Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

RuleConEx: an intrinsically interpretable neural network unifying logical rules and counterfactual explanations for data access control

Benoît Martin Azanguezet Quimatio
Université de Dschang
Erick Tsafack Nteudem
erickнteudem@gmail.com

Université de Dschang
Paul Valery Nguezet
Université de Dschang

Research Article

Keywords:

Posted Date: October 23rd, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11261894/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

RuleConEx: an intrinsically interpretable neural network unifying logical rules and counterfactual explanations for data access control

Benoît Martin Azanguezet Quimatio[1] · **Erick Tsafack Nteudem**[1] · **Paul Valery Nguezet**[1]

Received: date / Accepted: date

Abstract Deep learning based access control models such as DLBAC are accurate but opaque, which makes them difficult to audit against the accountability requirements of regulations such as the GDPR. Existing self-explaining models provide only one kind of explanation: global rules (HyperLogic) or local counterfactuals (HyConEx). We propose RuleConEx, a model trained jointly around a shared hypernetwork that produces, in a single forward pass, an access decision, IF-THEN rules decoded exactly from the network weights and expressed in the language of ABAC policies, and counterfactuals that are valid categorical tuples by construction. Rule faithfulness and counterfactual validity are measured rather than assumed. On the eleven DLBAC benchmarks, evaluated with five seeds, a baseline juxtaposing separately trained HyperLogic and HyConEx models, and statistical tests with Nemenyi and Holm corrections, RuleConEx reaches 92.1% mean accuracy, statistically indistinguishable from HyConEx, random forests and linear SVM, and 0.6 points below an opaque MLP. Its rules cover 80% of the requests with a strict faithfulness of 0.80, and 96% of its counterfactuals are valid, changing two attributes on average. Sweeping the weight of the counterfactual loss shows that this single coefficient continuously shifts the trade-off between user-oriented explainability (counterfactuals of 1.9 attributes) and auditor-oriented explainability, where the consistency between rules and counterfactuals rises from 0.18 to 0.43, with no statistically detectable difference in accuracy. RuleConEx therefore provides an explicit tuning mechanism for this trade-off, which two juxtaposed models do not offer.

Keywords Access control · Deep learning · Intrinsic interpretability · Hypernetwork · IF-THEN rules · Counterfactual explanations

E. Tsafack Nteudem (corresponding author)
[1]Department of Mathematics and Computer Science, Faculty of Science, University of Dschang, Dschang, Cameroon
E-mail: ericknteudem@gmail.com
ORCID: https://orcid.org/0009-0002-7619-5185

B. M. Azanguezet Quimatio
[1]Department of Mathematics and Computer Science, Faculty of Science, University of Dschang, Dschang, Cameroon
E-mail: azanguezet@gmail.com
ORCID: https://orcid.org/0009-0002-2638-7322

P. V. Nguezet
[1]Department of Mathematics and Computer Science, Faculty of Science, University of Dschang, Dschang, Cameroon
E-mail: nguezetpaulvalery@gmail.com
ORCID: https://orcid.org/0009-0008-2975-8153

1 Introduction

Access control is a fundamental cyber-security mechanism that determines which subjects may perform which operations on which resources, according to policies defined over contexts, users and actions [31]. For several decades these policies relied on predefined rule-based models: DAC, MAC, RBAC [31], ABAC [17], Or-BAC [19] and HOr-BAC [4]. Such models are readable and auditable, but they impose tedious manual policy engineering, and their rigidity makes them hard to maintain as soon as the environment evolves quickly, as in cloud or IoT deployments [26].

Deep learning models have emerged as an attractive alternative. Rather than writing rules by hand, the model learns access policies directly from historical authorisation data. Nobi et al. [26] showed with DLBAC (Deep Learning Based Access Control) that this approach outperforms classical policy mining on every classification metric. That performance comes at a price: deep networks make decisions through thousands of parameters whose interaction escapes direct interpretation [30].

This opacity raises a genuine legal problem. Article 22 of the GDPR requires organisations to be able to explain

automated decisions affecting individuals [12]. In an access control system, a denial must be justifiable to the administrator who audits the policy, to the user who wants to understand it, and to the supervisory authority that checks it. Tools such as SHAP [24] and LIME [29] were proposed to fill this gap, but they do not really explain the model: they produce a local approximation that can be manipulated [32] and that is only partly faithful to the actual decision mechanism. For a critical system, such an approximation is not sufficient.

Two recent models show that one can do better. HyConEx [25] generates, in a single forward pass, a prediction together with actionable counterfactuals, answering the question "what should be changed to obtain access?". HyperLogic [38] produces global IF-THEN rules through differentiable rule learning parameterised by a hypernetwork. Each of these models provides a partial answer, but neither covers both needs: HyConEx provides no auditable rules for the administrator, and HyperLogic provides no counterfactual for the denied user.

This paper addresses that gap. We propose RuleConEx (Rule-based Counterfactual eXplainer), to our knowledge the first architecture that produces, in a single forward pass, an access decision obtained by weighted fusion of a rule branch and a counterfactual branch, global symbolic IF-THEN rules decoded directly from the weights of a DR-Net parameterised by the hypernetwork, and local actionable counterfactuals for the user whose access is denied. The starting idea is to let both explanation branches share a common hypernetwork. With respect to the works it builds upon, RuleConEx contributes three elements. First, its rules are *exact*: the function computed by the rule branch is fully decodable into a rule list, and the faithfulness of that list to the final decision is measured rather than postulated. Second, the vocabulary of these rules is that of the ABAC policies of Xu and Stoller [37], namely attribute–value equalities and relational constraints between user and resource attributes; we show that these constraints are indispensable on the DLBAC benchmarks, which incidentally explains the poor decision-tree results reported so far on these data. Third, counterfactuals are modifications of categorical attributes, valid by construction, with a mask of immutable attributes.

The evaluation — eleven DLBAC datasets, five seeds, a "HyperLogic + HyConEx trained separately" baseline, an ablation study, and Friedman and Wilcoxon tests with Nemenyi and Holm corrections — measures accuracy, rule quality (faithfulness, coverage, length, precision, stability), counterfactual quality (validity, sparsity, plausibility, actionability) and the consistency between the two. It establishes our main result: this consistency is governed by a single coefficient of the loss, whose sweep over seven values continuously shifts the trade-off between short counterfactuals and covering, consistent rules, with no detectable difference in accuracy on our protocol. We verify in particular that the consistency gain is not an artefact of counterfactual length. Finally, we discuss what exposing both kinds of explanation implies for the security of the access control system itself.

Section 2 reviews the state of the art. Section 3 details the methodology: datasets, architecture and formalisation of each component. Section 4 presents the experimental protocol and the results. Section 5 analyses the explanations as an attack surface. Section 6 discusses the scope of the results, and Section 7 concludes with limitations and perspectives.

2 Related work

This work lies at the intersection of three research areas: access control, interpretable IF-THEN rule learning, and counterfactual explanations. This section reviews recent work in each of them, stating for every contribution the problem attacked, the approach taken and the result obtained, before examining attempts to combine both forms of explanation.

2.1 From rule-based to learned access control

Classical access control models have progressively enriched policy expressiveness. RBAC [31] introduced roles to factor permissions and simplify administration; ABAC [17] generalised it by conditioning decisions on arbitrary attributes of subjects, resources and environment; Or-BAC [19] added the organisational dimension, and HOr-BAC [4] extended it with a hierarchy of organisations, making it the most expressive of these models. All share the advantage of natural transparency, since the policy is written explicitly by an administrator. In return, Nobi et al. [26] identify three structural limitations in modern environments: dependence on manual engineering, rigidity under change, and poor scalability when the number of rules grows combinatorially with the number of entities.

To automate this engineering, *policy mining* extracts rules from access logs. Xu and Stoller [37] infer ABAC policies by greedy search for rules covering the observed authorisations; their policies are tuples ⟨user attribute expression, resource attribute expression, operations, constraints⟩, where the *constraints* are equalities between a user attribute and a resource attribute ("same department"), a point we exploit directly. Rhapsody [8] casts the problem as subgroup discovery with a reliability criterion that limits overfitting; Karimi et al. [20] propose an unsupervised k-modes approach. Classical classifiers such as EPDE-ML [22] have also been used for direct decisions, without providing any explanation. DLBAC [26] departs from these approaches by training a deep network (ResNet [16], DenseNet, Xception) directly on authorisation tuples, without prior feature engineering, and outperforms policy mining and standard classifiers on all classification and generalisation metrics.

Its complete opacity, acknowledged by its own authors as a priority for future research, is its main shortcoming: the post-hoc solutions considered (Integrated Gradients, tree distillation) remain unfaithful to the actual decision mechanism and vulnerable to manipulation [32]. Very recent work attacks explainability in ABAC without resorting to intrinsically interpretable deep learning. EXTree [6] structures the policy as a tree to produce actionable explanations upon denial, and LANTERN [21] translates audit logs into natural-language ABAC policies using a large language model. Both approaches are post-hoc and decoupled from the decision model, providing neither rules faithful by construction nor counterfactuals, but their very existence confirms the topicality of the problem addressed here.

