Techniques for Symbol Grounding with SATNet

Sever Topan[1, 2], David Rolnick[1, 3], Xujie Si[1, 3]

[1]McGill University, [2]NVIDIA, [3]Mila

October 2021

Introduction: Neurosymbolic Learning

$$\left.\text{Neural}\right] - \left[\text{Symbolic}\right.$$

Merge advances in statistical (neural) models with symbolic knowledge representation and logical reasoning

Introduction: Neurosymbolic Learning

$$\text{Neural}\Big] - \Big[\text{Symbolic}$$

Merge advances in statistical (neural) models with symbolic knowledge representation and logical reasoning

Potential to address limitations in DNN's:

Introduction: Neurosymbolic Learning

$$\left.\text{Neural}\right] - \left[\text{Symbolic}\right.$$

Merge advances in statistical (neural) models with symbolic knowledge representation and logical reasoning

Potential to address limitations in DNN's:

- Explainability

Introduction: Neurosymbolic Learning

$$\left.\mathrm{Neural}\right] - \left[\mathrm{Symbolic}\right.$$

Merge advances in statistical (neural) models with symbolic knowledge representation and logical reasoning

Potential to address limitations in DNN's:

- Explainability
- Adversarial Robustness

Introduction: Neurosymbolic Learning

$$\left.\text{Neural}\right] - \left[\text{Symbolic}\right.$$

Merge advances in statistical (neural) models with symbolic knowledge representation and logical reasoning

Potential to address limitations in DNN's:

- Explainability
- Adversarial Robustness
- Data Efficiency

Introduction: Neurosymbolic Learning

$$\text{Neural}\Bigg] - \Bigg[\text{Symbolic}$$

Merge advances in statistical (neural) models with symbolic knowledge representation and logical reasoning

Potential to address limitations in DNN's:

- Explainability
- Adversarial Robustness
- Data Efficiency
- Solve hard logic problems

Introduction: Symbol Grounding

$$\left.\text{Neural}\right] - \left[\text{Symbolic}\right.$$

Symbol Grounding

At the interface between a neural and a symbolic module, the meaning of the symbols must be established

Introduction: Symbol Grounding

$$\text{Neural}\Big] - \Big[\text{Symbolic}$$

Symbol Grounding

At the interface between a neural and a symbolic module, the meaning of the symbols must be established

This is known as **Symbol Grounding**

Prototypical Example: Symbol Grounding in Visual Sudoku

- MNIST digits visually represent input cells

Prototypical Example: Symbol Grounding in Visual Sudoku

- MNIST digits visually represent input cells
- Expected outputs represented numerically

Prototypical Example: Symbol Grounding in Visual Sudoku

- MNIST digits visually represent input cells
- Expected outputs represented numerically
- To solve this problem, the system must:
  - Learn the meanings of digits (neural)
  - Learn the rules of Sudoku (symbolic)

Prototypical Example: Symbol Grounding in Visual Sudoku

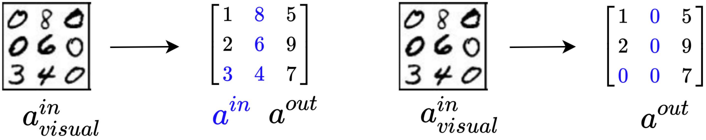
- MNIST digits visually represent input cells
- Expected outputs represented numerically
- To solve this problem, the system must:
  - Learn the meanings of digits (neural)
  - Learn the rules of Sudoku (symbolic)
- **Symbol Grounding**: understanding that the shape of the handwritten digit corresponds to one of 9 unique symbols

Prototypical Example: Symbol Grounding in Visual Sudoku

- MNIST digits visually represent input cells
- Expected outputs represented numerically
- To solve this problem, the system must:
  - Learn the meanings of digits (neural)
  - Learn the rules of Sudoku (symbolic)
- **Symbol Grounding**: understanding that the shape of the handwritten digit corresponds to one of 9 unique symbols
- Two levels of supervision for the problem:

Prototypical Example: Symbol Grounding in Visual Sudoku

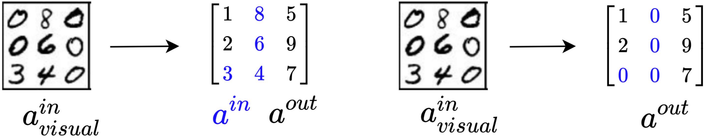
- MNIST digits visually represent input cells
- Expected outputs represented numerically
- To solve this problem, the system must:
  - Learn the meanings of digits (neural)
  - Learn the rules of Sudoku (symbolic)
- **Symbol Grounding**: understanding that the shape of the handwritten digit corresponds to one of 9 unique symbols
- Two levels of supervision for the problem:



Grounded Dataset
Trivial Symbol Grounding

Ungrounded Dataset
Difficult Symbol Grounding

Introduction: This Work

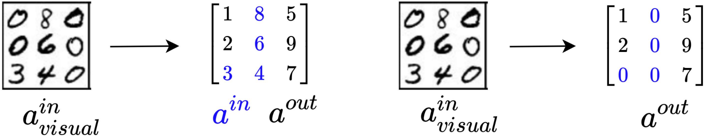


Grounded Dataset
Trivial Symbol Grounding

Ungrounded Dataset
Difficult Symbol Grounding

Previously, Ungrounded Visual Sudoku was an open problem

Introduction: This Work



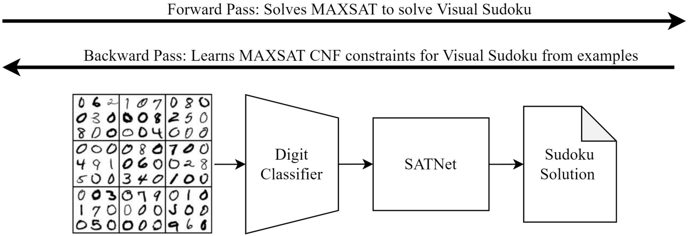
Grounded Dataset
Trivial Symbol Grounding

Ungrounded Dataset
Difficult Symbol Grounding

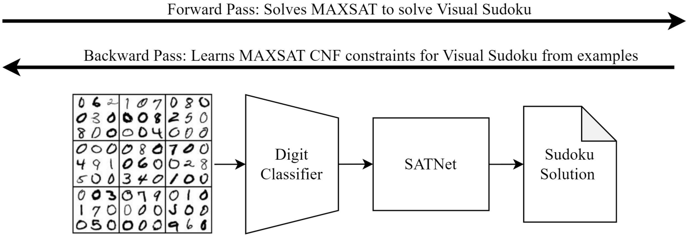
Previously, Ungrounded Visual Sudoku was an open problem

We present a framework for solving Ungrounded Visual MAXSAT problems, like Visual Sudoku, using SATNet (Wang et al. 2019)

Background: SATNet (Wang et al. 2019)

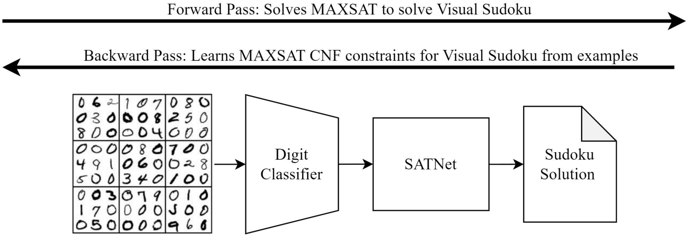


Background: SATNet (Wang et al. 2019)



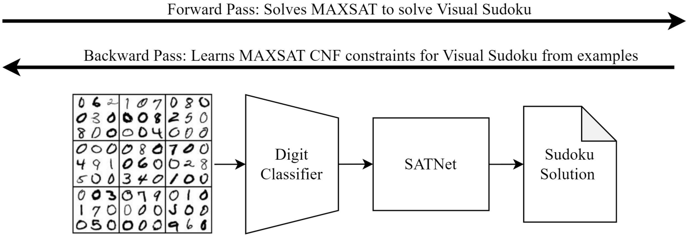
- A differentiable MAXSAT solver based on a semidefinite relaxation approach

Background: SATNet (Wang et al. 2019)



