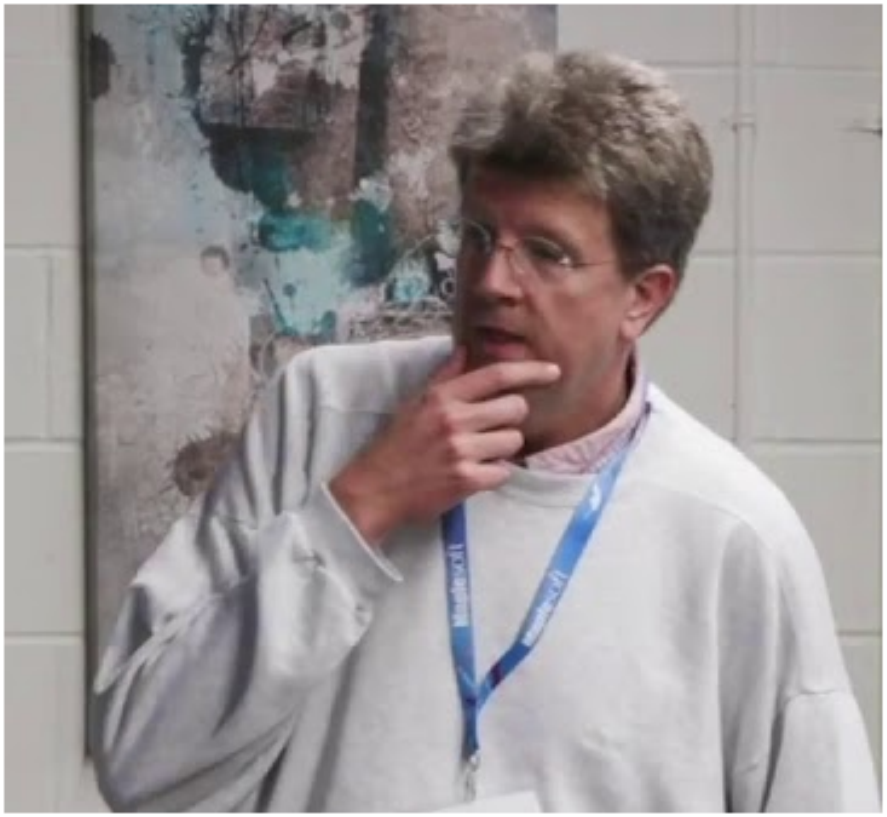
Machine Learning Symbolic Integration Algorithm Selection

Prof. Matthew England
Coventry University

International Conference on Symbolic Computation and Machine Learning (SCML 2026)
6–8 July 2026
Hagenberg, Austria

Acknowledgements

Collaboration with Rashid Barket and Jürgen Gerhard. Sponsored by Maplesoft.



This is a Survey


R. Barket, M. England, and J. Gerhard.
Generating Elementary Integrable Expressions
Proc. CASC '23, LNCS 14139, pp. 21 – 38, 2023.
https://doi.org/10.1007/978-3-031-41724-5_2


R. Barket, M. England, and J. Gerhard.
The Liouville Generator for Producing Integrable Expressions
Proc. CASC '24, LNCS 14938, pp. 47 – 62, 2024.
https://doi.org/10.1007/978-3-031-69070-9_4


R. Barket, U. Shafiq, M. England, and J. Gerhard.
Transformers to Predict the Applicability of Symbolic Integration Routines
Proc. MATH-AI at NeurIPS '24, 2024.
https://openreview.net/forum?id=b2Ni828As7


R. Barket, M. England, and J. Gerhard.
Symbolic Integration Algorithm Selection with ML: LSTMs vs Tree LSTMs.
Proc. ICMS '24, LNCS 14749, pp. 167–175, 2024.
https://doi.org/10.1007/978-3-031-64529-7_18


R. Barket, M. England, and J. Gerhard.
Tree-Based Deep Learning for Ranking Symbolic Integration Algorithms.
Submitted, 2025. Preprint: arXiv:2508.06383

Key Lessons Learnt

1. **Importance of Data Quality:** existing data / data generation methods contained biases and shortcomings; new computer algebra research to generate good data.
2. **Evaluate generalisation carefully:** With mathematics data, evaluation of generalisation is more difficult than using a ring-fenced test-set; you need truly independent datasets.
3. **Embedding choice:** Our initial work used sequential embeddings; but performance was much stronger with tree embeddings.
4. **Algorithm selection benefited from a two-stage architecture:** classification on algorithm success, followed by ranking on output quality.

Outline

Indefinite Integration in Maple

Symbolic indefinite integration is a key feature of CASs, including Maple where it is accessed via the command `int`: a meta-algorithm which does pre-processing and table lookup before accessing particular CAS algorithms for integration, of which there are at least 12 to choose from. Examples:

$$> f := 7x^3 + 3x^2 + 5x :$$

$$> \mathrm{int}(f, x)$$

$$\frac{7}{4}x^4 + x^3 + \frac{5}{2}x^2$$

$$> \mathrm{int}(\sin(x), x)$$

$$-\cos(x)$$

$$> \mathrm{int}\left(\frac{x}{x^3 - 1}, x\right)$$

$$-\frac{\ln\left(x^2 + x + 1\right)}{6} + \frac{\sqrt{3}\arctan\left(\frac{(2x+1)\sqrt{3}}{3}\right)}{3} + \frac{\ln(x-1)}{3}$$

Integration of Special Functions

Maple has 100s of special functions implemented, many of which can appear in integrals.

