Centre for Research Training

Science Foundation Ireland For what's next

Constraint Acquisition By Transformer

Samaksh Chandra, Steven Prestwich, Gregory Provan

University College Cork, Ireland

CPML 2024

HOST INSTITUTION

UCC University College Cork, Ireland Coláiste na hOllscoile Corcaigh

PARTNER INSTITUTIONS

University of Limerick Ollscoil Luimnigh

Ollscoil na Gaillimhe University of Galway

DCU

Trinity College Dublin Coláiste na Tríonóide, Baile Átha Cliath The University of Dublin

CRT·AI SFI Centre for Research Training in Artificial Intelligence

Overview

- Constraint acquisition (CA) is an active research area in Constraint Programming (CP).
- CA aims to automate CP by learning constraints for combinatorial problems from examples of solutions and (often) non-solutions. (Some methods are ACTIVE and involve user interaction, but we are interested in PASSIVE methods that do not.)
- Methods have been based on a range of approaches from machine learning and optimization, but so far generative AI has played only a small role.
- We present an attempt at such an approach for SAT, called CNFformer, using a transformer to learn to generate propositional satisfiability models.

Constraint Programming

- CP provides powerful modeling languages and solvers for constrained optimization and decision-making problems and has many applications.

- A constraint satisfaction problem (CSP) consists of a set of variables, each with a domain of possible values, and a network of constraints each defined on a subset of the variables. Modelling a new application as a CSP requires expertise, which can impede the uptake of CP. This has motivated the development of CA.

- Learning a CSP is NP-Hard and CA is known to be a challenging task.

Constraint Acquisition

- CA has been developed for over 20 years and a variety of approaches have been explored.

- From a set of possible constraints (candidates) called the bias, we must select which are to be learned, based on training data containing examples of solutions and (usually) non-solutions.

- CA has been identified as an important topic and as progress toward the Holy Grail of computer science: the user presents a problem to the computer in a natural manner, and the computer solves it.

Constraint Acquisition

- Existing CA methods are based on a wide variety of techniques from ML and other areas: inductive logic programming, version space learning, classifiers, sequential analysis, data mining, posing the CA problem as an optimization or CP problem, and approaches based on grammars or tensors.

- A few CA methods can generalize to different problem sizes, for example Model Seeker.

CA and Deep Learning

- This is an area that has received little attention, with most focus on end-to-end learning (learning how to solve constraint problems).
- Transformers are a form of artificial neural network that have performed impressively on many tasks previously the preserve of humans, and given rise to the field of Generative AI, so it is natural to wonder if they can also perform CA.
- If so, they might have abilities beyond those of current approaches...

Why Transformers?

- Transformer training is expensive, but if they can be trained to generate a constraint model from a dataset of labelled instances, they will be able to do so very quickly: they effectively learn how to handle an entire class of problem. Long execution times have plagued CA methods so this would be a valuable property.

- They might be able to handle very large biases. Most methods have only been tested on biases containing a few tens of thousands of candidates. Recent work has handled sizes of up to a billion candidates, but a bias containing (say) all possible all-different global constraints on a set of 100 variables is astronomically larger. Such biases are currently handled by making simplifying regularity assumptions, which transformers might learn automatically.

Why Transformers?

- Transformers can handle inputs of different size (eg images with different numbers of pixels) so they might be able to learn constraints for one problem size after being trained on examples of another size. Very few CA methods attempt to do this, and transformers would provide a new approach.

- Although transformers do not always generalize well, a great deal of current research aims to improve their ability and CA might benefit from progress in this area. Eg if a transformer could be trained to learn both planning and scheduling constraint models, it might be able to learn constraints for planning-scheduling hybrid problems.

Why Transformers?

- Transformers succeed at related tasks such as symbolic regression: trained on a large dataset of regression examples, they can outperform standard methods based on genetic programming.