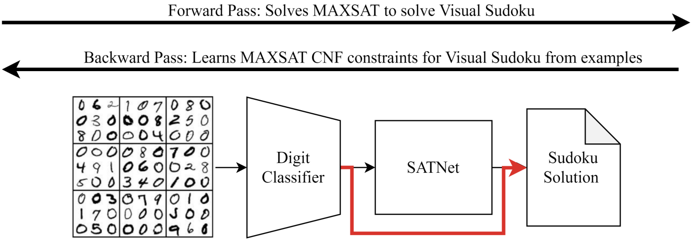
- A differentiable MAXSAT solver based on a semidefinite relaxation approach
- Can be integrated into larger DNN pipelines

Background: SATNet (Wang et al. 2019)



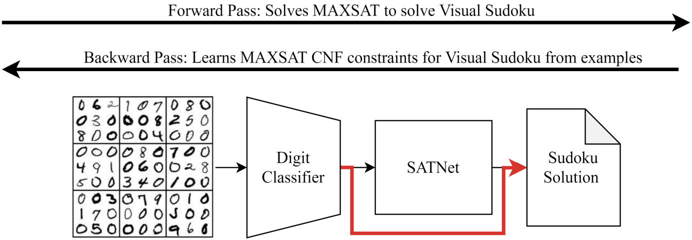
- A differentiable MAXSAT solver based on a semidefinite relaxation approach
- Can be integrated into larger DNN pipelines
- Can learn to solve grounded Visual Sudoku, while traditional DNN's cannot

Background: SATNet Limitations (Chang et al. 2020)



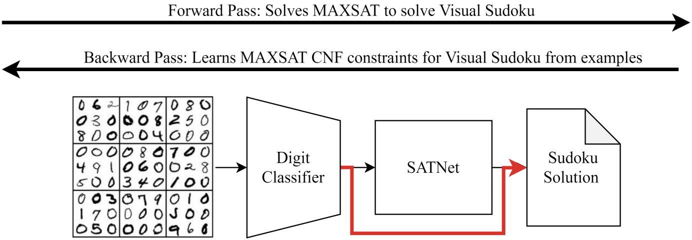
- **But**, SATNet previously could not solve Ungrounded problems, having 0% accuracy

Background: SATNet Limitations (Chang et al. 2020)



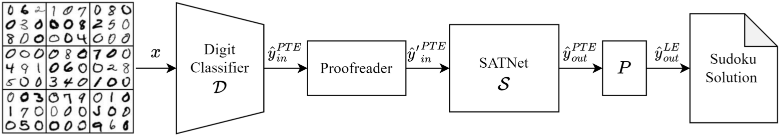
- **But**, SATNet previously could not solve Ungrounded problems, having 0% accuracy
- this issue is known as **label leakage**

Background: SATNet Limitations (Chang et al. 2020)



- **But**, SATNet previously could not solve Ungrounded problems, having 0% accuracy
- this issue is known as **label leakage**
- limits usefulness of DNN-SATNet hybrid architectures

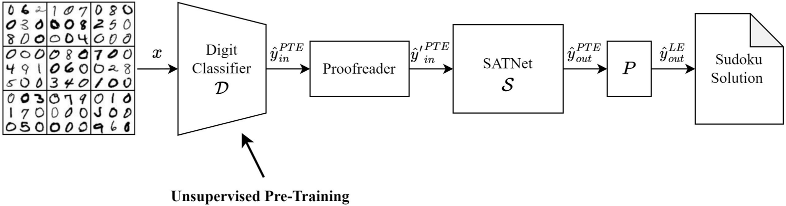
Method



Our proposed framework consists of the following steps:

1. Clustering
2. Self-Grounded Training
3. Proofreading

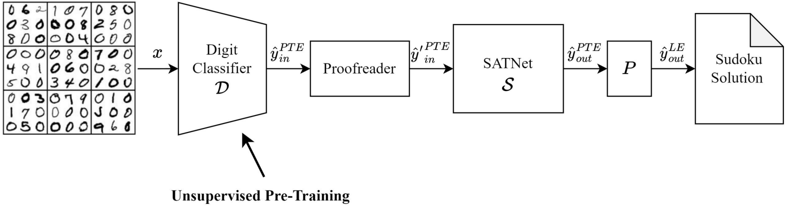
Method



Our proposed framework consists of the following steps:

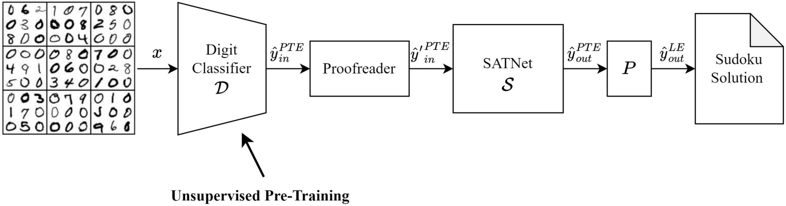
1. **Clustering**
2. Self-Grounded Training
3. Proofreading

Method: Clustering



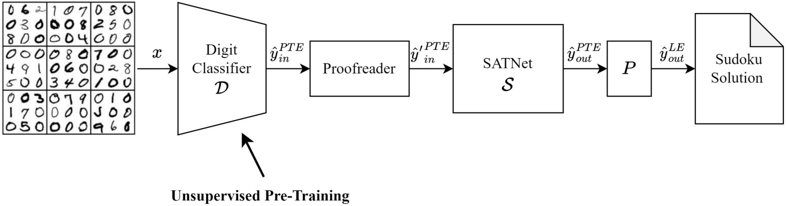
- Intuition: extract semantically relevant aspect of input images using clustering

Method: Clustering



- Intuition: extract semantically relevant aspect of input images using clustering
- Unsupervised pre-training using InfoGAN (Chen et al. 2016)

Method: Clustering



- Intuition: extract semantically relevant aspect of input images using clustering
- Unsupervised pre-training using InfoGAN (Chen et al. 2016)
- InfoGAN is able to cluster MNIST digits with about 95% accuracy

Aside: Permutation Invariance

- Inputs are clustered with 95% accuracy, but we don't know which number corresponds to which label

Aside: Permutation Invariance

- Inputs are clustered with 95% accuracy, but we don't know which number corresponds to which label
- We cannot use ground truth labels out of the box

Aside: Permutation Invariance

- Inputs are clustered with 95% accuracy, but we don't know which number corresponds to which label
- We cannot use ground truth labels out of the box
- Example: imagine we had a correct sudoku solver but assigned random labels to our clusters

Aside: Permutation Invariance

- Inputs are clustered with 95% accuracy, but we don't know which number corresponds to which label
- We cannot use ground truth labels out of the box
- Example: imagine we had a correct sudoku solver but assigned random labels to our clusters

1	8	3	9	4	5	7	6	2
4	2	5	8	3	6	1	7	9

Table: Two rows of a board predicted by a perfect sudoku model which uses InfoGAN clusters

	7	5			8	9		
		8	7					1

Table: Two rows of the corresponding Ground Truth

Aside: Permutation Invariance

1	8	3	9	4	5	7	6	2
4	2	5	8	3	6	1	7	9

Table: Two rows of a board predicted by a perfect model which uses InfoGAN.

	7	5			8	9		
		8	7					1

Table: Two rows of the corresponding Ground Truth

- Labels can be different as long as they are consistent

Aside: Permutation Invariance

1	**8**	3	9	4	5	7	6	2
4	2	5	**8**	3	6	1	7	9

Table: Two rows of a board predicted by a perfect model which uses InfoGAN.

	7	5			8	9		
		8	**7**					1

Table: Two rows of the corresponding Ground Truth

- Labels can be different as long as they are **consistent**

Aside: Permutation Invariance

1	**8**	3	9	4	5	7	6	2
4	2	5	**8**	3	6	1	7	9

Table: Two rows of a board predicted by a perfect model which uses InfoGAN.

	7	5			8	9		
		8	**7**					1

Table: Two rows of the corresponding Ground Truth

- Labels can be different as long as they are **consistent**
- This applies to other SAT-solvable games, beyond Sudoku

Aside: Permutation Invariance

1	**8**	3	9	4	5	7	6	2
4	2	5	**8**	3	6	1	7	9

Table: Two rows of a board predicted by a perfect model which uses InfoGAN.