2.2 Interpretable IF-THEN rule learning

IF-THEN rule learning is one of the most natural forms of interpretability [30]. Decision trees (CART), RIPPER [7], Bayesian Rule Sets [35] and CG [9] produce readable rules, but their search (greedy, stochastic or combinatorial) is decoupled from gradient optimisation. DR-Net [28] lifts this limitation by encoding each rule as a conjunction of literals whose weights are binarised in the forward pass and estimated by a straight-through estimator in the backward pass, making the whole trainable by backpropagation; rules are then extracted directly from the binary weights, so that, as long as the rule network alone decides, faithfulness is one by construction. HyperLogic [38] extends DR-Net with a hypernetwork that generates the parameters of the rule network from Gaussian noise, producing a diversified mixture of rule sets; a HyperGAN-style discriminator pulls the distribution of latent codes back towards the prior to prevent that diversity from collapsing. HyperLogic brings the readability that DLBAC lacks, but says nothing to the denied user about what could be changed, and was designed and evaluated for binary classification on generic datasets rather than for access control.

2.3 Counterfactual explanations

Counterfactual explanations answer the question "what should be modified to change the decision?", providing directly actionable information [34, 14]. Most methods (CEM [11], FACE [27], CEGP [33], PPCEF [36]) operate post hoc and require a separate optimisation for each instance, with costs that are sometimes prohibitive (up to 67,000 s on the Adult dataset for CEGP). Interpretable deep models for tabular data (TabNet [2], NAM [1], IMN [18]) provide attributions by construction, but none produces counterfactuals intrinsically. HyConEx [25] is the first model to produce simultaneously a prediction and plausible counterfactuals for all alternative classes in a single forward pass. A hypernetwork generates a local weight matrix $W = H(x)$ defining a local linear classifier, and the counterfactual towards class m is obtained by the translation $x' = x - W_m$, optionally projected onto the data manifold by a normalising flow. HyConEx reports 99.9% validity while being orders of magnitude faster than post-hoc methods, but provides no global auditable rule; its translation is continuous, so that on categorical attributes encoded as one-hot vectors the counterfactual is in general not a valid tuple; and, like HyperLogic, it has not been evaluated on access control data.

Table 1 Comparison of approaches along four requirements. ✓ satisfied; ○ partially; – not satisfied

Approach	Access decision	Exact IF-THEN rules	Valid counterfact.	Single pass
DLBAC	✓	–	–	✓
Decision trees	✓	✓	–	✓
HyperLogic	✓	✓	–	✓
HyConEx	✓	–	○ (cont.)	✓
CEM / CEGP / PPCEF	–	–	✓	–
RuleConEx (this work)	✓	✓	✓	✓

2.4 Combining both forms of explanation

Global rules and local counterfactuals are complementary, the former addressing the administrator who audits the policy and the latter the user who is denied access. This review found no work that intrinsically produces both forms of explanation within a single model. The only combination achievable with existing tools is to juxtapose a rule extractor and a counterfactual generator trained separately; we turn this into an explicit baseline ("HyperLogic + HyConEx independent") in order to measure, rather than assume, the possible inconsistency between the two explanations.

Table 1 summarises the situation along four requirements of an explainable access control system. No existing approach satisfies them simultaneously. One architectural lesson emerges: HyperLogic and HyConEx both rely on the same basic mechanism, a hypernetwork generating the parameters of a main network. This convergence suggests that a shared hypernetwork could parameterise a rule branch and a counterfactual branch at once. This is the starting hypothesis of RuleConEx; Section 4.7 measures its contribution.

Table 2 The eleven DLBAC datasets

Dataset	Type	$\|u\|$	$\|r\|$	$\|$auth$\|$	Meta.	Cl.
amazon1	Real	9,560	7,517	32,769	8	2
amazon2	Real	9,560	7,517	32,769	13	2
amazon3	Real	9,560	7,517	32,769	13	2
u4k-r4k-auth11k	Synth.	4,500	4,500	10,964	8	16
u4k-r4k-auth21k	Synth.	4,500	4,500	20,979	11	16
u4k-r4k-auth22k	Synth.	4,500	4,500	22,583	13	16
u4k-r6k-auth28k	Synth.	4,500	6,738	28,751	13	16
u4k-r7k-auth20k	Synth.	4,500	7,194	20,033	11	16
u5k-r5k-auth12k	Synth.	5,250	5,250	12,690	8	16
u5k-r5k-auth19k	Synth.	5,250	5,250	19,535	10	16
u6k-r6k-auth32k	Synth.	6,000	6,000	32,557	10	16

3 Methodology

3.1 Datasets

The experiments use the eleven benchmark datasets introduced by Nobi et al. [26] to evaluate DLBAC (Table 2). Three datasets come from real data released by Amazon: *amazon1* originates from the Kaggle *Amazon Employee Access Challenge* and contains about 9,560 users, 7,517 resources and more than 32,000 authorisation tuples, with a marked imbalance (about 94% of the tuples correspond to granted access); *amazon2* and *amazon3* are augmentations built by Nobi et al. [26] with the same users and resources and additional resource metadata, and the same imbalance. The remaining eight datasets are synthetic, generated following the method of Xu and Stoller [37] as adapted by Nobi et al. [26]: a set of rules ⟨user attribute expression, resource attribute expression, operations, constraints⟩ is used to generate users, resources and their authorisations. To bring these synthetic datasets closer to the complexity observed on the Amazon data, where tuples with very similar metadata receive different decisions, part of the metadata of each entity (beyond the first eight) is hidden at training time while still influencing the actual decisions, simulating policies based on partially observable attributes. These eight datasets thus span increasing complexity, from 4,500 to 6,000 users and 8 to 13 metadata per entity, for a 16-class classification problem (the four binary operations combined).

Formally, each dataset contains a set of authorisation tuples $\langle \text{uid}, \text{rid}, \mathbf{m}^u, \mathbf{m}^r, \mathbf{op} \rangle$, where $\mathbf{m}^u, \mathbf{m}^r$ are the user and resource metadata and $\mathbf{op} \in \{0,1\}^4$ encodes the decisions for four operations. The problem is cast as multi-class classification, predicting the class y from $\mathbf{m} = (\mathbf{m}^u, \mathbf{m}^r)$. For the synthetic datasets the four bits are combined into an integer $y = op_1 + 2op_2 + 4op_3 + 8op_4 \in \{0, \ldots, 15\}$; for the Amazon datasets the decision is binary (*grant*/*deny*).

Preprocessing follows the DLBAC$_\alpha$ protocol: (1) removal of the uid/rid identifiers; (2) masking of the hidden metadata; (3) label construction; (4) categorical encoding fitted on the training set; (5) stratified 80/20 train/validation split (fixed seed 42), the test set being the DLBAC test file. Each tuple is thus described by G categorical attributes ($G = 16$ on the synthetic datasets, `umeta0..7` and `rmeta0..7`; $G = 9$ or 13 on Amazon).

Table 3 Effect of relational literals (u4k-r4k-auth11k, test accuracy). DT-k: tree of depth k; in parentheses, number of leaves of the full tree

Input	DT-4	DT-6	Full DT	Logistic	RF
one-hot only	0.219	0.268	0.913 (694)	0.972	0.976
one-hot + equalities	0.724	0.883	0.976 (169)	0.989	0.994

Literal space. The model input is $\mathbf{x} \in \{0,1\}^{d+E}$, the concatenation of (i) the one-hot encoding of the G attributes (d columns; $d = 133$ to 170 on the synthetic datasets, up to 14,419 on amazon1) and (ii) *E relational literals* $\mathbf{1}[\texttt{umeta}_i = \texttt{rmeta}_j]$ for every user–resource attribute pair ($E = 64$ on the synthetic datasets). This vocabulary is exactly that of the ABAC policies of Xu and Stoller [37], whose constraints are equalities between user and resource attributes. Its importance is empirical as much as conceptual: on u4k-r4k-auth11k, a depth-6 decision tree reaches 27% accuracy on the one-hot encoding alone and 88% with relational literals (Table 3); without them, no short conjunction can express a policy such as "same department as the resource". All compared models receive this representation; a "one-hot only MLP" variant quantifies its contribution. The objective is to learn $f : \{0,1\}^{d+E} \to \mathscr{Y}$ producing, for any request $\mathbf{x}$, a decision, a global set of IF-THEN rules, and, for each target class $t \neq \hat{y}$, a valid tuple $\mathbf{x}'_t$ with $f(\mathbf{x}'_t) = t$ changing a minimal number of attributes.

