020] of the last convolutional layer. Let $G_j^o = g_j^o A_j$ be the (Grad-CAM) weighted forward activation map of filter j (in the last convolutional layer) resized to the input size, g_j^o the Grad-CAM neuron importance weight of filter j for class c in the last convolutional layer and A_j the feature map activations matrix of filter j. The relation characterisation for support ($\mathcal{R}^+$) is then defined as:

$$c_+(n, m) = \begin{cases} true & \text{if } n = n_j \wedge g_j^o > 0 \\ true & \text{if } n = n_{xy} \wedge m = n_j \wedge G_j^o(x, y) > 0 \\ false & \text{otherwise} \end{cases}$$

The dialectical strength of the arguments is defined as:

$$\sigma(\alpha) = \begin{cases} \sum_{j,x,y} G_j^o(x, y) & \text{if } \alpha = \alpha_o \in \mathcal{A}_2 \\ \sum_{x,y} G_j^o(x, y) & \text{if } \alpha = \alpha_j \in \mathcal{A}_1 \\ G_j^o(x, y) & \text{if } \alpha = \alpha_{xyj} \in \mathcal{A}_0 \end{cases}$$

This instantiation of DAX satisfies the properties of *dialectical monotonicity* and *additive monotonicity*.

In **step 3**, we transformed the SAF into a graphical interactive explanation. Figure 10 shows the interactive explanation of an image in its initial state in which the size of the filter activation maximization maps are proportional to the corresponding arguments strengths. Figure 11 shows the in-

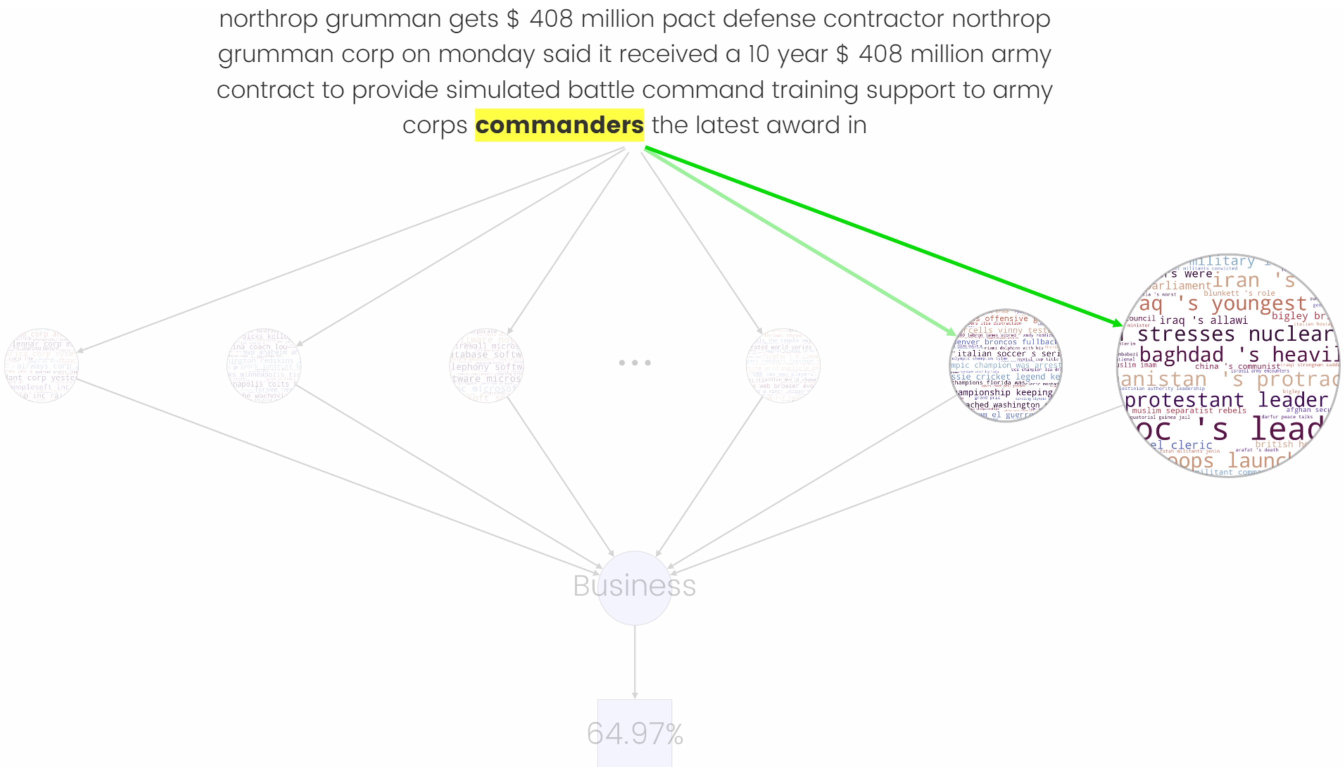

Figure 7: Graphical interactive DAX for text classification of an article from AG-News **after clicking on a word of the input text**. Arguments are on three levels: *input words* with highlighted in yellow the selected word, *convolutional filters/word clouds* that the (arguments corresponding to the) word influences, with sizes representing the corresponding arguments' strengths, and the most probable *class* and its probability. The arrows' green colour indicates support, while the intensity indicates the arguments' strength.

teractive explanation after clicking on the filter activation maximization map of a feature (that seems associated with the human-comprehensible visual concept of "guitar cords").