	7	5			8	9		
		8	**7**					1

Table: Two rows of the corresponding Ground Truth

- Labels can be different as long as they are **consistent**
- This applies to other SAT-solvable games, beyond Sudoku
- Common loss functions, such as l_2 norm or binary cross-entropy (BCE), will not work

Aside: Permutation Invariance

1	**8**	3	9	4	5	7	6	2
4	2	5	**8**	3	6	1	7	9

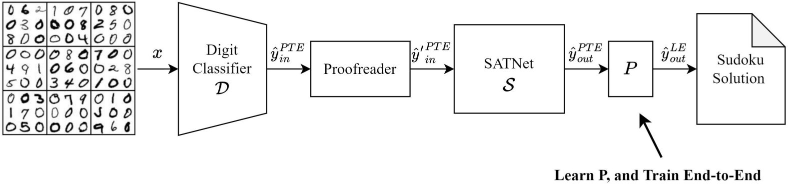
Table: Two rows of a board predicted by a perfect model which uses InfoGAN.

	7	5			8	9		
		8	**7**					1

Table: Two rows of the corresponding Ground Truth

- Labels can be different as long as they are **consistent**
- This applies to other SAT-solvable games, beyond Sudoku
- Common loss functions, such as l_2 norm or binary cross-entropy (BCE), will not work
- Need a different loss function

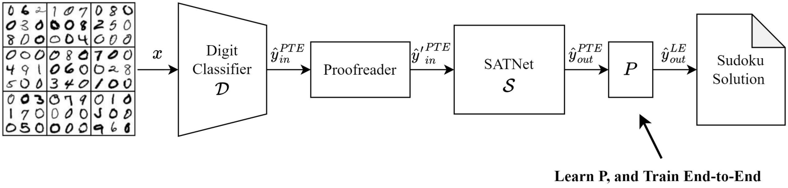
Method



Our proposed framework consists of the following steps:

1. Clustering
2. **Self-Grounded Training**
3. Proofreading

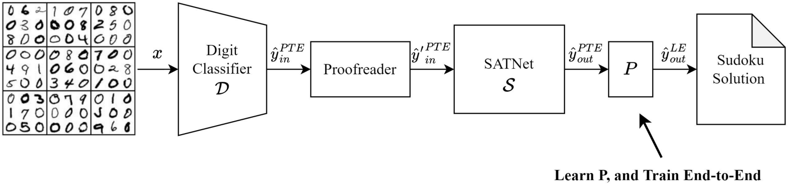
Method: Self-Grounded Training



Introduce the Symbol Grounding Loss (SGL):

$$\mathcal{L}(\hat{y}_{out}^{PTE}, y^{LE}) := 1 -_i (\max_j(\exp[-(y^{LE}(j), \hat{y}_{out}^{PTE}(i))])),$$

Method: Self-Grounded Training

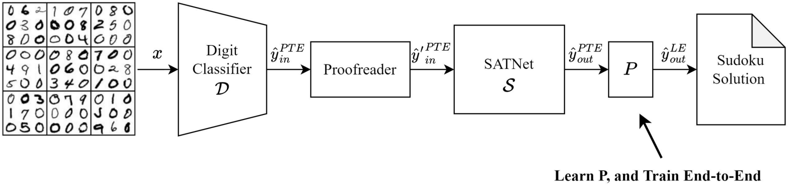


Introduce the Symbol Grounding Loss (SGL):

$$\mathcal{L}(\hat{y}_{out}^{PTE}, y^{LE}) := 1 -_i (\max_j(\exp[-(y^{LE}(j), \hat{y}_{out}^{PTE}(i))])),$$

- SGL infers a permutation matrix between predictions and labels (See paper for further details)

Method: Self-Grounded Training

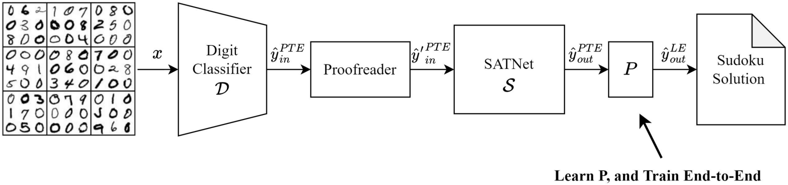


Introduce the Symbol Grounding Loss (SGL):

$$\mathcal{L}(\hat{y}_{out}^{PTE}, y^{LE}) := 1 -_i (\max_j(\exp[-(y^{LE}(j), \hat{y}_{out}^{PTE}(i))])),$$