3.2 Overall architecture

RuleConEx consists of three components assembled around a shared hypernetwork: a latent encoder, a parameter-generating hypernetwork, and two specialised prediction branches. Their outputs are fused into a single decision accompanied simultaneously by symbolic rules and counterfactuals (Fig. 1). The principle of a hypernetwork $H(\cdot;\theta)$, whose role is to generate the parameters $\mathbf{W} = H(\mathbf{q};\theta)$ of a so-called main network from a query $\mathbf{q}$, was introduced by Ha et al. [15]. In RuleConEx, a single TabResNet trunk [13] is queried by two queries of different nature: a learned *global* query, independent of the instance, whose output parameterises the rule network — the rules are therefore identical for all requests, a necessary condition for auditability — and the *local* query $\mathbf{z}(\mathbf{x})$, whose output parameterises a linear classifier specific to each instance, in the manner of HyConEx. Both branches share the trunk weights, are trained jointly, and cover the three explanation needs of the AIX360 framework [3]: global rules for the auditor, local

justification of a denial, and an actionable counterfactual for the user.

3.3 Components

Encoder. A two-layer MLP with normalisation maps $\mathbf{x} \in \{0,1\}^{d+E}$ to a compact latent representation $\mathbf{z} \in \mathbb{R}^{64}$, the local query of the hypernetwork.

Hypernetwork. The TabResNet trunk $T : \mathbb{R}^{64} \to \mathbb{R}^{128}$ receives, for the rule branch, the global query $\mathbf{q}_r = \mu + \sigma\varepsilon$ with $\varepsilon \sim \mathcal{N}(\mathbf{0}, I)$, $\sigma = 0.1$ and $\mu \in \mathbb{R}^{64}$ learned ($M = 3$ draws during training, $\varepsilon = \mathbf{0}$ at inference, so that the rule set is unique). Since the codes are drawn from the prior with fixed σ, the HyperGAN discriminator of HyperLogic is unnecessary here. For the counterfactual branch the trunk receives $\mathbf{z}(\mathbf{x})$.

Rule branch. It is a binarised DR-Net with $K = 48$ rule neurons. For each rule k the hypernetwork produces an embedding $\mathbf{e}_k \in \mathbb{R}^{32}$; the selection probability of positive literal j ("$x_j = 1$") is $p^+_{kj} = \sigma(\mathbf{e}_k \cdot \mathbf{L}^+_j/\sqrt{32} + b^+_j)$, where $\mathbf{L}^+ \in \mathbb{R}^{(d+E)\times 32}$ is a learned matrix of literal embeddings (a low-rank parameterisation, necessary for $d \approx 14{,}000$). Negative literals p^-_{kj} ("$x_j = 0$") are allowed only on relational columns ("$\texttt{umeta}_i \neq \texttt{rmeta}_j$"), since value negations are almost always satisfied and bring no readability. The hard selections $a_{kj} = \mathbf{1}[p^+_{kj} > 0.5]$ and $n_{kj} = \mathbf{1}[p^-_{kj} > 0.5]$ are obtained with a straight-through estimator. The number of violated literals and the rule activation are

$$v_k(\mathbf{x}) = \textstyle\sum_j a_{kj}(1 - x_j) + n_{kj}x_j, \tag{1}$$

$$o_k(\mathbf{x}) = \mathbf{1}[v_k(\mathbf{x}) = 0], \tag{2}$$

$$z^{\mathrm{r}}_c(\mathbf{x}) = \textstyle\sum_k \rho_k \,\mathrm{softmax}(\mathbf{v}_k)_c \, o_k(\mathbf{x}) + b_c, \tag{3}$$

computed exactly in the forward pass; in the backward pass we use the DR-Net linear surrogate $1 - \tilde{v}_k(\mathbf{x})$ (with the continuous probabilities), weighted by $\exp(-v_k(\mathbf{x})/\tau_g)$, $\tau_g = 1$, so that a rule only adjusts its literals from the examples it covers or nearly covers. Here $\rho_k > 0$ is the rule weight, $\mathrm{softmax}(\mathbf{v}_k)$ its vote distribution and $\mathbf{b}$ the default decision. *Decoding.* Rule k reads "IF $\bigwedge_{j:a_{kj}=1}(x_j = 1) \wedge \bigwedge_{j:n_{kj}=1}(x_j = 0)$ THEN $\arg\max_c v_{kc}$", with support and precision computed on the training set; rules without literals are folded into $\mathbf{b}$ and rules never active on the training set are discarded. The resulting list reproduces z^{r} exactly (weighted faithfulness); the reading in which each rule votes only for its THEN class defines the strict faithfulness that we measure.

Counterfactual branch. The hypernetwork produces $W(\mathbf{x}) \in \mathbb{R}^{C\times(d+E)}$ and $\mathbf{b}(\mathbf{x})$; $\mathbf{z}^{\mathrm{loc}} = W(\mathbf{x})\mathbf{x} + \mathbf{b}(\mathbf{x})$ as in HyConEx. For a target $t \neq \hat{y}$, let $\delta = W_t(\mathbf{x}) - W_{\hat{y}}(\mathbf{x})$. For every mutable attribute g and every candidate value v, the gain $\delta \cdot (\mathbf{x}^{(g\leftarrow v)} - \mathbf{x})$ is computed exactly, with both the one-hot columns *and* the relational literals recomputed; we keep the best value per attribute, sort attributes by decreasing gain and return the shortest prefix reaching the local margin $z^{\mathrm{loc}}_t - z^{\mathrm{loc}}_{\hat{y}} \geq \kappa$ ($\kappa = 1$). The G prefixes of that path are then evaluated by the full model in a single batched forward pass, and the shortest one that reaches t is retained. The counterfactual is a valid categorical tuple by construction ("$\texttt{rmeta0}$: $64 \to 63$"), its sparsity is the number of modified attributes, and a mask of immutable attributes (by default the resource attributes) restricts modifications to what can actually be changed.

Fusion. The logits of both branches are combined by a weighted sum with learned, softmax-normalised weights:

$$\mathbf{z}^{\mathrm{fus}} = \alpha_h \mathbf{z}^{\mathrm{loc}} + \alpha_r \mathbf{z}^{\mathrm{r}}, \quad \hat{y} = \arg\max_c \mathrm{softmax}(\mathbf{z}^{\mathrm{fus}})_c. \tag{4}$$

Per-branch logits remain available for audit; their agreement rate is reported.

3.4 Loss and training

Training minimises a composite loss that jointly optimises classification, rule quality and counterfactual validity:

$$\begin{aligned}\mathscr{L} = {} & \mathrm{CE}(\hat{\mathbf{p}}, y) + \lambda_r \overline{\mathrm{CE}}(\hat{\mathbf{p}}^{\mathrm{r}}, y) + \lambda_h \mathrm{CE}(\hat{\mathbf{p}}^{\mathrm{loc}}, y) \\ & + \lambda_{\mathrm{cf}} \mathrm{CE}(f(\mathbf{x}'_t), t) + \lambda_{\mathrm{sp}} \overline{|\mathbf{p}|} \\ & + \lambda_{\mathrm{len}} \overline{\max(0, \ell_k - 8)} + \lambda_{\mathrm{ent}} \overline{H(\mathrm{softmax}(\mathbf{v}_k))},\end{aligned} \tag{5}$$

where $\overline{\mathrm{CE}}$ is averaged over the M draws, $\mathbf{x}'_t$ is the (detached) counterfactual towards a random target $t \neq y$, ℓ_k is the hard length of rule k (penalised beyond eight literals) and H the entropy of the vote (favouring single-class rules). Gradients are clipped (max norm 1) and parameters updated with AdamW [23] ($\eta = 10^{-3}$, cosine decay, $\lambda_{\mathrm{wd}} = 10^{-5}$), with a learning rate ten times larger for literal parameters, since under Adam crossing the 0.5 threshold would otherwise be too slow. The best epoch is selected on $\mathrm{acc}_{\mathrm{val}} + \mathrm{acc}^{\mathrm{r}}_{\mathrm{val}}$, the rule branch converging more slowly than the counterfactual branch. Hyperparameters (Table 4) were set on u4k-r4k-auth11k (seed 42) and then frozen; training runs for 30 epochs (20 on Amazon) with early stopping (patience 10).

At inference, RuleConEx returns in a single forward pass: (i) the decision $\hat{y}$; (ii) the global rules and those active on the request; (iii) the counterfactuals $\{\mathbf{x}'_t, \Delta_t\}$ for all targets; (iv) the per-branch logits.

