Algorithm 1: Unification Networks

```
Input: Invariant example $G$, variableness network $\psi$, example $K$, features network $\phi$, unifying features network $\phi_U$,
       upstream predictor network $f$
Output: Predicted label for example $K$

1  begin ▷ Unification Network
2      return $f \circ g(G, K)$                          ▷ Predictions using Soft Unification $g$
3  begin ▷ Soft Unification function $g$
4      foreach symbol $s$ in $G$ do
5          $\boldsymbol{A}_{s,:} \leftarrow \phi(s)$                      ▷ Features of $G$, $\boldsymbol{A} \in \mathbb{R}^{|G| \times d}$
6          $\boldsymbol{B}_{s,:} \leftarrow \phi_U(s)$                    ▷ Unifying features of $G$, $\boldsymbol{B} \in \mathbb{R}^{|G| \times d}$
7      foreach symbol $s$ in $K$ do
8          $\boldsymbol{C}_{s,:} \leftarrow \phi(s)$                      ▷ Features of $K$, $\boldsymbol{C} \in \mathbb{R}^{|K| \times d}$
9          $\boldsymbol{D}_{s,:} \leftarrow \phi_U(s)$                    ▷ Unifying features of $K$, $\boldsymbol{D} \in \mathbb{R}^{|K| \times d}$
10     Let $\boldsymbol{P} = \mathrm{softmax}(\boldsymbol{B}\boldsymbol{D}^T)$          ▷ Attention map over symbols, $\boldsymbol{P} \in \mathbb{R}^{|G| \times |K|}$
11     Let $\boldsymbol{E} = \boldsymbol{P}\boldsymbol{C}$                        ▷ Attended representations of $G$, $\boldsymbol{E} \in \mathbb{R}^{|G| \times d}$
12     foreach symbol $s$ in $G$ do
13         $\boldsymbol{U}_{s,:} \leftarrow \psi(s)\boldsymbol{E}_{s,:} + (1 - \psi(s))\boldsymbol{A}_{s,:}$   ▷ Unified representation of $G$ with $K$, $\boldsymbol{U} \in \mathbb{R}^{|G| \times d}$
14     return $\boldsymbol{U}$
```

Dear reviewers, thank you for your comments. We are pleased with the unanimous consensus on the novelty of our contribution, and recognition by some reviewers of a comprehensive evaluation. We thank Reviewer 3 for their suggestion of an algorithm box to clarify Section 2. We present Algorithm 1 above that defines our approach and the soft unification function g with all the learnable components ψ, ϕ, ϕ_U, f. We compute 2 sets of features (see L87, L90, L refers to lines in paper), then use the unifying features to let each symbol in G attend to a symbol in K. Depending on how much each symbol in G is a variable, determined by ψ, the representation of G is interpolated between its and K's features. We plan to include the algorithm in Section 2 and simplify the notation, move Figure 2 to Section 3 and add further explanation using the extra page if the paper gets accepted. We hope these changes will resolve clarity issues also raised by Reviewers 1 and 4. Reviewer 3 asks about hyper-parameters and their selection as well as the number examples used for invariances. In Appendix A, second paragraph of each model details the dimensions, layers etc. used. These were selected based on similar previous works on the datasets. We use one example to learn from but have experimented with multiple (L209) of which the predictions are aggregated (e.g. sum), Appendix D Figure 8.

Reviewers 1 and 4 point out some baselines were missing in the text. We included them (e.g. DMN, G2N2) only as references in the captions due to space constraints; we will expand their definitions to clarify further. So in response to Reviewer 4, in Table 3 DMN and IMA are baselines without soft-unification and our approach achieves comparable results to DMN with half the data size. We explain weak supervision briefly in L213 and will clarify strong supervision further. To answer Reviewer 4, we do not constrain the same symbol variable to attain the same value because it can appear in different contexts. When visualising invariants, the choice of variable symbols is given by the variabless ψ which is the same for same symbols. We leave contextualised variablisation as future work (L119). However, the value assignment can be context dependant (L123, L140).

Reviewer 2 points out there is no quantitative analysis of identifying invariants. This was not included because (i) the focus of our evaluation was to solve an upstream task with less examples using our approach whilst maintaining task performance, and (ii) our approach might solve the task without necessarily using the *expected* invariants as shown in Appendix D Figure 11 and 12. For completeness, we can add this analysis to our synthetic dataset where we know what are the expected invariances. Reviewer 2 mentions attention weights in Figure 5 are low ≤ 0.3 (also mentioned by Reviewer 4) which might indicate low confidence. There might be a misunderstanding here as these are the variableness ψ of the symbols, not the attention weights. We purposefully penalise the magnitude of ψ in equation 3 (Sparsity) so we expect them to be low. This is because we want to find the minimum variablisation of G to correctly predict K's label (see L252). In Figures 1 and 6, we threshold ψ for visualisation purposes (L227) although the interpretation of what symbols are variables is not binary, i.e. is it or is it not a variable. This is a soft view of the notion of a variable. Therefore, to answer Reviewer 2's comment on soft-interpolation, this soft view indeed produces varying symbol embeddings as intended. But this does not hurt the interpretability of the attention maps which are instead computed from the unifying features (Algorithm 1, line 10). The interpolation happens after the attention is computed.

Reviewer 2 asks about context embeddings to disamguiate unification. This is indeed used in some models through unifying features ϕ_U (L123, L140). In other words, in the sentence "***X***:lily is a ***Y***:frog" what ***X*** unifies with takes into account that it is related to a frog / animal, e.g. Figure 3 pink diamond represents unification RNN (L140). We thank Reviewer 2 for the suggested related work [a] and will cite it; however, we do not claim our approach is more interpretable than Neural Turing Machines (NTM) (L273). We discuss NTMs as an approach that lacks an explicit representation like our invariants (L278) in relation to generalisation. We acknowledge that interpretability can be subjective and biased (L281), for example we might want to think ***X***:Mary means *someone* but we do not know how the model uses these representations (L297, L300). We discuss why and why not our results could be interpretable but refrain from claiming that our model provides an explanation like Grad-CAM (L301) or is more interpretable than NTMs.

We thank you for your reviews and hope to have addressed your questions. We intend to incorporate all the suggestions and remaining remarks into the paper.