- SGL infers a permutation matrix between predictions and labels (See paper for further details)
- System is trained end-to-end under the Symbol Grounding Loss

Method: Self-Grounded Training

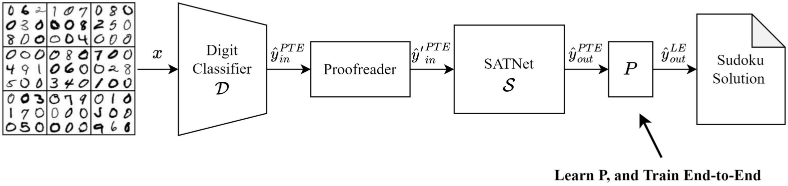


Introduce the Symbol Grounding Loss (SGL):

$$\mathcal{L}(\hat{y}^{PTE}_{out}, y^{LE}) := 1 -_i (\max_j(\exp[-(y^{LE}(j), \hat{y}^{PTE}_{out}(i))])),$$

- SGL infers a permutation matrix between predictions and labels (See paper for further details)
- System is trained end-to-end under the Symbol Grounding Loss
- A permutation matrix P is implicitly learned by SGL

Method: Self-Grounded Training

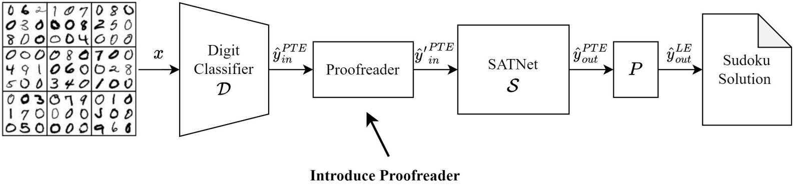


Introduce the Symbol Grounding Loss (SGL):

$$\mathcal{L}(\hat{y}^{PTE}_{out}, y^{LE}) := 1 -_i (\max_j(\exp[-(y^{LE}(j), \hat{y}^{PTE}_{out}(i))])),$$

- SGL infers a permutation matrix between predictions and labels (See paper for further details)
- System is trained end-to-end under the Symbol Grounding Loss
- A permutation matrix P is implicitly learned by SGL
- Once P has converged, continue training under standard BCE

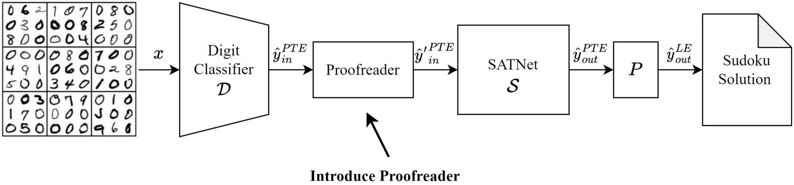
Method



Our proposed framework consists of the following steps:

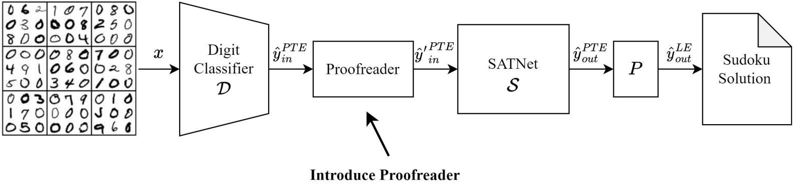
1. Clustering
2. Self-Grounded Training
3. **Proofreading**

Method: Proofreading



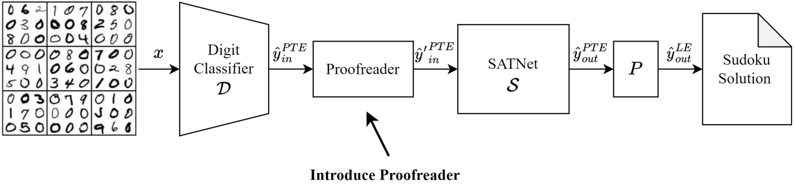
- Insert a linear layer before SATNet

Method: Proofreading



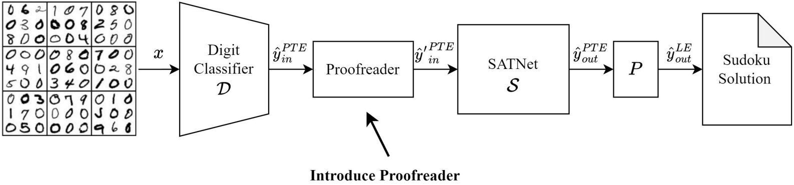
- Insert a linear layer before SATNet
- Initialize to a slightly noisy identity transform