- Transformers form the basis of large language models (LLMs) which show signs of generalization ability beyond what was expected (Geoff Hinton and others say they are more than “stochastic parrots”) so they might have unexpected CA abilities. LLMs have recently been used to generate constraint models from natural language descriptions.

- These observations are speculative, and it is not obvious whether transformers can achieve all this, but understanding the proficiency of LLMs in logical reasoning is a hot research area.

Our Contribution

- We devise the first methodology for training transformer models to autonomously learn SAT clauses from examples, which we call CNFformer.

Truth Assignments (solutions + non-solutions) Sampled from Random 3-SAT CNF → Transformer Model → Generated 3-SAT CNF

- We form a new connection between the CP and ML communities.

- We present experimental results [TO BE ANNOUNCED]

Transformer Operations in SAT Learning

- T_θ is a transformer which interprets truth assignments as "input language" and performs stochastic formula generation to transform them into logical formulae, the "output language".

- It acts as a many-to-many formula generator.

- By generating the most probable formula token for truth assignment tokens, T_θ optimizes the formula generation process.

- T_θ represents a significant advancement, utilizing self attention mechanisms to process sequential data more efficiently than RNNs and LSTMs.

Transformer Operations in SAT Learning

- We define a finite set V as vocabulary for truth assignments (V_B) and CNF formulae ($V_\emptyset$). We consider only well-formed formulae generated from $V_\emptyset$ which are formulae in proper k-CNF syntax.

- Training Task: Given a vocabulary pair (V_B, $V_\emptyset$) and an i.i.d. dataset of sequence pairs {z, w}, learn a conditional probability distribution to estimate P(w|z) where z is a sequence of truth assignments $\tilde{X}$, w is the formula ø for which the set of truth assignments $\tilde{X}$ makes ø true.

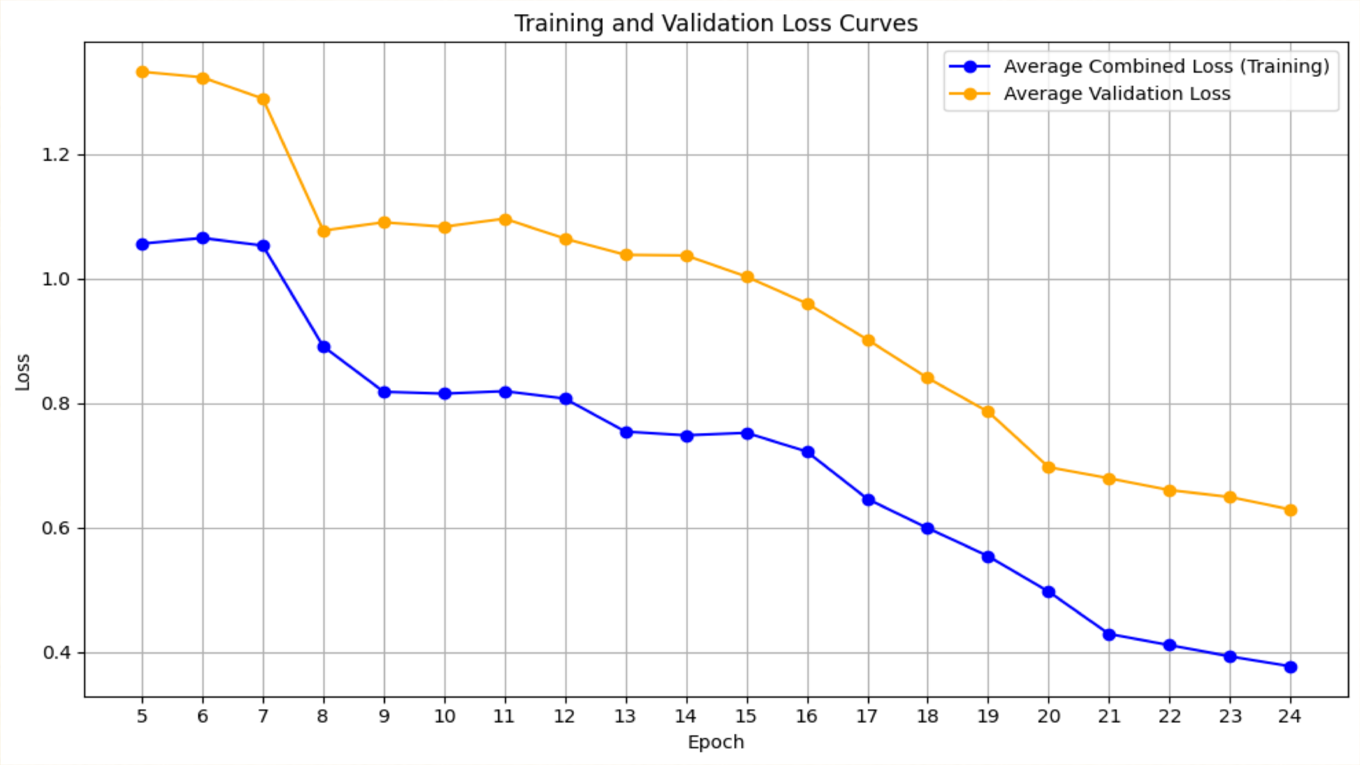
- Acquisition Task: Given a set of input truth assignments z, return a SAT formula w. T_θ projects the combination of X (truth assignment sub-string) and C (Context) into Y (resulting SAT formula) with a certain probability π which ranges between 0 and 1.