B.3 Feed-forward neural network for binary classification of tabular data

We used the COMPAS dataset [ProRepublica, 2016] understanding *sex*, *age*, *race*, *juv_fel_count*, *juv_misd_count*, *juv_other_count*, *priors_count*, *is_recid*, *is_violent_recid*, *custody*, *charge_desc*, *charge_degree* as categorical input features, and *two_year_recid* as binary prediction. We removed all the records with unknown value and transformed in categorical variables *custody*, *priors_count*, *juv_other_count*, *juv_fel_count* and *juv_misd_count*. We trained the network in batches of size 32 for a maximum of 10 epochs with early stopping with patience 5. We split the dataset in training (5213 samples) and test (1738 samples) sets. The trained (10 epochs) network has prediction accuracy of 70.2% on the test set. The fact that the prediction accuracy for this dataset is relatively low does not affect our results since the goal of our method is to *explain* the internal mechanism of the classifier in making a prediction (no matter how accurately). To run the empirical experiments in Section 5, we used the test set.

In **step 2**, we chose to extract a TAF $\langle \mathcal{A}, \mathcal{R}^-, \mathcal{R}^+, \mathcal{R}^{!+} \rangle$. The set of arguments $\mathcal{A}$ that can therefore be divided in 3 disjoint sets: $\mathcal{A}_3 = \{\alpha_o\}$, where α_o is the *output argument*; $\mathcal{A}_2$ is the set of *intermediate arguments* α_j representing a hidden neuron n_j in the second stratum; $\mathcal{A}_1$ is the set of *input arguments* α_{ij} representing an input neuron n_i in the first stratum influencing a hidden neuron n_j in the second stratum. Figure 12 shows an argumentation framework extracted at this step containing 92 arguments.

We let a_x be the activation of neuron n_x and w_{xy} the connection weight between neuron n_x and neuron n_y. We define the relation characterisations as:

$$c_-(n_x, n_y) = true \quad \text{iff } w_{xy}a_x < 0$$
$$c_+(n_x, n_y) = true \quad \text{iff } w_{xy}a_x > 0$$
$$c_{!+}(n_x, n_y) = true \quad \text{iff } a_y > 0 \wedge a_y - w_{xy}a_x \leq 0$$

The dialectical strength is defined as:

$$\sigma(\alpha) = \begin{cases} a_o & \text{if } \alpha = \alpha_o \in \mathcal{A}_2 \\ |w_{jo} \cdot a_j| & \text{if } \alpha = \alpha_j \in \mathcal{A}_1 \\ |w_{ij} \cdot w_{jo} \cdot a_i| & \text{if } \alpha = \alpha_{ij} \in \mathcal{A}_0 \end{cases}$$

Formally, a notion of dialectical strength σ satisfies the dialectial property of *binary counter-factuality* on a TAF $\langle \mathcal{A}, \mathcal{R}^-, \mathcal{R}^+, \mathcal{R}^{!+} \rangle$ iff for any $\alpha, \beta \in \mathcal{A}$ such that $(\alpha, \beta) \in \mathcal{R}^{!+}$ it holds that $\sigma(\beta) > 0$ and $\sigma(\beta) - \sigma(\alpha) \leq 0$.

For **step 3**, we transform the TAF into a graphical explanation divided into 3 levels reflecting the strata structure (see Figure 13): (1) the input features (represented by the darker slices of the pie charts), (2) the hidden features (represented by the pie charts), and (3) the prediction and its probability.