$$> \text{int}\left(\exp\left(-x^2\right), x\right)$$

$$\frac{\sqrt{\pi}\,\text{erf}(x)}{2}$$

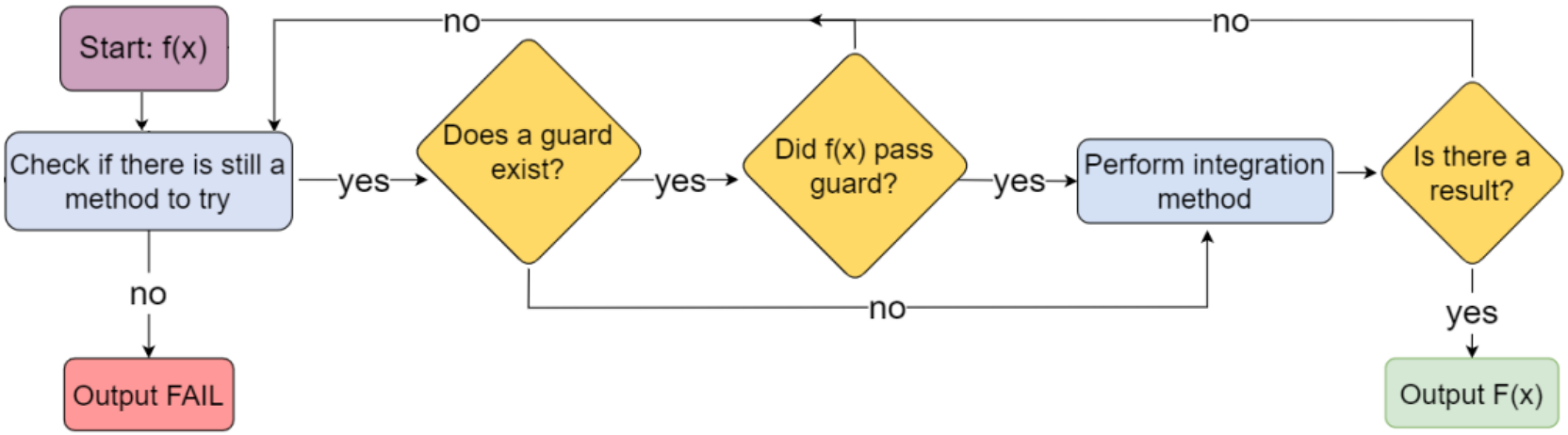
$$> \text{int}\left(\frac{1}{\text{sqrt}\left(2t^4 - 3t^2 - 2\right)}, t = 2..3\right)$$

$$\frac{\sqrt{5}\,\text{EllipticF}\left(\frac{\sqrt{7}}{3}, \frac{\sqrt{5}}{5}\right)}{5} - \frac{\sqrt{5}\,\text{EllipticF}\left(\frac{\sqrt{2}}{2}, \frac{\sqrt{5}}{5}\right)}{5}$$

This adds a layer of complexity which makes the direct ML solution more difficult.

How `int` Currently Works

The current default approach for `int` is to try all 12 sub-algorithms in a fixed order for an input, until one of them succeeds. In some cases "*guard code*" allows for a method to be skipped early.

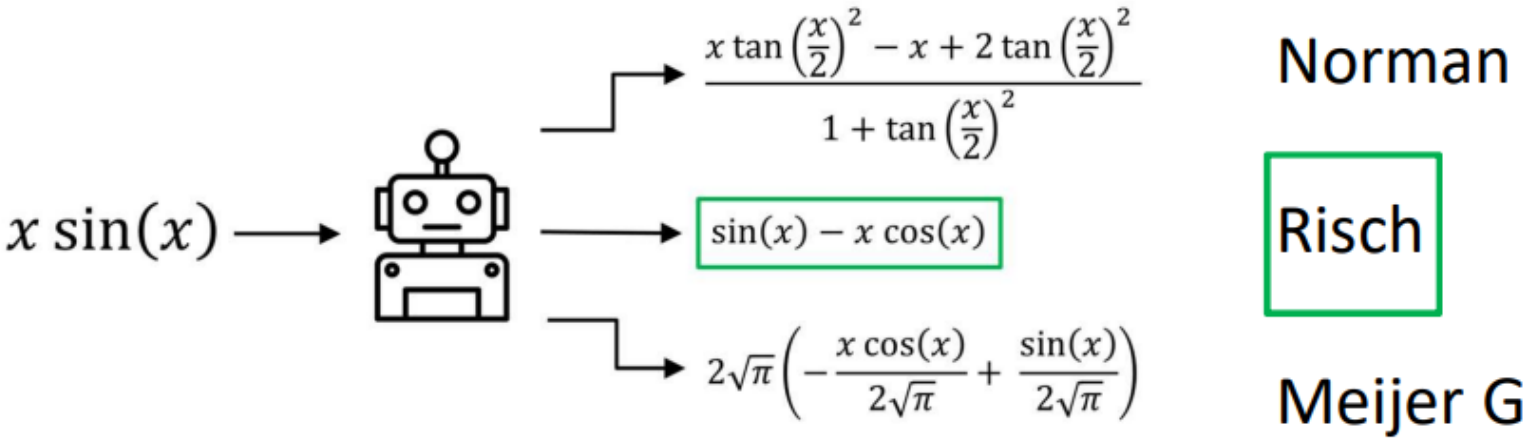


A high-level overview of how the indefinite integral command works in Maple to calculate $F(x) = \int f(x)dx$

Hypothesis: ML can make this more efficient by tailoring the `int` meta-algorithm flow to a particular problem instance.

Prioritise Output Representation

There is clearly scope for computational savings from avoiding executing sub-algorithms that fail before identifying one that works. We could save even more time by trying to identify the fastest sub-algorithms. But we actually prioritise using algorithms that give "*nice looking*" answers. E.g.:



Our lessons learned are not dependent on this metric choice.