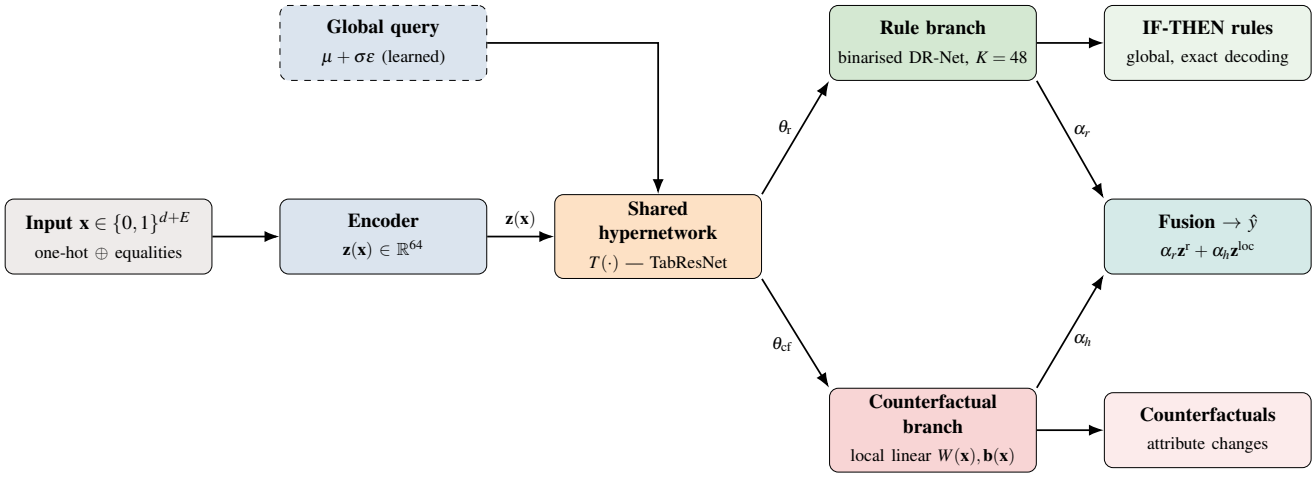

Fig. 1 Architecture of RuleConEx. The shared hypernetwork is queried twice: by a learned global query, independent of the request, which parameterises the rule branch — hence a single rule set for all requests — and by the local query $\mathbf{z}(\mathbf{x})$, which parameterises the counterfactual branch. One forward pass yields the decision, the decodable rules and the counterfactuals

Table 4 Main hyperparameters of RuleConEx

Component	Parameter	Value
Encoder	latent dimension	64
Hypernet.	trunk (blocks / width); σ; M	2 / 128; 0.1; 3 / 1
Rules	neurons K; embedding rank; τ_g	48; 32; 1
Rules	maximum penalised length	8
CF	local margin κ	1
Fusion	α_h, α_r (init.)	0.5 / 0.5
Loss	$\lambda_r, \lambda_h, \lambda_{\mathrm{cf}}$	1; 0.5; 0.1
Loss	$\lambda_{\mathrm{sp}}, \lambda_{\mathrm{len}}, \lambda_{\mathrm{ent}}$	10^{-3}; 0.1; 0.05
Optim.	AdamW, batch, epochs	10^{-3} ($\times$10 literals), 128, 30

4 Results

4.1 Experimental protocol

RuleConEx is implemented in Python/PyTorch 2.5; the complete source code (preprocessing, model, training, evaluation, benchmark, statistical tests) is available at `https://github.com/erickTsaf237/Master-work/tree/main/HyConEx_from_scratch/ruleconex2`. Experiments were run under Windows 10 with an NVIDIA GeForce 940MX GPU (2 GB).

Repetition and tests. Every (model, dataset) pair is trained with five seeds (42–46); the train/validation split is fixed, the seed affecting initialisation, batch order and noise. Tables report mean $\pm$ standard deviation. Models are compared with a Friedman test on the datasets $\times$ models matrix (means over seeds), with Iman–Davenport correction and Nemenyi critical distance ($\alpha = 0.05$) [10], and with a paired Wilcoxon test over the 55 (dataset, seed) pairs between RuleConEx and each other model, with Holm correction.

Compared models. All receive the same one-hot + equalities input: RuleConEx; HyperLogic (rule branch alone, after [38]); HyConEx (counterfactual branch alone, after [25]); **HyperLogic + HyConEx independent**, the two previous models trained separately, combining the rules of one with the counterfactuals of the other; an MLP (opaque reference, 2×256); an MLP on the one-hot encoding only; a random forest (200 trees); a CART decision tree; and a linear SVM. The published DLBAC figures do not take part in the ranking (different protocol, hardware and computing budget).

Metrics. Classification: accuracy, macro F1, macro one-vs-rest AUC. Rules: number of active rules, length, coverage (fraction of test requests on which at least one rule fires), strict and weighted faithfulness (agreement between the decoded rules and the final decision), accuracy of the rules alone, precision, redundancy and across-seed stability (Jaccard). Counterfactuals, over 1,000 test requests (500 on Amazon) and all targets: validity, flip success, sparsity, plausibility (Hamming distance to the nearest training example of the target class, compared with that of a genuine request), and the "immutable resource" regime. Rule–counterfactual consistency: `cons_explained` (valid counterfactuals on which a rule of the target class fires), `cons_precision` (modified attributes read by those rules), and agreement between branches. Time: training, decision latency, rule decoding, counterfactual generation.

4.2 Classification performance

Table 5 reports accuracy per dataset. The Friedman test rejects equality ($\chi^2 = 45.6$, $p = 10^{-7}$; Iman–Davenport $p = 2 \cdot 10^{-11}$); with a Nemenyi critical distance of 3.17, RuleConEx (mean rank 4.18) is separable neither from the MLP (1.95), nor from RF (2.73), SVM (3.68) or HyConEx (4.27),

Table 5 Accuracy (%) per DLBAC dataset: mean ± standard deviation over five seeds. Bold: best model on that dataset. Maj.: accuracy of a constant majority-class classifier. HypLog: HyperLogic (rule branch alone); HyConEx: counterfactual branch alone; MLP_{1h}: MLP without relational literals. Last line: mean Friedman rank (1 = best)

Dataset	Maj.	RuleConEx	HypLog	HyConEx	MLP	MLP_{1h}	RF	DT	SVM
amazon1	*94.1*	94.4 ±0.2	94.1 ±0.0	94.2 ±0.2	94.4 ±0.1	94.5 ±0.1	**94.7 ±0.1**	93.9 ±0.1	94.2 ±0.0
amazon2	*94.4*	94.6 ±0.1	94.4 ±0.0	94.4 ±0.1	94.6 ±0.2	94.6 ±0.2	**94.9 ±0.1**	94.1 ±0.1	93.9 ±0.0
amazon3	*94.2*	94.2 ±0.3	94.2 ±0.0	94.6 ±0.0	94.6 ±0.1	94.7 ±0.1	**94.8 ±0.0**	94.4 ±0.1	93.8 ±0.0
u4k-r4k-11k	*13.0*	98.9 ±0.1	88.1 ±3.2	98.6 ±0.3	98.9 ±0.1	97.8 ±0.1	**99.2 ±0.1**	97.7 ±0.1	98.9 ±0.0
u4k-r4k-21k	*15.5*	92.3 ±0.6	85.2 ±0.9	92.2 ±0.5	93.0 ±0.2	89.6 ±0.3	92.3 ±0.1	88.6 ±0.0	**93.6 ±0.0**
u4k-r4k-22k	*21.1*	74.6 ±0.9	61.0 ±0.8	74.0 ±0.8	76.0 ±0.3	72.9 ±0.8	73.9 ±0.1	67.1 ±0.3	**76.6 ±0.0**
u4k-r6k-28k	*12.0*	84.7 ±0.3	67.8 ±2.0	84.8 ±0.3	**86.7 ±0.4**	85.3 ±0.3	86.3 ±0.1	81.5 ±0.2	83.6 ±0.0
u4k-r7k-20k	*16.2*	91.9 ±0.7	88.5 ±2.6	91.8 ±0.2	**92.6 ±0.1**	89.2 ±0.5	89.7 ±0.1	88.6 ±0.1	92.1 ±0.0
u5k-r5k-12k	*12.5*	99.8 ±0.1	94.2 ±4.0	99.8 ±0.1	100.0 ±0.0	99.4 ±0.2	**100.0 ±0.0**	99.1 ±0.1	100.0 ±0.0
u5k-r5k-19k	*20.6*	93.3 ±0.5	88.3 ±2.8	93.5 ±0.2	**94.4 ±0.2**	90.4 ±0.3	92.4 ±0.0	90.8 ±0.1	94.4 ±0.0
u6k-r6k-32k	*15.1*	94.5 ±0.4	88.5 ±1.3	94.3 ±0.2	94.8 ±0.2	92.3 ±0.2	94.1 ±0.1	92.9 ±0.1	**95.2 ±0.0**
Mean	—	92.1	85.9	92.0	92.7	91.0	92.0	89.9	92.4
Mean rank	—	4.18	7.55	4.27	1.95	4.91	2.73	6.73	3.68