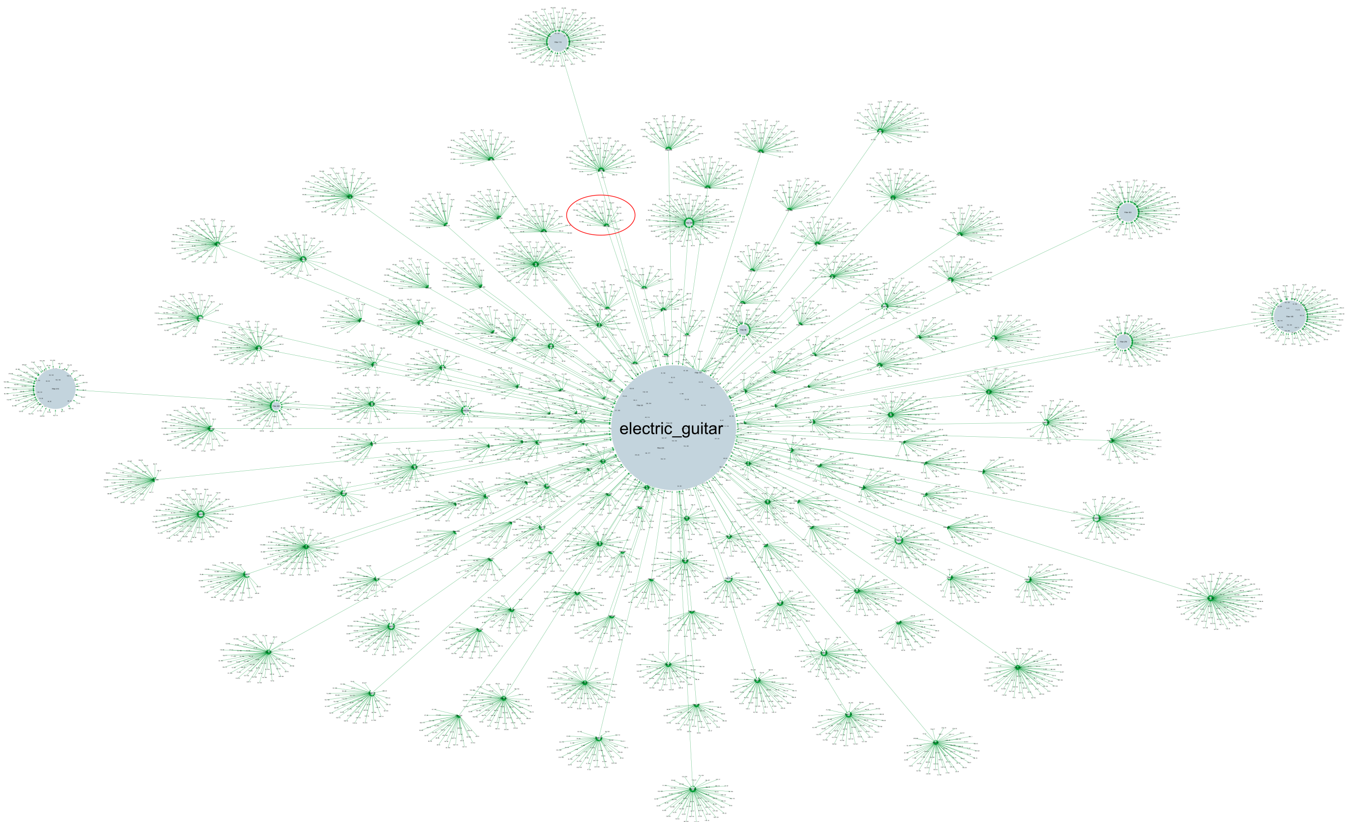

Figure 8: Excerpt of the support argumentation framework from which the DAX for explaining the prediction from an image from ImageNet is obtained. The bubbles are arguments $\alpha \in \mathcal{A}$ with sizes corresponding to their dialectical strength $\sigma(\alpha)$ and as label their mapping to nodes $\rho(\alpha)$. Green arrows indicate support. Note that only a subset of 5145 out of the 1,472,224 arguments are visualized in this figure. The part within the red oval is magnified in Figure 9 in which (for that portion of the SAF) we show also the arguments that in this excerpt were hidden.

As shown in Figure 14, we associate (through the mapping χ) a human-comprehensible meaning to each intermediate argument, by showing a pie chart with the input features that minimize (in red) or maximize (in green or blue) the activation of each of the hidden neurons (associated with the arguments). We represent the dialectical relations by highlighting with dark red, dark green and blue the slices of the input features (corresponding to arguments) attacking, supporting or strictly/critically supporting (resp.) the intermediate argument. Thus, given the peculiarity of this FFNN in which an input feature can only be 0 or 1 (because of the one-hot encoding), we joined together the visualisation of χ (i.e. the meaning of intermediate arguments) and the visualisation of the input features using a pie chart (and their strength given by the size of slice).

C Details of empirical evaluation

Computational Cost. Formally, for the CNN for text, given the time cost for a prediction c_f (1.7 ± 0.4 ms, in our experiments) and the cost to back-propagate the output to the inputs c_b (50.2 ± 11.0 ms, in our experiments) while we let c be any other (constant or negligible) computational cost, the generation of the word clouds (a one-time cost for χ) using n_S training samples (we used 1000 to generate the example in Figure 7) has time complexity $O(n_S \cdot (c_f + c_b) \cdot |N_2| + c)$. The time complexity to generate a single DAX is instead $O(c_f + c_b \cdot |N_2| + c)$.

For the CNN for image classification, given the time cost for an iteration of the feature activation maximization algorithm c_g (39.1 ± 8.5 ms, in our experiments) while we let c be any other (constant or negligible) computational cost, the generation of the feature activation maximization maps (a one-time cost for χ) with n_I iteration (we used between 200 and 3000 to generate the Figure 10, depending on the filter convergence characteristics) has time complexity $O(c_g \cdot |N_2| \cdot n_I + c)$. Given the time cost for a prediction c_f (7.5 ± 0.8 ms, in our experiments) and that to back-propagate the output to the inputs c_b (7.8 ± 0.7 ms, in our experiments), the time complexity to generate a single DAX is instead $O(c_f + c_b \cdot |N_2| + c)$.