Example from the Dataset

The difference in answer sizes can be significant!

```
> f := 1 - x*cos(cos(x))*sin(x) + sin(cos(x))
```

$$f := 1 - x\cos(\cos(x))\sin(x) + \sin(\cos(x))$$

```
> # Maple's answer
  int(f, x)
```

$$x + x\tan\left(\frac{x}{2}\right)^2 + x\tan\left(\frac{1-\tan\left(\frac{x}{2}\right)^2}{2\left(1+\tan\left(\frac{x}{2}\right)^2\right)}\right)^2 + x\tan\left(\frac{x}{2}\right)^2\tan\left(\frac{1-\tan\left(\frac{x}{2}\right)^2}{2\left(1+\tan\left(\frac{x}{2}\right)^2\right)}\right)^2 + 2x\tan\left(\frac{1-\tan\left(\frac{x}{2}\right)^2}{2\left(1+\tan\left(\frac{x}{2}\right)^2\right)}\right) + 2x\tan\left(\frac{x}{2}\right)^2\tan\left(\frac{1-\tan\left(\frac{x}{2}\right)^2}{2\left(1+\tan\left(\frac{x}{2}\right)^2\right)}\right)$$
$$\left(1+\tan\left(\frac{1-\tan\left(\frac{x}{2}\right)^2}{2\left(1+\tan\left(\frac{x}{2}\right)^2\right)}\right)^2\right)\left(1+\tan\left(\frac{x}{2}\right)^2\right)$$

```
> # ML-suggested answer

  int(f, x, method=risch)
```

$$x + \sin(\cos(x))\,x$$

We also have much bigger examples that wouldn't fit on a slide.

Outline

What Data Did We Start With?

Maplesoft have an internal test suite developed over the years: $\sim 47k$ examples. Significant, but not sufficient for deep learning. We use this as the validation dataset and seek to use an alternative – synthetic – dataset for testing.

We started with the same synthetic data as Lample and Charton:

G. Lample and D. Charton
Deep Learning for Symbolic Mathematics.
Proc. ICLR 2020.
https://doi.org/10.48550/arXiv.1912.01412

Lample & Carton (2020) Data Generation (Slide 10/24)

Lample and Charton identified three approaches to generating synthetic (integrand, integral) pairs:

FWD: Integrate a random expression f using a CAS to get F and add (f, F) to dataset.

BWD: Differentiate a random expression f with a CAS to get f' and add (f', f) to the dataset.

IBP: Differentiate two random expressions f, g to get f', g'. Then if $\int f'g$ is "*known*" we may calculate

$$\int fg' := fg - \int f'g$$

and add the pair $(fg', fg - \int f'g)$ to the dataset.

Problems with Lample & Carton (2020) Data (Slide 11/24)

1. Potential Biases: The FWD dataset has integrands much shorter than integrals; the BWD dataset the opposite. Training on one dataset and testing on the other shows *very* poor accuracy. No independent validation dataset to test on.

2. Potential Data Leakage: There are lots of *very* similar expressions. Substituting all coefficients with a symbolic CONST token reduced the datasets to 35%, 75% and 24% of their sizes.

E. Davis.

The Use of Deep Learning for Symbolic Integration: A Review of [LC20]

https://arxiv.org/abs/1912.05752

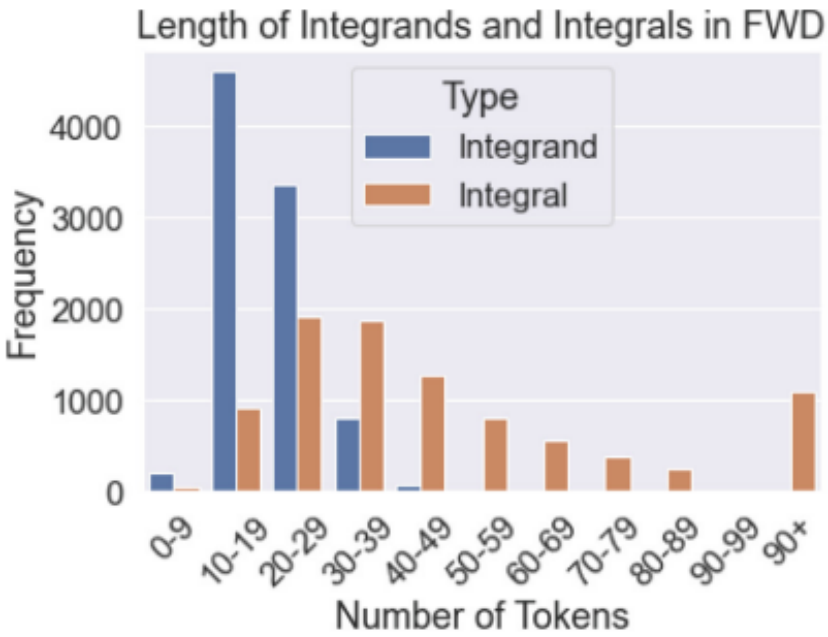
B. Piotrowski, C. Brown, J. Urban and C. Kaliszyk.