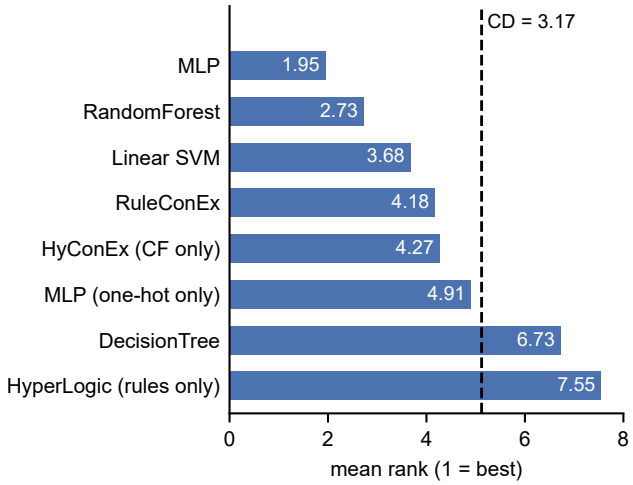

Fig. 2 Mean Friedman ranks over the eleven datasets (accuracy). Any model to the right of the dashed line is significantly worse than the best one in the sense of Nemenyi ($\alpha = 0.05$)

but is separable from the decision tree (6.73) and from HyperLogic alone (7.55), as shown in Fig. 2. The paired Wilcoxon test refines this picture: RuleConEx is *indistinguishable* from HyConEx ($p = 0.98$), from the random forest ($p = 0.98$) and from the SVM; it is *inferior* to the MLP by 0.63 points ($p_{\text{Holm}} = 4 \cdot 10^{-8}$, 47 losses out of 55); and it is *superior* to HyperLogic alone (+6.2 points), to the decision tree (+2.2) and to the MLP without equalities (+1.1), all at $p_{\text{Holm}} < 10^{-4}$. Standard deviations are below 0.01 except for HyperLogic alone (0.01–0.04): the rule branch alone is unstable, and fusion stabilises it. The same conclusions hold for macro F1 (RuleConEx 0.828, MLP 0.841; −0.013, $p = 4 \cdot 10^{-4}$) and AUC (0.948 vs 0.953; −0.005, $p = 10^{-8}$). The price of intrinsic interpretability, measured against the best opaque model trained under the same conditions, is therefore about half a point of accuracy: significant, but small.

One remark on the baselines: the decision tree reaches 0.899 on average, whereas earlier comparisons on the same data reported it below 0.50; the difference stems entirely from the relational literals (Table 3), which calls for caution when reading published DLBAC/tree comparisons.

Class imbalance on Amazon: accuracy measures nothing there. The three Amazon datasets contain 94% granted accesses. The "Maj." column of Table 5 gives the accuracy of a constant classifier always answering *grant*: 94.1 to 94.4%. Every evaluated model lies within 0.8 points of that value, and some below it. On these datasets accuracy therefore separates no model and must not be read as a measure of quality; only AUC and macro F1, which weight the minority *deny* class, are informative. Table 6 reports them. The gap is considerable: the random forest reaches 0.856 AUC and RuleConEx 0.822, whereas HyperLogic trained alone remains at chance level (AUC 0.504, macro F1 0.486) while displaying 94.2% accuracy — it predicts *grant* for almost every request. This is also the clearest argument in favour of joint training: the same rule branch, trained together with the counterfactual branch, moves the model from 0.504 to 0.822 AUC. We therefore draw no accuracy-based conclusion on Amazon, and report these datasets mainly for the robustness of the explanations to a one-hot cardinality of order 10^4.

4.3 Rule quality

Table 7 summarises the explanation metrics. On the eight synthetic datasets, RuleConEx decodes 46 to 48 active rules of 4.8 to 6.8 literals, almost exclusively relational, covering 92 to 100% of the test requests, with a strict faithfulness of 0.84 to 0.91 on six datasets (0.49 and 0.58 on the two hardest, where the counterfactual branch dominates the fusion); used alone, these rules reach 0.60 to 0.93 accuracy.

Table 6 Amazon datasets (means over the three datasets and five seeds). The italic line is a constant *grant* classifier. Accuracy does not separate the models; AUC and macro F1 do

Model	Accuracy (%)	F1 macro	AUC
Majority class	*94.2*	*0.485*	*0.500*
RandomForest	94.8	0.659	0.856
MLP	94.6	0.641	0.837
MLP_{1h}	94.6	0.645	0.836
HyConEx	94.4	0.620	0.828
RuleConEx	94.4	0.614	0.822
SVM	94.0	0.678	0.817
DecisionTree	94.2	0.706	0.688
HyperLogic	94.2	0.486	0.504

Table 7 Explanation metrics (means over 11 datasets × 5 seeds). HL: HyperLogic alone; HC: HyConEx alone; Indep.: rules from HL combined with counterfactuals from HC

	HL / HC	Indep.	RuleConEx
Active rules (out of 48)	35.6	35.6	39.9
Mean length (literals)	4.80	4.80	4.51
Coverage	0.736	0.736	0.796
Faithfulness strict / weighted	0.750 / 0.873	0.750 / 0.873	0.804 / 0.838
Accuracy of rules alone	0.739	0.739	0.819
Mean rule precision	0.545	0.545	0.653
Redundancy (Jaccard)	0.202	0.202	0.248
CF validity	0.988	0.988	0.958
Flip success	1.000	1.000	1.000
CF sparsity (attributes)	1.95	1.95	1.95
CF plausibility (Hamming; ref. 3.55)	7.89	7.89	7.98
CF validity, immutable resource	0.758	0.758	0.682
`cons_explained`	—	0.222	0.183
`cons_precision`	—	0.612	0.563
Agreement between branches	—	0.875	0.850

Compared with those of a HyperLogic model trained alone, they are shorter (4.5 vs 4.8 literals) and more precise (+0.05 to +0.10 per dataset) at equal coverage; on Amazon, HyperLogic alone finds almost no rule (coverage 2 to 17%) whereas joint training finds 10 to 26 (coverage 32 to 45%). Across-seed stability is 0.55 to 0.73 (Jaccard) on the synthetic datasets, identical to HyperLogic alone, and 0.30 to 0.62 on Amazon (HyperLogic alone: 0.00–0.18).

The rules read as ABAC policies. On the synthetic datasets the label `ops_pattern_`c encodes the four operations in binary ($c = op_1 + 2\,op_2 + 4\,op_3 + 8\,op_4$). Thus `ops_pattern_12` grants op_3 and op_4, while `ops_pattern_0` is a full denial. Three rules decoded on u4k-r4k-auth11k (seed 42; support and precision on the training set):

```
R20: IF umeta0=rmeta0 AND umeta1=rmeta1
     AND umeta4=rmeta4 AND umeta0!=rmeta4
     AND umeta2!=rmeta2
   THEN ops_pattern_12 = {op3, op4}
   (support=381, precision=1.00)
```

```
R30: IF umeta1=rmeta1 AND umeta3=rmeta3
     AND umeta7=rmeta0 AND umeta5!=rmeta5
   THEN ops_pattern_10 = {op2, op4}
   (support=336, precision=0.91)
```

```
R22: IF umeta0!=rmeta0 AND umeta2!=rmeta1
     AND umeta4!=rmeta4 AND umeta5!=rmeta5
   THEN ops_pattern_0 = none
   (support=711, precision=1.00)
```

The first two grant operations when the user and the resource share several attributes, the third denies all access in the absence of such matches; replacing $\texttt{umeta}_i$ by the attribute name (department, role, ...) is enough to make them readable by an administrator.

4.4 Counterfactuals

Counterfactuals are valid in 86 to 97% of cases on the synthetic datasets (mean 0.958; 100% on Amazon), at least one target is reached for 100% of the requests, and they modify 2.1 to 2.5 attributes (1.0 to 1.1 on Amazon). HyConEx alone is slightly more valid (0.988) at equal sparsity: fusing with the rule branch makes the local linear classifier a little less faithful to the full model. Plausibility (8.2 to 9.4 attributes out of 16) remains clearly above that of a genuine request (3.4 to 4.2), as it does for HyConEx alone (7.9). Under the "immutable resource" regime, validity drops to 0.68 (HyConEx 0.76): part of the accesses depends on user–resource equalities. Examples (seed 42; classes decoded into operations as in Section 4.3):

```
u4k-r4k-auth11k, request #3
decision: full denial
  -> {op2} : rmeta0 : 64 -> 63 [valid]
  -> {op3} : rmeta0 : 64 -> 31 [valid]
  -> {op1} : rmeta7 : 51 -> 89 [NOT valid, flagged]
u5k-r5k-auth19k, request #3
decision: {op1, op3, op4}
  -> {op1,op2,op3,op4} : rmeta3 : 75 -> 9 [valid]
  -> full denial : umeta4 : 3 -> 20 [valid]
```

4.5 Rule–counterfactual consistency

Over the 55 (dataset, seed) pairs, with the default setting ($\lambda_{\text{cf}} = 0.1$) favouring short counterfactuals, the fraction of valid counterfactuals on which a rule of the target class fires is 0.18 for RuleConEx against 0.22 for the "HyperLogic + HyConEx independent" juxtaposition ($p = 0.005$); the fraction of modified attributes read by those rules is 0.56 against 0.61 ($p = 0.001$); the agreement between branches is 0.85 against 0.88 ($p = 2 \cdot 10^{-6}$). These gaps are small and, as the ablation shows, do not come from sharing the hypernetwork but from the counterfactual loss term, which governs the balance between short counterfactuals and consistency: relaxing it makes RuleConEx far exceed the juxtaposition (0.43).