For the FFNN for COMPAS, the generation of the base for the pie chart (a one-time cost for χ) has a constant time complexity since it it can be computed based on the input weights. Given the time cost for a prediction c_f (0.6 ± 0.1 ms, in our experiments) while we let c be any other (constant or negligible) computational cost, the time complexity to generate a single DAX is $O(c_f + c)$.

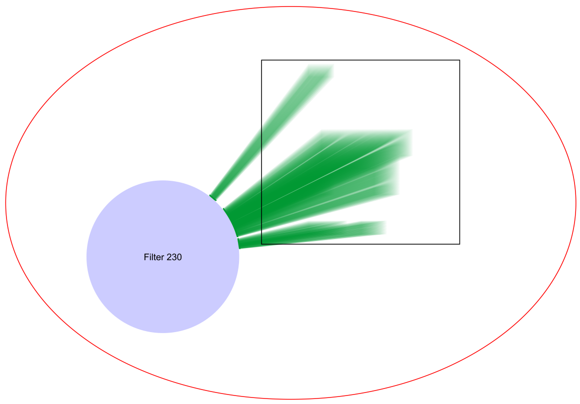

Figure 9: Zoom of the support argumentation framework from which the DAX for explaining the prediction from an image from ImageNet is obtained. The bubbles are arguments $\alpha \in \mathcal{A}$ with sizes corresponding to their dialectical strength $\sigma(\alpha)$ and as label their mapping to nodes $\rho(\alpha)$. Green arrows indicate support. This figure shows all the arguments corresponding to pixels (inside the grey rectangle) for the 230th filter of the CNN.

C.1 Software Platform and Computing Infrastructure

To run our experiments, we used a single Nvidia RTX 2080 Ti GPU with 11GB of memory on a machine with an Intel i9-9900X processor and 32GB of RAM. We used a Python 3.6 environment with numpy 1.18.1, pandas 1.0.3, Keras 2.2.4, TensorFlow 1.12, SpaCy 2.2.3, shap 0.31.0, keras-vis 0.4.1, innvestigate 1.0.8 and lime 0.2.0.0 as software platform to run our experiments.

C.2 Hyper-parameters training

CNN for Text Classification. The hyper-parameters of the network were chosen after exploring all the following combinations:

- batch size $\in \{64, 128, 256, 512\}$
- patience $\in \{3, 5\}$
- dropout $\in \{0.0, 0.1, 0.2, 0.3, 0.5\}$
- filters sizes sets $\in$ $\{$ $[(10, 2), (10, 3), (10, 4), (10, 5)]$, $[(5, 2), (5, 3), (5, 4), (5, 5)]$, $[(15, 2), (15, 3), (15, 4)]$, $[(20, 2), (10, 3), (5, 4)]$, $[(10, 2), (10, 3), (10, 4)]$, $[(12, 2), (10, 3), (8, 4)]$, $[(15, 2), (10, 3), (5, 4)]$, $[(7, 2), (7, 3), (7, 4)]$, $[(5, 2), (5, 3), (5, 4)]$, $[(10, 2), (10, 3), (5, 4)]$, $[(5, 2), (15, 3), (5, 4)]$, $[(5, 2), (12, 3), (5, 4)]$, $[(5, 2), (10, 3), (5, 4)]$, $[(5, 2), (10, 3), (10, 4)]$, $[(5, 2), (5, 3), (10, 4)]$, $[(15, 2), (15, 3)]$, $[(20, 2), (20, 3)]$, $[(10, 2), (15, 3)]$, $[(15, 2), (10, 3)]$, $[(15, 3), (10, 4)]$ $\}$ where $[(n_1, s_1), \ldots, (n_m, s_m)]$ represent a setting in which n_1 filters of size s_1 were used (and n_2 of size s_2, etc.).

FFNN for tabular data on COMPAS. The hyper-parameters of the network were chosen after exploring the following alternatives:

- hidden layer activation function $\in \{tanh, relu, tanh + relu\}$
- batch size $\in \{4, 8, 16, 32, 64\}$

D Details of the human experiments

Figure 15 shows in details the background of the 72 workers of Amazon Mechanical Turk that participated in the experiments.