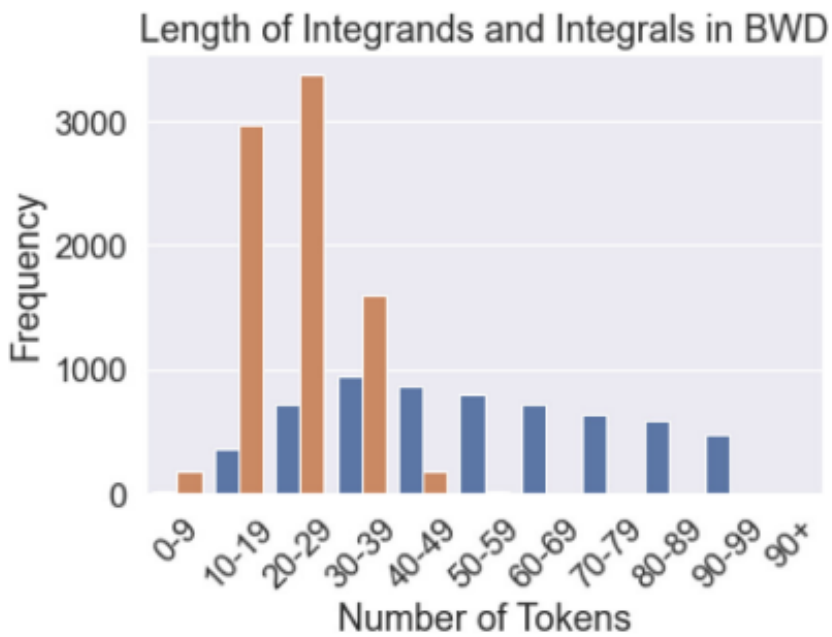
Can Neural Networks Learn Symbolic Rewriting?.

In: Proc. AITP '19, 2019.

https://graphreason.github.io/papers/40.pdf

From Barket et al. (2023)





Problems with Lample & Carton (2020) Data (Slide 11/24)

1. Potential Biases: The FWD dataset has integrands much shorter than integrals; the BWD dataset the opposite. Training one one dataset and testing on the other shows *very* poor accuracy. No independent validation dataset to test on.

2. Potential Data Leakage: There are lots of *very* similar expressions. Substituting all coefficients with a symbolic CONST token reduced the datasets to 35%, 75% and 24% of their sizes.

E. Davis.

The Use of Deep Learning for Symbolic Integration: A Review of [LC20]

https://arxiv.org/abs/1912.05752

B. Piotrowski, C. Brown, J. Urban and C. Kaliszyk.

Can Neural Networks Learn Symbolic Rewriting?.

In: Proc. AITP '19, 2019.

https://graphreason.github.io/papers/40.pdf

New Data Generation Methods

We reverse engineered classical computer algebra algorithms to form data generators.

R. Barket, M. England, and J. Gerhard. *Generating Elementary Integrable Expressions*

Proc. CASC '23, LNCS 14139, pp. 21 – 38, 2023.
https://doi.org/10.1007/978-3-031-41724-5_2

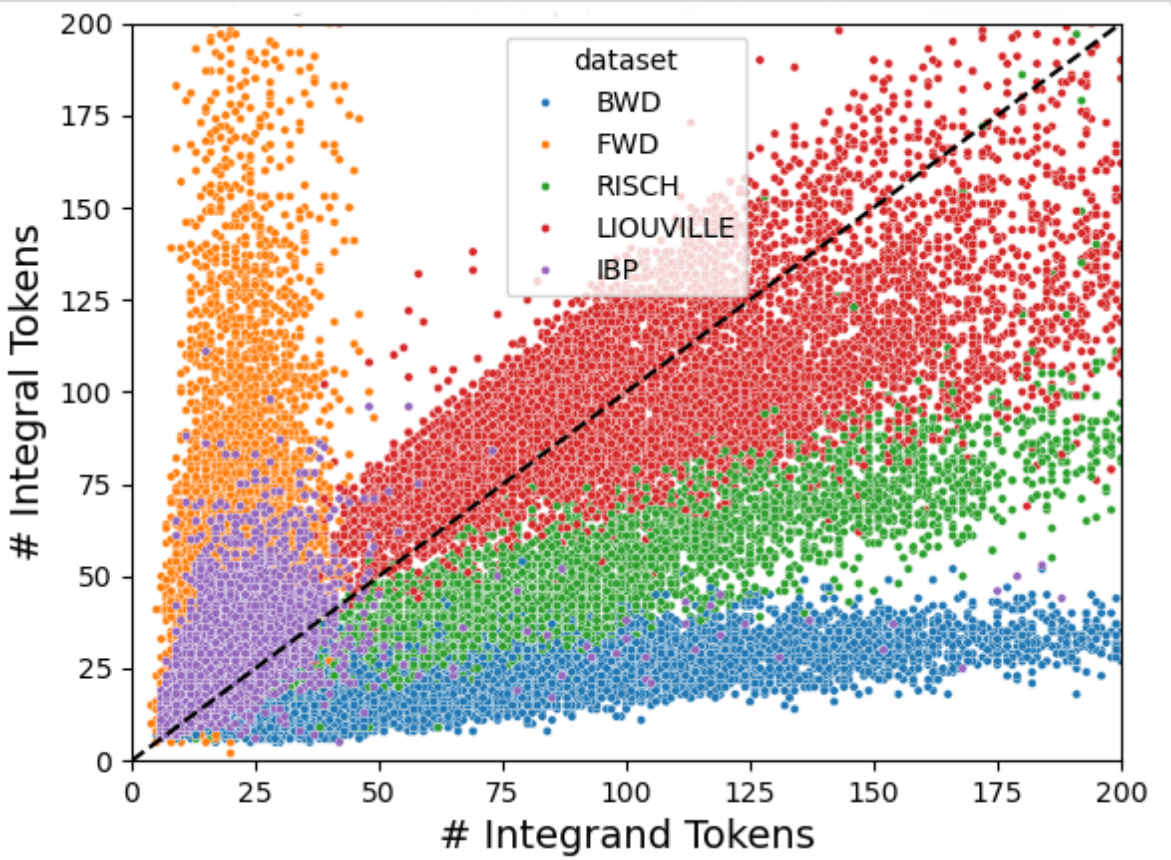
First the Risch algorithm: finding the root of the Trager-Rothstein resultant, which gave a computational barrier.