4.6 Training and inference time

Table 8 separates training from inference. RuleConEx trains in 213 s per dataset on average; at inference a decision costs

Table 8 Mean times over datasets and seeds: training (s), single-request decision latency on GPU (ms), rule decoding (ms), counterfactuals towards all targets (ms per request), parameters

Model	Train.	Decision	Rules	Counterf.	Param.
MLP	11	—	—	—	0.9 M
Linear SVM	2	—	—	—	—
RandomForest	106	—	—	—	—
DecisionTree	477	—	—	—	—
HyperLogic	55	3.8	22	—	1.1 M
HyConEx	122	1.9	—	6.4	1.7 M
Independent	177	—	22	6.4	2.7 M
RuleConEx	213	5.1	26	6.3	2.2 M

5.1 ms in single-request latency (0.03 ms in batch), full rule decoding takes 26 ms and counterfactuals towards all classes 6 ms per request. The overhead is therefore entirely at training time; decision and explanations fit within a few milliseconds, compatible with online access control.

4.7 Ablation study

Table 9 compares the full model with nine variants (12 pairs; paired Wilcoxon, Holm correction). Five lessons follow. *(1) The shared hypernetwork is neutral*: two separate trunks change neither accuracy ($p = 1$) nor counterfactual validity ($p = 0.55$), and slightly reduce consistency (0.135 vs 0.176, not significant) and faithfulness; it is joint training, rather than weight sharing, that carries the observed benefits. *(2) Gaussian noise is superfluous*: with $M = 1$, $\sigma = 0$, all indicators are equal or better (faithfulness 0.82) for 25% shorter training. *(3) Accuracy comes from the counterfactual branch, and rules cost nothing in accuracy*: without the counterfactual branch, -7.6 points ($p_{\text{Holm}} = 0.009$); without the rule branch, accuracy is unchanged and counterfactual validity rises from 0.971 to 0.992. *(4) The counterfactual loss governs the trade-off between short counterfactuals and rule–counterfactual consistency*: without λ_{cf}, accuracy is unchanged and counterfactuals remain valid (0.928), but they modify 3.7 attributes instead of 1.9; in exchange, coverage rises to 0.86, faithfulness to 0.83, and 43% of the counterfactuals are justified by a rule of the target class ($p_{\text{Holm}} = 0.01$), nearly twice the juxtaposition (0.22). This coefficient is in fact a continuous control, characterised in Section 4.8. *(5) The literal language determines rule quality*: without relational literals, -2.7 points ($p_{\text{Holm}} = 0.02$) and collapsed rules (accuracy 0.49, 1.8 literals); without negations, rules of 2.4 literals that are less faithful (0.71) and less consistent; $K = 24$ is not enough (coverage 0.71) while $K = 96$ improves the rules (0.85, coverage 0.82, consistency 0.24) for a non-significant -0.6 points.

4.8 Tuning the user/auditor trade-off

The term λ_{cf} pushes the model to classify in the target class the tuples obtained by modifying one or two attributes. It therefore brings the decision boundaries closer to each request, which shortens counterfactuals but makes the learned function harder to describe with conjunctions. We swept it over seven values ($0 \leq \lambda_{\text{cf}} \leq 0.1$, logarithmic scale) on the four ablation datasets and three seeds, comparing variants on exactly the same (dataset, seed) pairs (Table 10).

A continuous trade-off, with no detectable effect on accuracy. Both quantities vary monotonically: counterfactual sparsity goes from 3.66 to 1.92 attributes and standardised consistency from 0.528 to 0.187, most of the variation occurring below 10^{-2}. Accuracy stays between 0.903 and 0.905 across the seven settings, and no paired Wilcoxon test against the default setting is significant ($p \geq 0.47$): with twelve (dataset, seed) pairs we therefore detect no difference in accuracy between settings, which does not rule out an effect below our experimental resolution. The trade-off thus does not oppose explainability to performance, but two forms of explainability to each other: a high λ_{cf} serves the denied user, who is owed a short and feasible modification; a low λ_{cf} serves the auditor, who is owed covering rules of which the counterfactuals are instances. The setting $\lambda_{\text{cf}} = 10^{-3}$ occupies a defensible middle ground: 2.94 modified attributes for a consistency of 0.247, significantly above the default ($+0.071$, $p = 0.034$), and a rule faithfulness of 0.822 against 0.758. To our knowledge no intrinsic explanation model provides such a control: two juxtaposed models offer no parameter that could bring their explanations closer.

The consistency gain is not an artefact of length. A counterfactual that modifies more attributes mechanically has more chances of activating a rule of the target class; the high consistency of $\lambda_{\text{cf}} = 0$ could therefore be a mere length effect. We rule this out in two ways. First, the gap persists at a fixed number of modified attributes: at one attribute, 0.622 against 0.209; at two, 0.517 against 0.179; at three or more, 0.403 against 0.166 (last three columns of Table 10). Second, standardising each setting on the sparsity profile of the default setting *increases* the gap instead of reducing it (0.528 against 0.187), because at $\lambda_{\text{cf}} = 0$ consistency *decreases* with counterfactual length: the shortest counterfactuals are the best justified by a rule, the opposite of what the suspected artefact would produce.

5 Explanations as an attack surface

In an explainable AI setting, an explanation is a good in itself. In a security setting it is also an attack surface, and an access control system that publishes its reasoning must state

Table 9 Ablation (u4k-r4k-auth11k, u4k-r4k-auth22k, u5k-r5k-auth19k, amazon1; 3 seeds; means). Rule acc.: accuracy of the rule branch alone; Faith.: strict faithfulness; Cov.: coverage; Len.: literals per rule; CF val.: validity; Spars.: modified attributes; Cons.: `cons_explained`

Variant	Acc.	Rule acc.	Faith.	Cov.	Len.	CF val.	Spars.	Cons.
Full RuleConEx	0.903	0.821	0.758	0.753	4.48	0.971	1.92	0.176
– shared hypernetwork	0.902	0.813	0.721	0.721	4.33	0.963	1.98	0.135
– Gaussian noise ($M = 1$)	0.905	0.820	0.815	0.751	4.25	0.974	1.97	0.183
– rule branch	0.901	—	—	—	—	0.992	1.99	—
– counterfactual branch	0.827	0.827	0.675	0.774	5.05	—	—	—
– counterfactual loss ($\lambda_{\text{cf}} = 0$)	0.904	0.823	0.832	0.861	4.71	0.928	3.66	0.430
– relational literals	0.876	0.486	0.401	0.482	1.75	0.988	1.78	0.109
– negations ($\neq$)	0.902	0.820	0.711	0.752	2.39	0.972	1.96	0.157
$K = 24$	0.904	0.761	0.647	0.706	4.55	0.970	1.92	0.114
$K = 96$	0.897	0.848	0.731	0.824	4.45	0.956	1.89	0.242

Table 10 Sweep of the counterfactual loss weight λ_{cf} (4 datasets $\times$ 3 seeds, paired comparisons). *Raw consistency*: fraction of valid counterfactuals justified by a rule of the target class; *standardised*: the same, reweighted by the sparsity profile of the default setting; the last three columns give it at a fixed number of modified attributes

λ_{cf}	Accuracy	Modified	CF	Consistency		Consistency by sparsity		
		attributes	validity	raw	standard.	1 attr.	2 attr.	$\geq$3 attr.
0	0.904	3.66	0.928	0.430	0.528	0.622	0.517	0.403
0.001	0.903	2.94	0.956	0.247	0.307	0.371	0.273	0.260
0.003	0.905	2.70	0.954	0.229	0.270	0.315	0.234	0.255
0.01	0.903	2.41	0.955	0.205	0.250	0.267	0.250	0.225
0.02	0.903	2.29	0.950	0.174	0.211	0.237	0.200	0.188
0.05	0.903	2.11	0.953	0.182	0.207	0.234	0.195	0.185
0.1 (default)	0.903	1.92	0.971	0.176	0.187	0.209	0.179	0.166

who may see what. RuleConEx exposes two very different objects, which we argue call for different privileges.