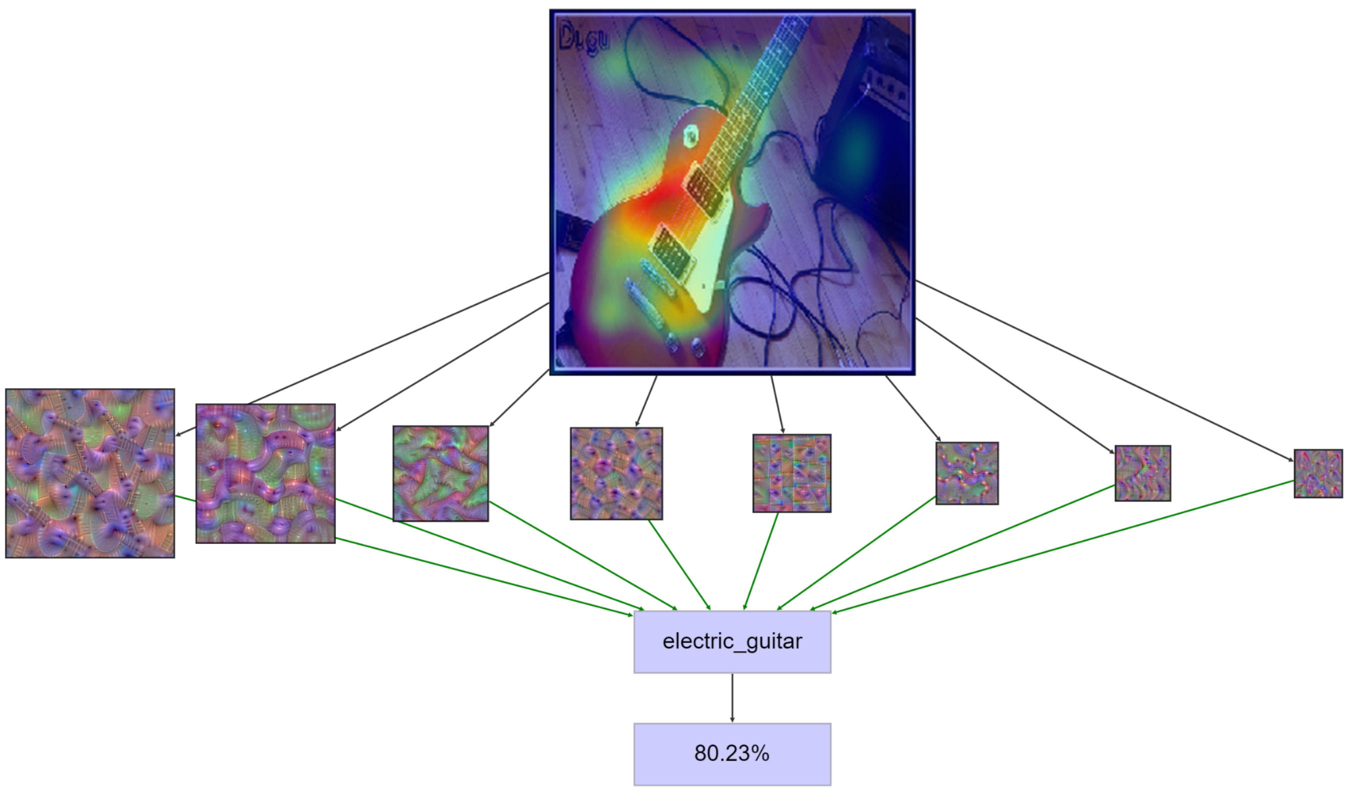

Figure 10: Graphical interactive DAX for image classification of an image from ImageNet in its **initial state** (i.e. prior to interactions of the user with the explanation). Arguments are on three levels: *pixel in the input image* with heatmap of a pixel given by the sum of the (dialectical) strengths of all arguments representing that pixel, *convolutional filters/activation maximization maps* with sizes representing the corresponding arguments' strengths, and the most probable *class* and its probability. The green colour of the arrows indicates support. In the input image the strongest supporting pixels of the predicted class are red, then green, then blue.