R. Barket, M. England, and J. Gerhard. *The Liouville Generator for Producing Integrable Expressions*

Proc. CASC '24, LNCS 14938, pp. 47 – 62, 2024.
https://doi.org/10.1007/978-3-031-69070-9_4

Then later the Parallel-Risch, or Risch-Normal algorithm which hypothesises a solution structure and can handle higher degree denominators and special functions more easily.

New Data Generation Methods Remove Biases (Slide 13/24)



Lesson Learnt

Importance of Data Quality: new computer algebra research is needed to generate data free of bias for ML.

Outline

Constant Embedding Choice

We use a large dataset drawn from all of the data generation methods. Although we hypothesise that the latter generator could actually be sufficient alone.

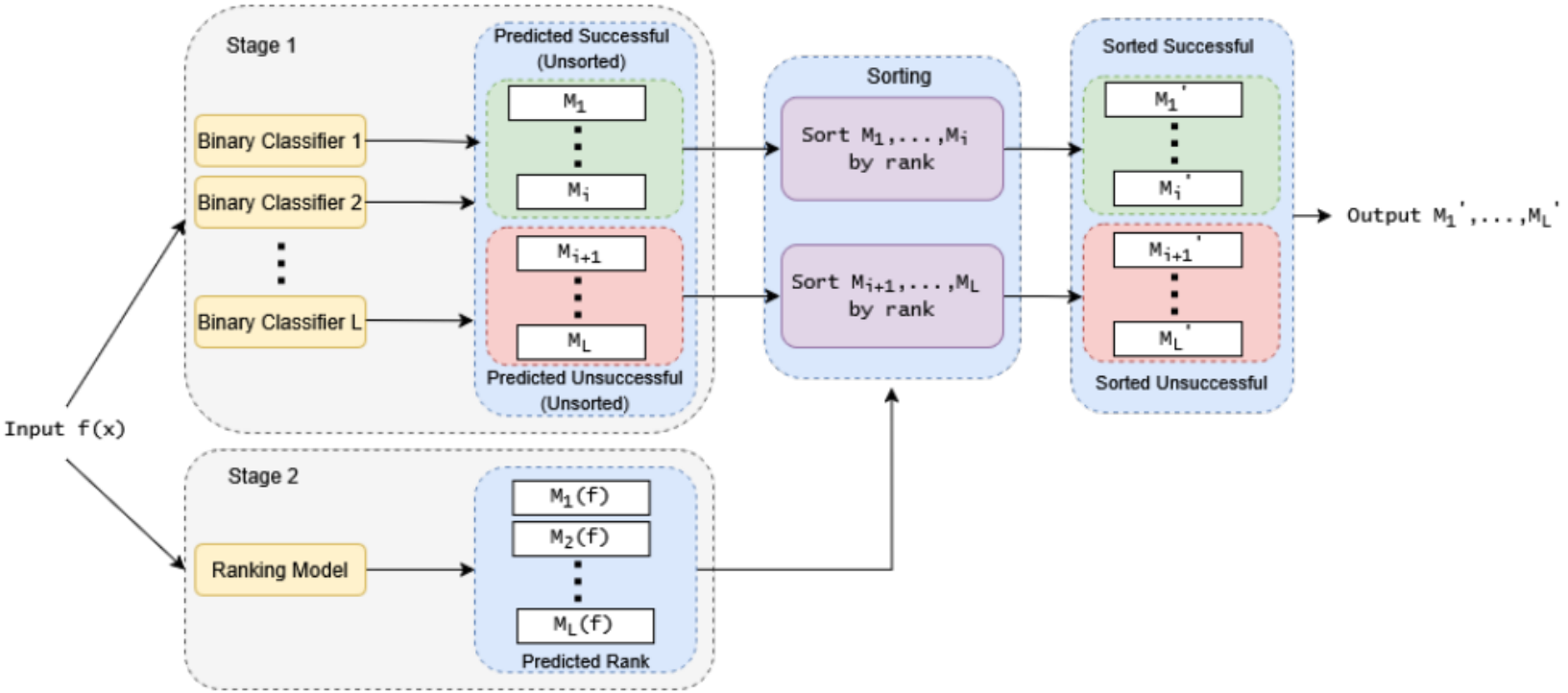
The examples contain constants up to three digits long. However, for ML training we:

- tokenize to extract minus signs separately;
- replace constants > 2 by symbolic tokens encoding only their number of digits: CONST1, CONST2, CONST3.
- Any duplicate examples after this replacement are discarded.

E.g. $2x^2 - 15x + 1$ becomes $2x^2 - \text{CONST2}x + 1$ for ML training. This removes the data leakage fears from earlier.

Selecting from Admissible Methods

We have settled on a two-stage architecture: guard (multi-label classification) followed by ranking of admissible methods using the RankNet loss function (Burges et al., 2002).



Guarding Integration Methods

Task: train a binary classifier to predict whether an integration method will succeed on a particular input.

Aim: To potentially replace guard code for methods that have them; provide guards for those that do not.

Method: Transformers reading integrands as strings (in Polish notation to reduce length), similar to Lample & Charton (2020).