Transformer Operations in SAT Learning

- For an input sequence X = $\{x_1, x_2, \ldots, x_n\}$ the self-attention mechanism computes matrices Q, K, and V which are learned parameter matrices through which the attention scores are calculated between elements. This computation reflects how much each element of the sequence should attend to every other element, enabling the model to capture long-range dependencies.

- In order to compensate for the lack of sequential order awareness in parallel processing, positional encodings are added to the input embeddings. These encodings provide information about the order of tokens in the sequence. Each layer of the transformer T_θ applies self-attention, followed by a position-wise feed forward network.

Transformer Operations in SAT Learning



Total Loss Function = Cross Entropy Loss + Custom Equisatisfiable Loss

Cross Entropy Loss =

$$-\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij}\log p_{ij}$$

N -> Number of examples in dataset
C -> No. of classes in LED model
y_{ij}-> Binary Indicator if class j is correct classification of example i
p_{ij}->predicted prob of class i belongs to class j

Custom Equisatisfiable Loss =

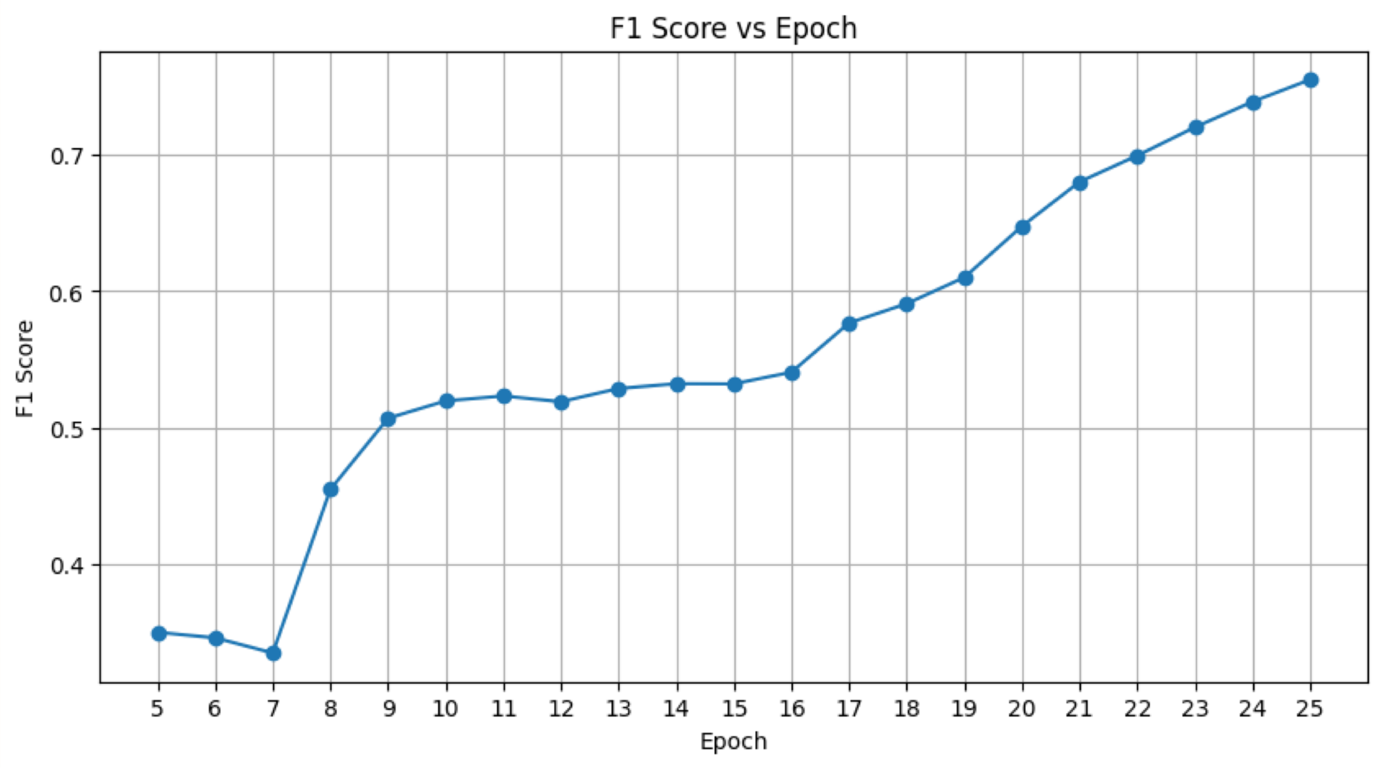
2 – Precision + (α×unsatisfied clause ratio in target model) – recall + (β×unsatisfied clause ratio in learned model)

Results

- To be announced at the event! We aim to test the transformer's ability to learn out-of-distribution, eg other 3-SAT formulae and generalization to different numbers of variables. We are also investigating the use of fine-tuning and transfer learning.

- However, we can present a key result: a transformer can learn to generate small random 3-SAT models from below the phase transition (10 variables 20 clauses) from a set of truth assignments labelled as solutions/non-solutions. Training is lengthy but accuracy increases steadily, currently reaching an F1 score of 0.75