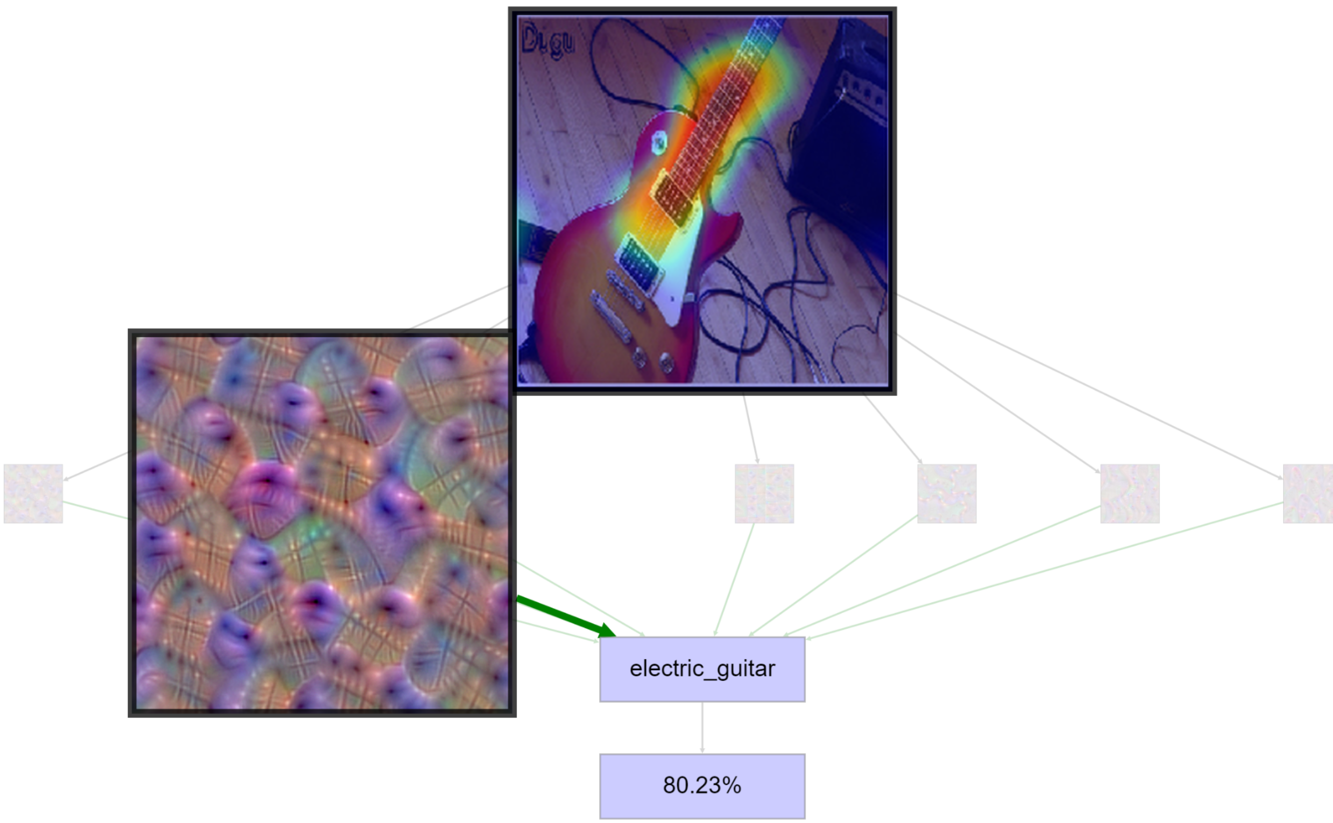

Figure 11: Graphical interactive DAX for image classification of an image from ImageNet **after clicking on a filter activation maximization map**. As in Figure 10, arguments are on three levels: *pixel in the input image* with a heatmap that indicates strengths of the arguments representing that pixel influencing a convolutional filter, the **selected** *convolutional filter/activation maximization map*, and the most probable *class* and its probability. The green colour of the arrows indicates support. In the input image the strongest supporting pixels of the predicted class are red, then green, then blue.