R. Barket, U. Shafiq, M. England, and J. Gerhard.
Transformers to Predict the Applicability of Symbolic Integration Routines
Proc. MATH-AI at NeurIPS '24, 2024.
https://openreview.net/forum?id=b2Ni828As7

Guarding Integration Methods — Results

Method	Accuracy (%)		Precision (%)	
	Transformer	Guard	Transformer	Guard
Trager	**98.21**	67.55	**92.21**	15.88
MeijerG	**96.78**	88.72	**89.58**	57.78
PseudoElliptic	**99.28**	62.61	**62.86**	2.03
Gosper	**94.04**	92.51	**92.08**	80.21

Results for methods **with** a guard.

Method	Accuracy (%)	
	Transformer	Guard
Default	**94.86**	82.15
DDivides	**94.13**	28.18
Parts	**93.10**	37.05
Risch	**94.53**	89.35
Norman	**95.74**	71.67
Orering	**97.21**	37.88
ParallelRisch	**97.82**	82.73

Results for methods **without** a guard.

The "guard" in this case is simply predicting to use the method every time.

Representing Mathematical Data

Is treating mathematics as a string the right approach? It does not reflect the commutativity in the expressions being integrated! Are there alternatives?

Text Sequence

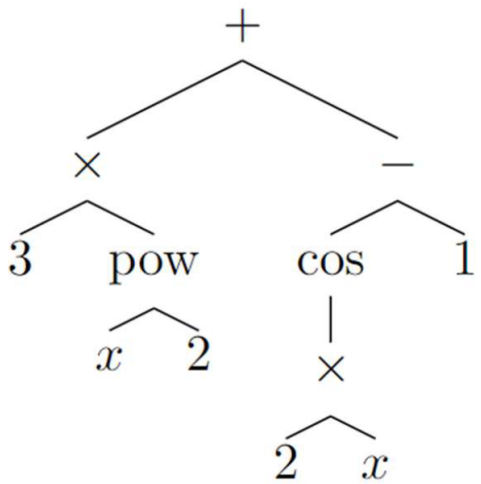
['add',
'-1',
'add',
'mul',
'3',
'pow',
'x',
'2',
'cos',
'mul',
'2',
'x']

$$3x^2 + \cos(2x) - 1$$

Feature Vector

	Vars	Terms	Has Trig	...
$3x^2 + \cos(2x) - 1$	1	3	True	...

Graph



Experiments

Experiments to compare tree vs. sequential embeddings; and two-stage architecture vs. simpler classification approach.

Train on 1M examples from the data generators. Evaluate on (a) hold out test set; and (b) a separate Maplesoft test suit.

Key evaluation metric used is length of the output expression (correctness is guaranteed; we are looking for simplicity).

R. Barket, M. England, and J. Gerhard.

Symbolic Integration Algorithm Selection with Machine Learning: LSTMs vs Tree LSTMs.

Proc. ICMS '24, LNCS 14749, pp. 167–175, 2024.

https://doi.org/10.1007/978-3-031-64529-7_18

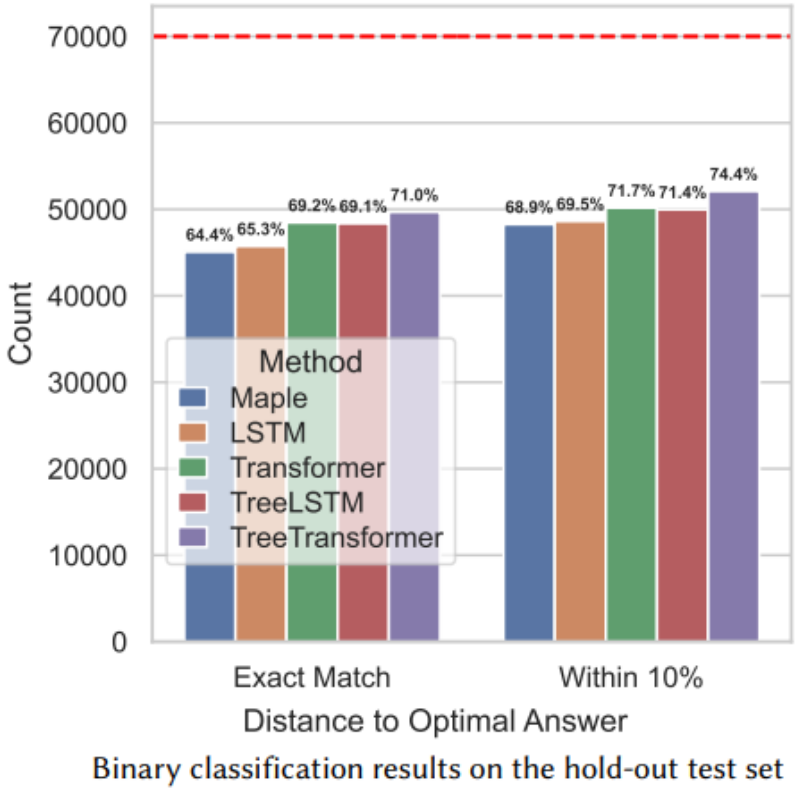
R. Barket, M. England, and J. Gerhard.

Tree-Based Deep Learning for Ranking Symbolic Integration Algorithms.