Global rules. The decoded rule set is, by construction, a readable description of the learned policy: 48 conjunctions over user–resource attribute equalities, with support and precision. For an auditor this is exactly the intended artefact. For an outsider it is a policy disclosure: it reveals which attribute combinations grant which operations, that is, precisely the information needed to target an identity or a resource whose attributes should be forged. Rules must therefore be treated as an administrative artefact, released to the auditor and to the administrator, and not returned with the decision of an individual request.

Counterfactuals. The counterfactual answers "what should change to obtain this access". This is the information the denied user legitimately needs, but each query also reveals one point of the decision boundary. An adversary able to query the system repeatedly obtains, for the price of one forward pass per request, a sequence of (request, minimal modification) pairs that constitutes an efficient oracle for reconstructing the policy — a model extraction attack specialised to access control, and considerably cheaper than probing the decision alone, since every answer carries the direction and the magnitude of the change. Two mitigations follow directly from our design. The immutable-attribute mask already restricts answers to the attributes the requester may legitimately change, which both improves actionability and reduces leakage about resource attributes; and limiting each answer to the target class actually requested, rather than returning counterfactuals for all classes as our evaluation does, divides the information disclosed per query by the number of classes. Rate limiting and auditing of counterfactual queries remain necessary.

Threat model and privilege separation. We therefore consider three roles. The *auditor* obtains the global rules and the per-branch logits, and is trusted with the policy. The *administrator* obtains the same objects plus the counterfactuals for the requests under review. The *requester* obtains the decision and, upon denial, a single counterfactual restricted to mutable attributes. Under this separation, the information disclosed to an untrusted party is bounded by one minimal modification per denied request. We do not claim that this bound is sufficient: quantifying how many counterfactual queries are needed to recover a given fraction of the policy, and designing a privacy-preserving counterfactual mechanism, are open questions that we leave to future work. We stress that the problem is not specific to RuleConEx — any actionable counterfactual method deployed on an access control system faces it — but that it must be stated, which the explainability literature on access control has so far not done.

6 Discussion

Scope of the results. Among the models compared under identical conditions, RuleConEx is the only one that combines exact global rules with valid categorical counterfactuals, with an accuracy indistinguishable from HyConEx, random forests and SVM, and 0.6 points below the opaque MLP. Its contribution therefore does not lie in a gain in accuracy, but in this combination and in the fact that the quality of each explanation is measured. Whether the gap with the MLP constitutes an "acceptable trade-off" is a qualitative judgement, defended in the literature on intrinsically interpretable models [30], but one that remains to be confirmed with real users.

The central contribution: a tunable trade-off, not an acquired property. The initial intuition — that sharing the hypernetwork alone would make rules and counterfactuals consistent — is too simple: the ablation shows that this sharing is neutral. What the experiments do establish is more interesting and, in our view, more defensible: consistency between the two explanations is governed by a single coefficient of the loss, whose sweep continuously shifts the trade-off between user-oriented and auditor-oriented explainability, with no difference in accuracy detectable on our protocol (Section 4.8). This is, from our point of view, the main contribution of this work: not a more accurate model, nor explanations that are consistent by construction, but a single model in which the nature of the explanation becomes a design parameter, assumed and measured, that the administrator sets according to whether the system serves the denied user or the auditor. Two juxtaposed models offer no equivalent control.

This reading calls for two caveats. First, the consistency gain is paid for in counterfactual length (3.66 against 1.92 attributes at $\lambda_{\text{cf}} = 0$); it is a shift of the trade-off, not a free improvement. Second, even at the best setting consistency plateaus at 0.43: most counterfactuals are still not justified by a rule of the target class. Making consistency structural rather than tunable — by selecting the counterfactual among the modifications that activate a rule of the target class — remains the natural continuation of this work.

On the role of relational literals. The most transferable result is perhaps Table 3: DLBAC policies are made of user–resource constraints, and any symbolic model deprived of them is condemned to long and imprecise rules; the language of the rules must be that of the target policy, not that of the encoding.

Comparison with DLBAC, HyperLogic and HyConEx. The published DLBAC figures (93.15% mean accuracy [26]) remain indicative (different architectures, hardware and preprocessing); our local baselines, led by the MLP (92.7%), are close to them, and RuleConEx is 0.6 points behind. This position matches the observation of Nobi et al. [26], for whom explainability is the research priority of DLBAC, and provides an intrinsic answer to it. HyperLogic reports faithfulness of one by construction; we measure it (0.75–0.80 strict, 0.84–0.87 weighted) because the rule branch is not the sole decider. HyConEx reports 99.9% validity on continuous vectors; our 95.8% (98.8% for the branch alone) are on valid categorical tuples. The local reimplementations obtain accuracy equal (HyConEx) or inferior (HyperLogic) to that of RuleConEx; neither had been evaluated on multi-class access control data.

Comparison with post-hoc approaches. Against post-hoc counterfactual methods (CEM, FACE, CEGP, PPCEF), which optimise each instance at a cost that can reach several hours, RuleConEx produces its counterfactuals in a single forward pass; its overhead is at training time, not at inference (5 ms). Against EXTree and LANTERN, which are decoupled from the decision model, it offers explanations that are faithful by construction and expressed in the ABAC language itself.

7 Conclusion

Deep learning based access control systems are accurate but opaque, post-hoc approaches lack faithfulness, and existing self-explaining models each cover a single kind of explanation. By letting HyperLogic and HyConEx share their central mechanism, the hypernetwork, RuleConEx produces in a single forward pass an access decision, exact IF-THEN rules in the ABAC language, and valid categorical counterfactuals. On the eleven DLBAC datasets and five seeds it reaches 92.1% mean accuracy, indistinguishable from HyConEx, random forests and SVM and 0.6 points below the opaque MLP; its rules cover 80% of the requests with a strict faithfulness of 0.80 and 96% of its counterfactuals are valid, modifying two attributes.

Its main result, however, lies elsewhere: consistency between the two explanations does not follow from weight sharing, but is governed by the weight of the counterfactual loss. A sweep over seven values shows that this single coefficient continuously shifts the trade-off between user-oriented explainability (counterfactuals of 1.9 attributes) and auditor-oriented explainability (covering rules justifying 43% of the counterfactuals, against 18% at the default setting and 22% for two juxtaposed models), with no statistically detectable difference in accuracy; we verify that this gain is not an artefact of counterfactual length. The experiments further establish that the HyperLogic noise is superfluous and that the literal language (user–resource equalities and their negations) is the decisive factor for rule quality. On the heavily

imbalanced Amazon datasets, finally, accuracy separates no model and only AUC and macro F1 are informative.

Limitations. Hyperparameters were set on a single dataset and frozen; the ablation explores nine variants and λ_{cf} seven values, but the other loss coefficients were not swept, and the trade-off was characterised on only four of the eleven datasets. Counterfactuals are valid but of limited plausibility with respect to the data, and their validity drops when resource attributes are immutable. On Amazon, rules cover only a third of the requests and some, of low precision, should be filtered before display. Since DLBAC attributes are anonymous (`umeta3`), the semantic rendering of rules could not be demonstrated. Finally, neither the perceived quality of the explanations nor their usefulness to an administrator or an auditor has been evaluated with expert users; the results remain a quantitative evaluation on benchmark datasets, which cannot by itself establish GDPR compliance, of which explainability is only one dimension. The analysis of Section 5 is likewise qualitative: we state the attack surface without quantifying it.

Perspectives. The most direct continuation is to characterise the tuning of λ_{cf}, then to make consistency structural by constraining the counterfactual with the rules (or conversely), and to add a plausibility term to the choice of values. Linking the attributes $\texttt{umeta}_i$ to the metadata of a real system would make the rules immediately readable (`department` $=$ `Engineering` and `role_level` $\neq$ `admin`); converting them into Or-BAC or HOr-BAC permission relations would bridge deep learning and deployed formal systems, as suggested by Nobi et al. [26]. Fairness constraints in the loss would penalise rules discriminating on protected attributes [5]. Quantifying policy leakage through counterfactual queries, as discussed in Section 5, is an open problem of its own. Finally, a study with real administrators or auditors is necessary to establish whether the rules and counterfactuals produced are understandable and useful in an audit context.

Statements and Declarations

Funding. The authors did not receive support from any organisation for the submitted work.

Competing interests. The authors have no competing interests to declare that are relevant to the content of this article.

Ethics approval. Not applicable. This study does not involve human participants or animals.

Data availability. All datasets analysed during this study are publicly available benchmarks released with DLBAC [26] at `https://github.com/dlbac/DlbacAlpha`. The preprocessing scripts that reproduce the exact splits used here are included in the code repository below. No new data were generated.

Code availability. The complete source code — preprocessing, model, training, evaluation, benchmark runner and statistical analysis — is openly available at `https://github.com/erickTsaf237/Master-work/tree/main/HyConEx_from_scratch/ruleconex2`.