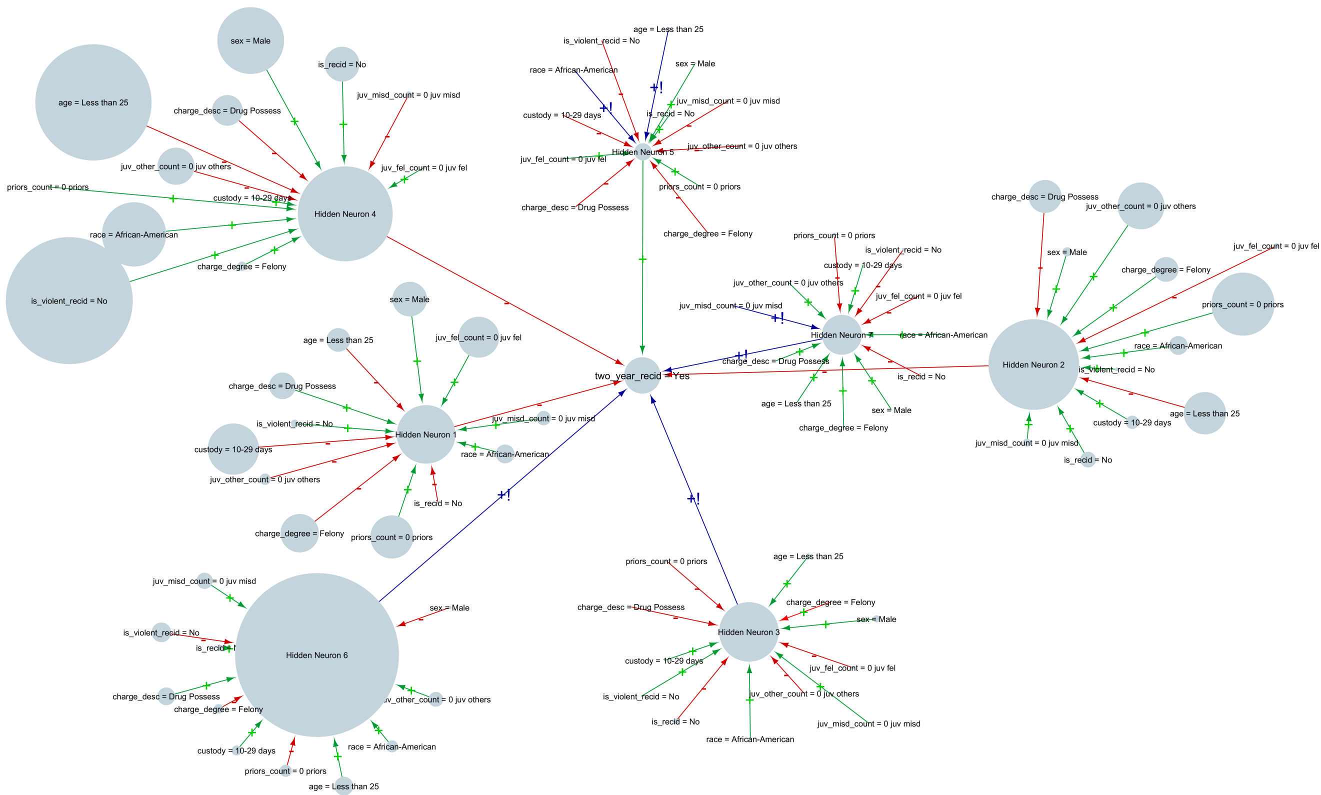

Figure 12: Argumentation framework from which the DAX for explaining the prediction from a sample from the COMPAS dataset is obtained. The bubbles are arguments $\alpha \in \mathcal{A}$ with sizes corresponding to their dialectical strength $\sigma(\alpha)$ and as label their mapping to nodes $\rho(\alpha)$ with the current value. Red, green and blue arrows indicate attack, support, and critical support, resp..

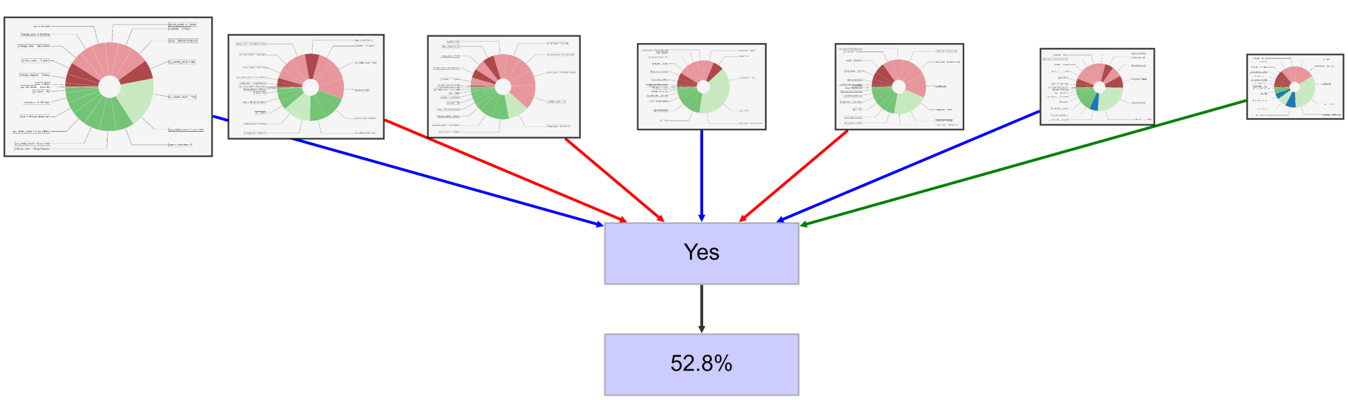

Figure 13: Graphical interactive DAX of a sample from the COMPAS test set for a FFNN for binary classification, in its **initial state** (i.e., prior to interactions of the user with the explanation). Arguments are on three levels: those representing *hidden neurons* (i.e. the pie charts) influencing the predicted class with sizes representing the corresponding their strengths, those representing *input features* visualized as the darker slices of the pie charts corresponding to hidden neurons, and the most probable *class* and its probability. The red, green and blue colours of the arrows indicate attack, support and critical/strict support, resp..