Results


F1 Score vs Epoch
F1 Score
0.4
0.5
0.6
0.7
Epoch
5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25

Results

- We evaluate the learned formulae against the target formulae from a known strategy discussed in the “COUNT-CP” paper.

- Performance is measured in terms of Precision and Recall.

- Precision defines what percentage of the learned feasible region is actually feasible in the target model. This is done by randomly sampling solutions from the learned model and checking them in the target model.

- Recall defines what percentage of the target feasible region is captured by the learned model. This is done by randomly sampling solutions from the target model and checking them in the learned model

Results

Precision	Recall	F1-Score
0.62	0.96	0.75

The above is calculated on test data of 100 target formulae

Centre for Research Training
Science Foundation Ireland For what's next
HOST INSTITUTION
University College Cork, Ireland
Coláiste na hOllscoile Corcaigh
PARTNER INSTITUTIONS
University of Limerick
Ollscoil Luimnigh
Ollscoil na Gaillimhe
University of Galway
DCU
Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin
CRT·AI
SFI Centre for Research Training in Artificial Intelligence

Conclusion

- We believe that transformers will make a great contribution to constraint acquisition.

- Training is currently very expensive, but the field will benefit from future progress on generative AI.

Acknowledgements

This work was conducted with the financial support of the Science Foundation Ireland Centre for Research Training in Artificial Intelligence under Grant No. 18/CRT/6223.