Method: Proofreading



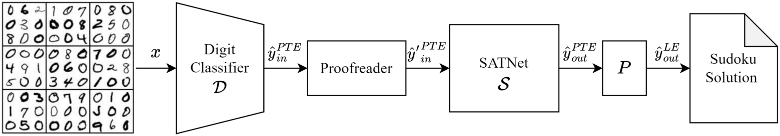
- Insert a linear layer before SATNet
- Initialize to a slightly noisy identity transform
- Freeze rest of system, train proofreader

Method: Proofreading



- Insert a linear layer before SATNet
- Initialize to a slightly noisy identity transform
- Freeze rest of system, train proofreader
- Improves accuracy marginally in both our method and prior SATNet architectures

Method



Our proposed framework consists of the following steps:

1. Clustering
2. Self-Grounded Training
3. Proofreading

Results: Ungrounded Visual Sudoku

Model Configuration	**Grounded vs. Ungrounded Data**	**Total Board Accuracy** (%)	**Per-Cell Accuracy** (%)
Original SATNet	grounded	66.5 ± 1.0	98.8 ± 0.1
Original SATNet	ungrounded	0 ± 0.0	11.2 ± 0.1
Our Method	**ungrounded**	$\mathbf{64.8 \pm 3.0}$	$\mathbf{98.4 \pm 0.2}$

Results: Effect of Proofreader

Model Configuration	**Proofreader Present?**	**Total Board Accuracy (%)**
Original Non-visual	no	96.6 ± 0.3
Original Non-visual	**yes**	$\mathbf{97.1} \pm \mathbf{0.3}$
Original Visual	no	66.5 ± 1.0
Original Visual	**yes**	$\mathbf{67.6} \pm \mathbf{1.2}$
Our Method	no	62.8 ± 3.2
Our Method	**yes**	$\mathbf{64.8} \pm \mathbf{3.0}$

Limitations & Future Work

- In an ablation test, we find that our system requires roughly at least 88% clustering accuracy in order for the rest of the pipeline to progress

Limitations & Future Work

- In an ablation test, we find that our system requires roughly at least 88% clustering accuracy in order for the rest of the pipeline to progress
- Our approach requires prior knowledge of the *number* of symbols

Limitations & Future Work

- In an ablation test, we find that our system requires roughly at least 88% clustering accuracy in order for the rest of the pipeline to progress
- Our approach requires prior knowledge of the *number* of symbols
- Above can be alleviated but Symbol Grounding Loss supporting a general surjective mapping instead of permutation

Conclusion

In this work we:

Conclusion

In this work we:

- Distinguish between grounded and ungrounded variants of Visual MAXSAT problems

Conclusion

In this work we:

- Distinguish between grounded and ungrounded variants of Visual MAXSAT problems
- Present a framework which enables SATNet to solve ungrounded datasets

Conclusion

In this work we:

- Distinguish between grounded and ungrounded variants of Visual MAXSAT problems
- Present a framework which enables SATNet to solve ungrounded datasets
- New state-of-the-art for Ungrounded Visual Sudoku, previously 0%

Conclusion

In this work we:

- Distinguish between grounded and ungrounded variants of Visual MAXSAT problems
- Present a framework which enables SATNet to solve ungrounded datasets
- New state-of-the-art for Ungrounded Visual Sudoku, previously 0%
- Describe a proofreading methodology which improves both our architecture and prior models

Conclusion

In this work we:

- Distinguish between grounded and ungrounded variants of Visual MAXSAT problems
- Present a framework which enables SATNet to solve ungrounded datasets
- New state-of-the-art for Ungrounded Visual Sudoku, previously 0%
- Describe a proofreading methodology which improves both our architecture and prior models
- Available: `github.com/SeverTopan/SATNet`

Techniques for Symbol Grounding with SATNet

Sever Topan[1, 2], David Rolnick[1, 3], Xujie Si[1, 3]

[1]McGill University, [2]NVIDIA, [3]Mila

October 2021