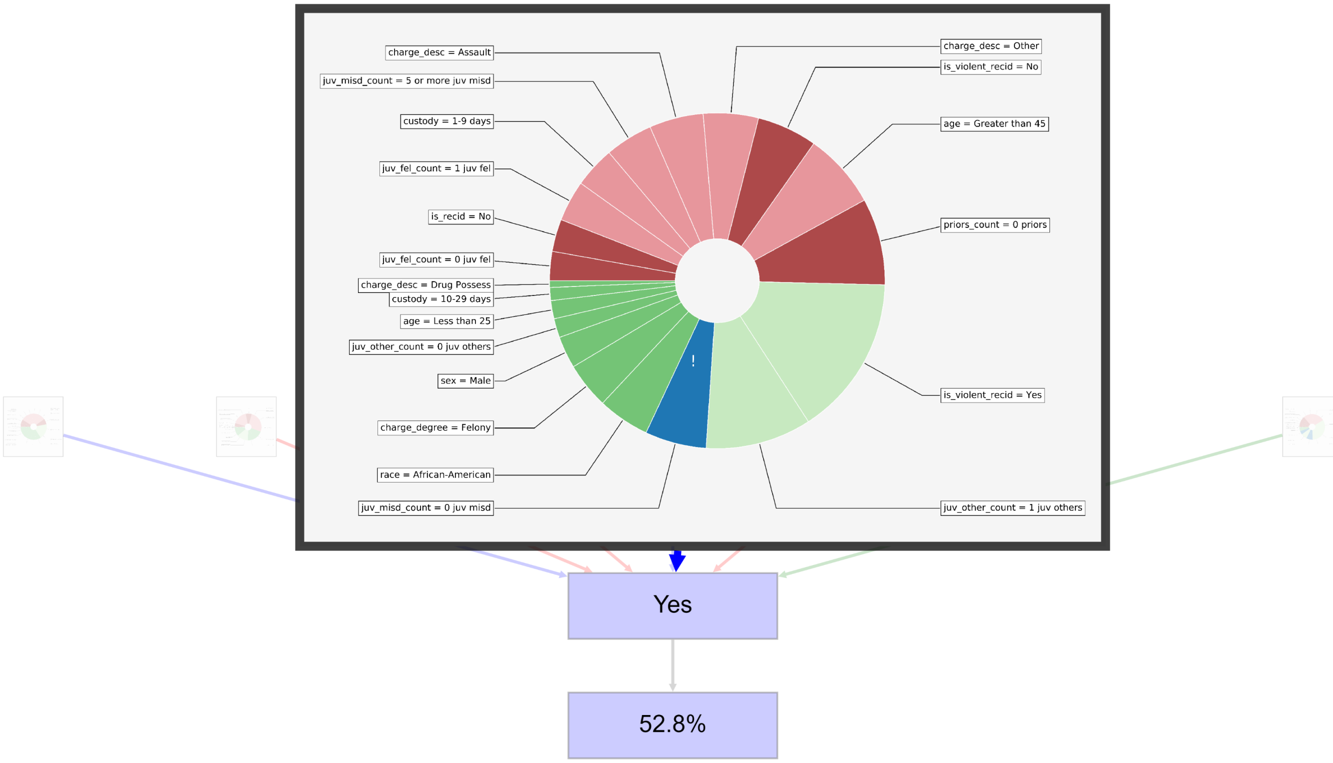

Figure 14: Graphical interactive DAX of a sample from the COMPAS test set for a FFNN for binary classification, **after clicking on the (second from the right) pie chart**. As in Figure 13, arguments are on three levels: the single (argument representing the) ***selected*** *hidden neuron* (i.e. the pie chart), the arguments representing *input features* visualized as the darker slices of the pie chart with size proportional to their strengths, and the most probable *class* and its probability. In the pie chart, slices coloured of (light or dark) red represent a subset of (the most important) input features that minimize the activation of the hidden neuron represented by the intermediate argument, while slices coloured of green or blue represent the ones that maximize it; slices coloured of dark red, dark green and blue highlight input features that for this sample are attacking, supporting and critically/strictly supporting the intermediate argument, resp..

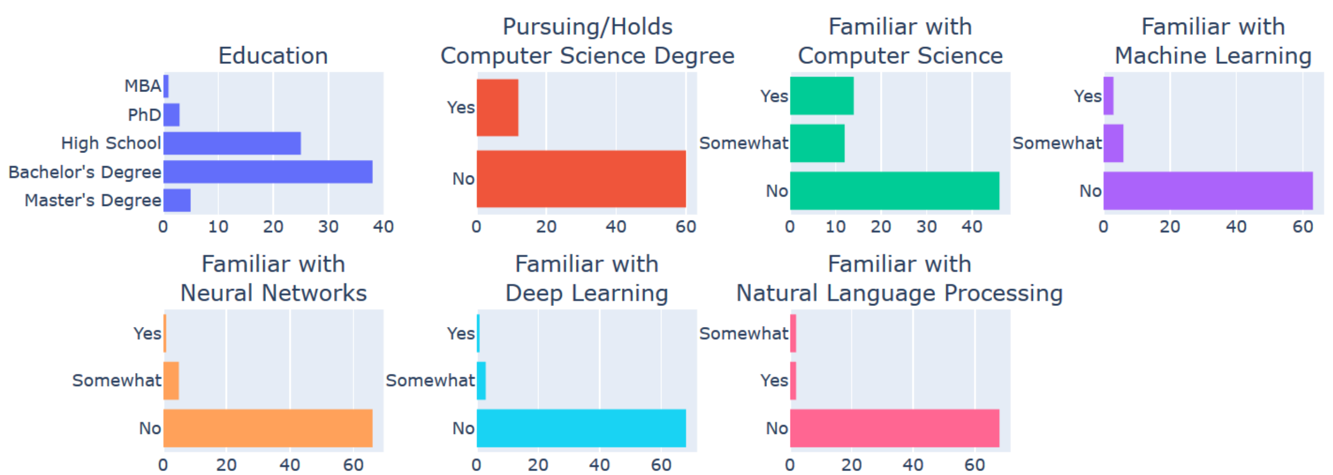

Figure 15: Backgrounds of participants in the human experiments