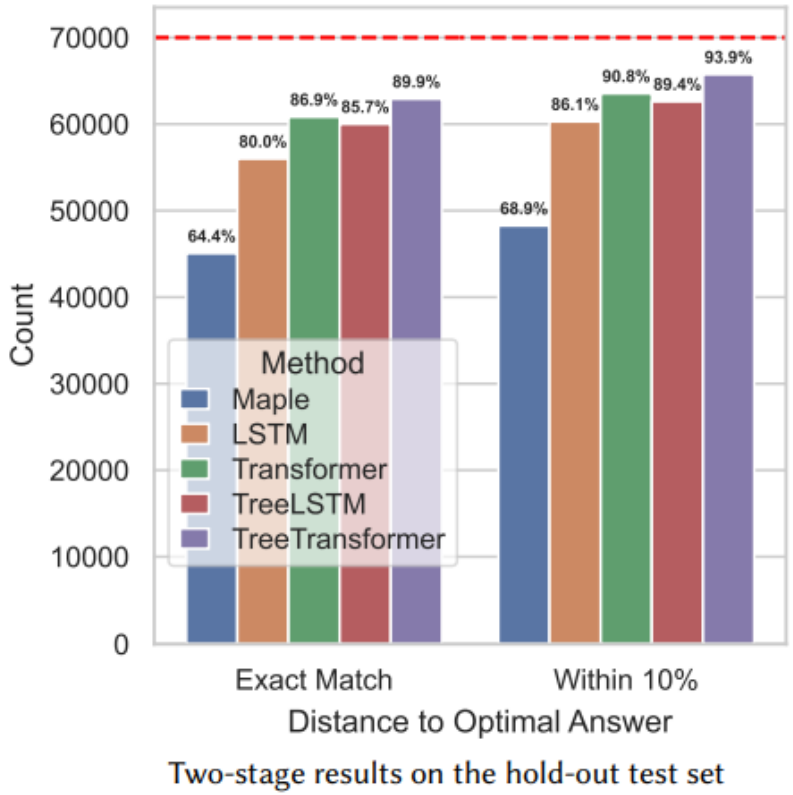
Submitted, 2025. Preprint: arXiv:2508.06383

https://doi.org/10.48550/arXiv.2508.06383

Results on Hold Out Test Data



Binary classification results on the hold-out test set

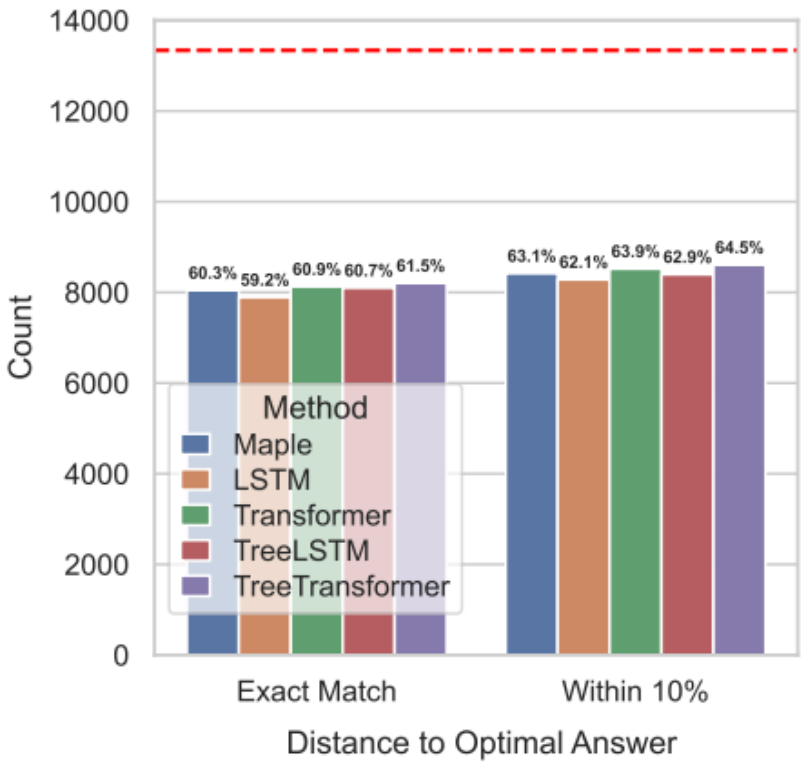


Two-stage results on the hold-out test set

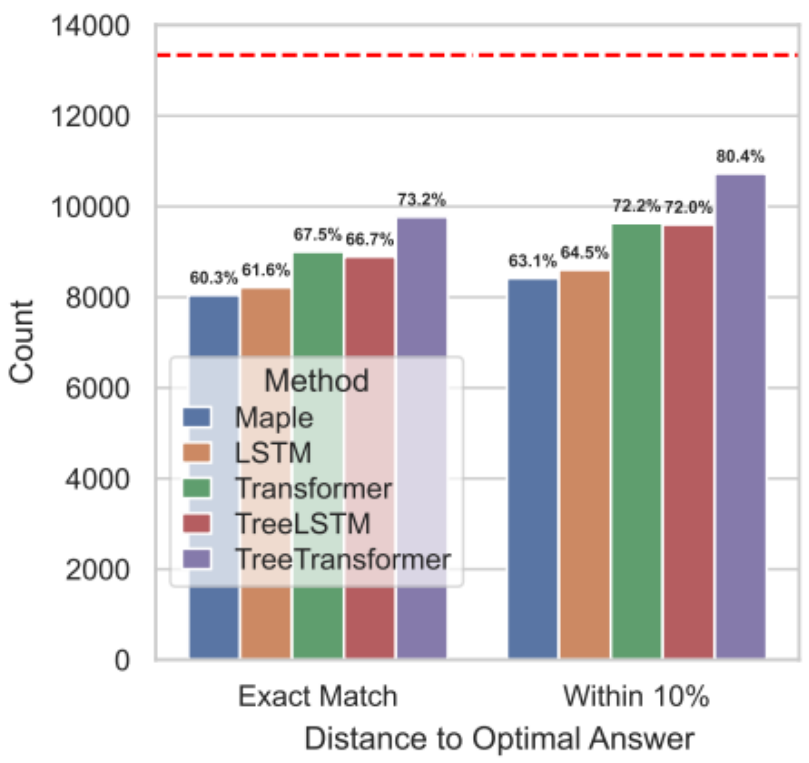
Lessons Learnt

Embedding choice: Tree embeddings stronger than sequential.
Algorithm selection benefits from a two-stage architecture.