Author contributions. Conceptualisation: B. M. Azanguezet Quimatio and E. Tsafack Nteudem; methodology and software: E. Tsafack Nteudem; formal analysis and investigation: E. Tsafack Nteudem and P. V. Nguezet; writing — original draft: E. Tsafack Nteudem; writing — review and editing: B. M. Azanguezet Quimatio and P. V. Nguezet; supervision: B. M. Azanguezet Quimatio. All authors read and approved the final manuscript.

References

1. Agarwal, R., Melnick, L., Frosst, N., Zhang, X., Lengerich, B., Caruana, R., Hinton, G.: Neural additive models: interpretable machine learning with neural nets. In: Advances in Neural Information Processing Systems 34 (2021). `https://doi.org/10.48550/arXiv.2004.13912`
2. Arik, S.Ö., Pfister, T.: TabNet: attentive interpretable tabular learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 6679–6687 (2021). `https://doi.org/10.1609/aaai.v35i8.16826`
3. Arya, V., Bellamy, R.K.E., Chen, P.-Y., et al.: AI Explainability 360: impact and design. arXiv:2109.12151 (2021). `https://doi.org/10.48550/arXiv.2109.12151`
4. Azanguezet Quimatio, B.M., Tindo, G., Fotso, L.P.: HOr-BAC: an access control based on hierarchical organizational. Int. J. Comput. Sci. Inf. Technol. Res. (2016)
5. Balagopalan, A., Zhang, H., Hamidieh, K., Hartvigsen, T., Rudzicz, F., Ghassemi, M.: The road to explainability is paved with bias: measuring the fairness of explanations. In: ACM Conference on Fairness, Accountability, and Transparency, pp. 1194–1206 (2022). `https://doi.org/10.1145/3531146.3533179`
6. Chowdary, S.P., Sural, S.: EXTree: towards supporting explainability in attribute-based access control. arXiv:2604.12850 (2026). `https://doi.org/10.48550/arXiv.2604.12850`
7. Cohen, W.W.: Fast effective rule induction. In: Proceedings of the 12th International Conference on Machine Learning, pp. 115–123 (1995). `https://doi.org/10.1016/B978-1-55860-377-6.50023-2`
8. Cotrini, C., Weghorn, T., Basin, D.: Mining ABAC rules from sparse logs. In: IEEE European Symposium on Security and Privacy, pp. 31–46 (2018). `https://doi.org/10.1109/EuroSP.2018.00011`
9. Dash, S., Günlük, O., Wei, D.: Boolean decision rules via column generation. In: Advances in Neural Information Processing Systems 31 (2018). `https://doi.org/10.48550/arXiv.1805.09901`

10. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. J. Mach. Learn. Res. **7**, 1–30 (2006)
11. Dhurandhar, A., Chen, P.-Y., Luss, R., Tu, C.-C., Ting, P., Shanmugam, K., Das, P.: Explanations based on the missing: towards contrastive explanations with pertinent negatives. In: Advances in Neural Information Processing Systems 31 (2018). https://doi.org/10.48550/arXiv.1802.07623
12. Goodman, B., Flaxman, S.: European Union regulations on algorithmic decision-making and a "right to explanation". AI Mag. **38**(3), 50–57 (2017). https://doi.org/10.1609/aimag.v38i3.2741
13. Gorishniy, Y., Rubachev, I., Khrulkov, V., Babenko, A.: Revisiting deep learning models for tabular data. In: Advances in Neural Information Processing Systems 34 (2021). https://doi.org/10.48550/arXiv.2106.11959
14. Guidotti, R.: Counterfactual explanations and how to find them: literature review and benchmarking. Data Min. Knowl. Discov. **38**, 2770–2824 (2024). https://doi.org/10.1007/s10618-022-00831-6
15. Ha, D., Dai, A., Le, Q.V.: HyperNetworks. arXiv:1609.09106 (2016). https://doi.org/10.48550/arXiv.1609.09106
16. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016). https://doi.org/10.1109/CVPR.2016.90
17. Hu, V.C., Ferraiolo, D., Kuhn, R., et al.: Guide to attribute based access control (ABAC) definition and considerations. NIST Special Publication 800-162 (2014). https://doi.org/10.6028/NIST.SP.800-162
18. Kadra, A., Janowski, S., Wistuba, M., Grabocka, J.: Interpretable mesomorphic networks for tabular data. In: Advances in Neural Information Processing Systems 37 (2024). https://doi.org/10.52202/079017-0998
19. Abou El Kalam, A., Baida, R.E., Balbiani, P., et al.: Organization based access control. In: IEEE International Workshop on Policies for Distributed Systems and Networks, pp. 120–131 (2003). https://doi.org/10.1109/POLICY.2003.1206966
20. Karimi, L., Joshi, J.: An unsupervised learning based approach for mining attribute based access control policies. In: IEEE International Conference on Big Data, pp. 1427–1436 (2018). https://doi.org/10.1109/BigData.2018.8622037
21. Kumar, G., Sundaram, R., Sural, S.: Generation of human comprehensible access control policies from audit logs. arXiv:2603.14341 (2026). https://doi.org/10.48550/arXiv.2603.14341
22. Liu, A., Du, X., Wang, N.: Efficient access control permission decision engine based on machine learning. Secur. Commun. Netw. **2021**, 1–11 (2021). https://doi.org/10.1155/2021/3970485
23. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: International Conference on Learning Representations (2019). https://doi.org/10.48550/arXiv.1711.05101
24. Lundberg, S.M., Lee, S.-I.: A unified approach to interpreting model predictions. In: Advances in Neural Information Processing Systems 30 (2017). https://doi.org/10.48550/arXiv.1705.07874
25. Marszałek, P., et al.: HyConEx: hypernetwork classifier with counterfactual explanations for tabular data. Neurocomputing (2026). https://doi.org/10.1016/j.neucom.2026.132748
26. Nobi, M.N., Krishnan, R., Huang, Y., Shakarami, M., Sandhu, R.: Toward deep learning based access control. In: Proceedings of the 12th ACM Conference on Data and Application Security and Privacy, pp. 143–154 (2022). https://doi.org/10.1145/3508398.3511497
27. Poyiadzi, R., Sokol, K., Santos-Rodriguez, R., De Bie, T., Flach, P.: FACE: feasible and actionable counterfactual explanations. In: AAAI/ACM Conference on AI, Ethics, and Society, pp. 344–350 (2020). https://doi.org/10.1145/3375627.3375850
28. Qiao, L., Wang, W., Lin, B.: Learning accurate and interpretable decision rule sets from neural networks. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 4303–4311 (2021). https://doi.org/10.1609/aaai.v35i5.16555
29. Ribeiro, M.T., Singh, S., Guestrin, C.: "Why should I trust you?": explaining the predictions of any classifier. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1135–1144 (2016). https://doi.org/10.1145/2939672.2939778
30. Rudin, C.: Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nat. Mach. Intell. **1**(5), 206–215 (2019). https://doi.org/10.1038/s42256-019-0048-x
31. Sandhu, R.S., Coyne, E.J., Feinstein, H.L., Youman, C.E.: Role-based access control models. Computer **29**(2), 38–47 (1996). https://doi.org/10.1109/2.485845
32. Slack, D., Hilgard, S., Jia, E., Singh, S., Lakkaraju, H.: Fooling LIME and SHAP: adversarial attacks on post hoc explanation methods. In: AAAI/ACM Conference on AI, Ethics, and Society, pp. 180–186 (2020). https://doi.org/10.1145/3375627.3375830
33. Van Looveren, A., Klaise, J.: Interpretable counterfactual explanations guided by prototypes. In: Machine Learning and Knowledge Discovery in Databases (ECML PKDD), pp. 650–665 (2021). https://doi.org/10.1007/978-3-030-86520-7_40
34. Wachter, S., Mittelstadt, B., Russell, C.: Counterfactual explanations without opening the black box: automated decisions and the GDPR. Harv. J. Law Technol. **31**(2), 841–887 (2017). https://doi.org/10.2139/ssrn.3063289
35. Wang, T., Rudin, C., Doshi-Velez, F., Liu, Y., Klampfl, E., MacNeille, P.: Bayesian rule sets for interpretable classification. In: IEEE International Conference on Data Mining, pp. 1269–1274 (2016). https://doi.org/10.1109/ICDM.2016.0171
36. Wielopolski, P., Furman, O., Stefanowski, J., Zięba, M.: Probabilistically plausible counterfactual explanations with normalizing flows. In: European Conference on Artificial Intelligence (2024). https://doi.org/10.3233/FAIA240584
37. Xu, Z., Stoller, S.D.: Mining attribute-based access control policies. IEEE Trans. Dependable Secure Comput. **12**(5), 533–545 (2015). https://doi.org/10.1109/TDSC.2014.2369048
38. Yang, Y., Gao, W., Guo, Y., Zhang, Y.: HyperLogic: enhancing diversity and accuracy in rule learning with HyperNets. In: Advances in Neural Information Processing Systems 37 (2024). https://doi.org/10.52202/079017-0118