Results Validation on Maple Test Suite



Binary classification results on the Maple Test Suite



Two-stage results on the Maple Test Suite

Lesson Learnt

Evaluate generalisation carefully: Testing with truly independent dataset shows base gains weren't real; more sophisticated architecture and embeddings are needed for true learning.

Integrand	Maple Result	Tree Transformer
$\frac{\ln(x^4+1)}{x}$	$\ln(x)\ln(x^4+1) - \ln(x)\ln\left(\frac{\frac{\sqrt{2}}{2}+\frac{i\sqrt{2}}{2}-x}{\frac{\sqrt{2}}{2}+\frac{i\sqrt{2}}{2}}\right) - \operatorname{dilog}\left(\frac{\frac{\sqrt{2}}{2}+\frac{i\sqrt{2}}{2}-x}{\frac{\sqrt{2}}{2}+\frac{i\sqrt{2}}{2}}\right) - \ln(x)\ln\left(\frac{-\frac{\sqrt{2}}{2}+\frac{i\sqrt{2}}{2}-x}{-\frac{\sqrt{2}}{2}+\frac{i\sqrt{2}}{2}}\right) - \operatorname{dilog}\left(\frac{-\frac{\sqrt{2}}{2}+\frac{i\sqrt{2}}{2}-x}{-\frac{\sqrt{2}}{2}+\frac{i\sqrt{2}}{2}}\right) - \ln(x)\ln\left(\frac{-\frac{\sqrt{2}}{2}-\frac{i\sqrt{2}}{2}-x}{-\frac{\sqrt{2}}{2}-\frac{i\sqrt{2}}{2}}\right) - \operatorname{dilog}\left(\frac{-\frac{\sqrt{2}}{2}-\frac{i\sqrt{2}}{2}-x}{-\frac{\sqrt{2}}{2}-\frac{i\sqrt{2}}{2}}\right) - \ln(x)\ln\left(\frac{\frac{\sqrt{2}}{2}-\frac{i\sqrt{2}}{2}-x}{\frac{\sqrt{2}}{2}-\frac{i\sqrt{2}}{2}}\right) - \operatorname{dilog}\left(\frac{\frac{\sqrt{2}}{2}-\frac{i\sqrt{2}}{2}-x}{\frac{\sqrt{2}}{2}-\frac{i\sqrt{2}}{2}}\right)$	$\frac{\operatorname{polylog}(2,-x^4)}{4}$
$\operatorname{AiryAi}(x)^2$	$-\frac{18^{2/3}2^{1/3}3^{2/3}x^4\operatorname{BesselI}\left(\frac{2}{3},\frac{2x^{3/2}}{3}\right)^2}{162(x^{3/2})^{4/3}} + \frac{2\cdot 18^{2/3}2^{1/3}3^{2/3}x\operatorname{BesselI}\left(\frac{2}{3},\frac{2x^{3/2}}{3}\right)^2}{81(x^{3/2})^{4/3}} + \frac{18^{2/3}2^{1/3}3^{2/3}x^4\operatorname{BesselI}\left(\frac{5}{3},\frac{2x^{3/2}}{3}\right)^2}{162(x^{3/2})^{4/3}} - \frac{18^{2/3}2^{1/3}3^{2/3}\sqrt{x}\operatorname{BesselI}\left(\frac{1}{3},\frac{2x^{3/2}}{3}\right)\operatorname{BesselI}\left(\frac{2}{3},\frac{2x^{3/2}}{3}\right)}{81} - \frac{18^{2/3}2^{1/3}3^{2/3}\operatorname{BesselI}\left(\frac{1}{3},\frac{2x^{3/2}}{3}\right)\operatorname{BesselI}\left(\frac{5}{3},\frac{2x^{3/2}}{3}\right)}{81} + \frac{18^{2/3}2^{1/3}3^{2/3}x^2\operatorname{BesselI}\left(\frac{4}{3},\frac{2x^{3/2}}{3}\right)\operatorname{BesselI}\left(\frac{2}{3},\frac{2x^{3/2}}{3}\right)}{81} + \frac{3^{5/6}2^{2/3}\Gamma\left(\frac{5}{6}\right)x^3\operatorname{hypergeom}\left([\frac{5}{6},1],[\frac{4}{3},\frac{5}{3},2],\frac{4x^3}{9}\right)}{36\pi^{3/2}}$	$\operatorname{AiryAi}(x)^2x - \operatorname{AiryAi}(1,x)^2$

Summary: Our two stage architecture with tree transformer is the clear state-of-the-art for the problem; should be part of Maple 2027.

Outline

LIGs on Transformer Guards

We experimented with Layered Integrated Gradients (LIGs) applied to our embedding layer to compute an attribution score for each input token.

One pattern spotted in these explanations is the high scores applied to the `abs` token for the Risch sub-algorithm classifier.

[CLS] mul INT+ CONST1 mul x pow **abs** cos mul INT+ 2 pow x INT+ 2 INT- 1

Upon inspection: this sub-algorithm cannot be applied when there is `abs` in the input, but the current Maple guard code did not check for this. An improvement offered to the CAS developers independent of whether they embed the ML or not.

Thanks for Listening

Matthew.England@coventry.ac.uk
https://matthewengland.coventry.domains