SYMBA: Symbolic Computation of Squared Amplitudes in High Energy Physics with Machine Learning

8 May 2023

Abdulhakim Alnuqaydan
University of Kentucky

Outline

- Introduction
- SYMBA (Symbolic Computation of Squared Amplitudes)
- Future

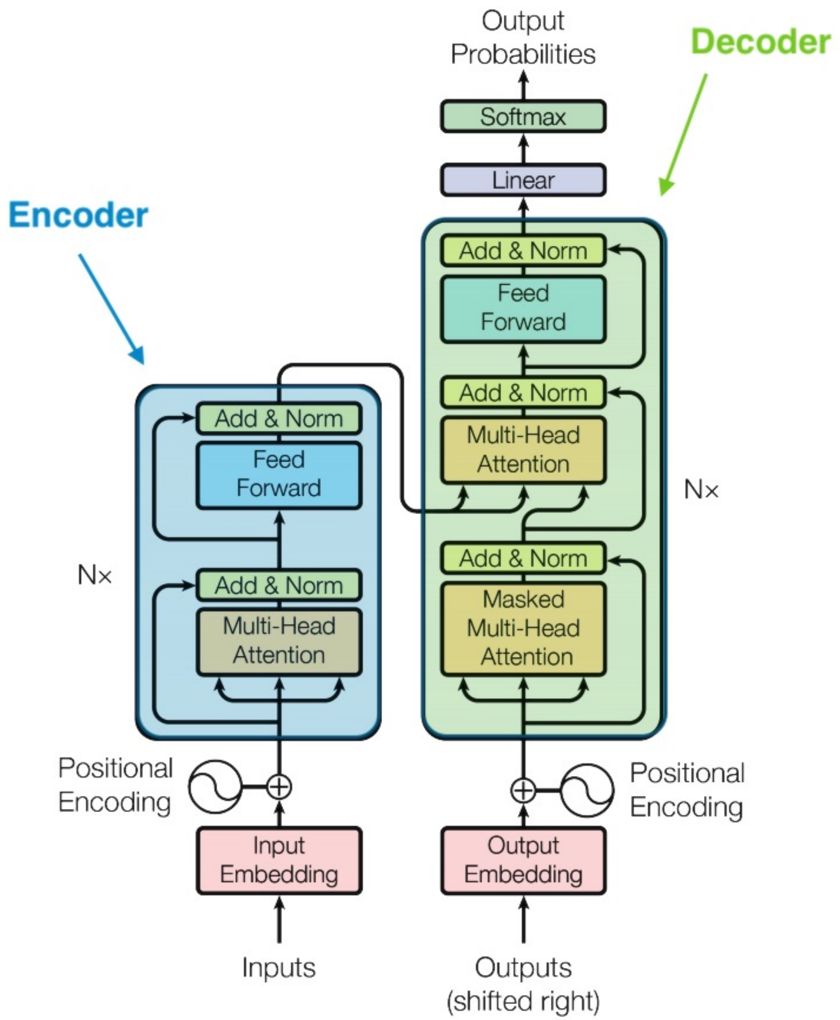
Sequence to sequence (seq2seq):

- Machine learning model that maps an input sequence (words, symbol,...) to an output of sequence
- Used in:
  - Natural language processing (NLP) tasks: translation, summarizations, text generations (GPT-3/GPT-4)
  - Image captioning
  - Symbolic mathematical calculations (Integrations, solving ODEs, …) [Lample, Charton, 2019]
- One of the most powerful model: **Transformer Model**

Transformer

A *transformer* Deep learning model that adapts the mechanism of **self-attention**, differentially weighting the significance of each part of the input data.

It was introduced in 2017 by a team at Google Brain. (Vaswani et al, 2017)



SYMBA (Symbolic Computation of Squared Amplitudes)

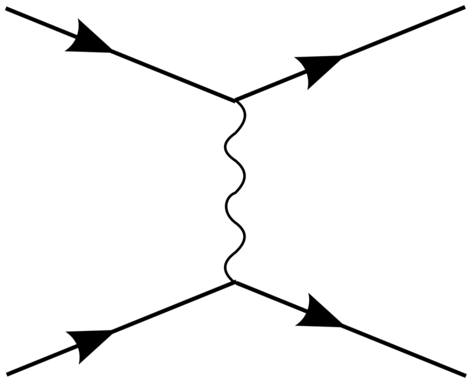
We use Transformer model to compute symbolically the square of the particle interaction amplitude, a key element of a cross section calculation.

Abdulhakim Alnuqaydan
PhD student
University of Kentucky

Sergei Gleyzer
Professor
University of Alabama

Harrison Prosper
Professor
Florida State University

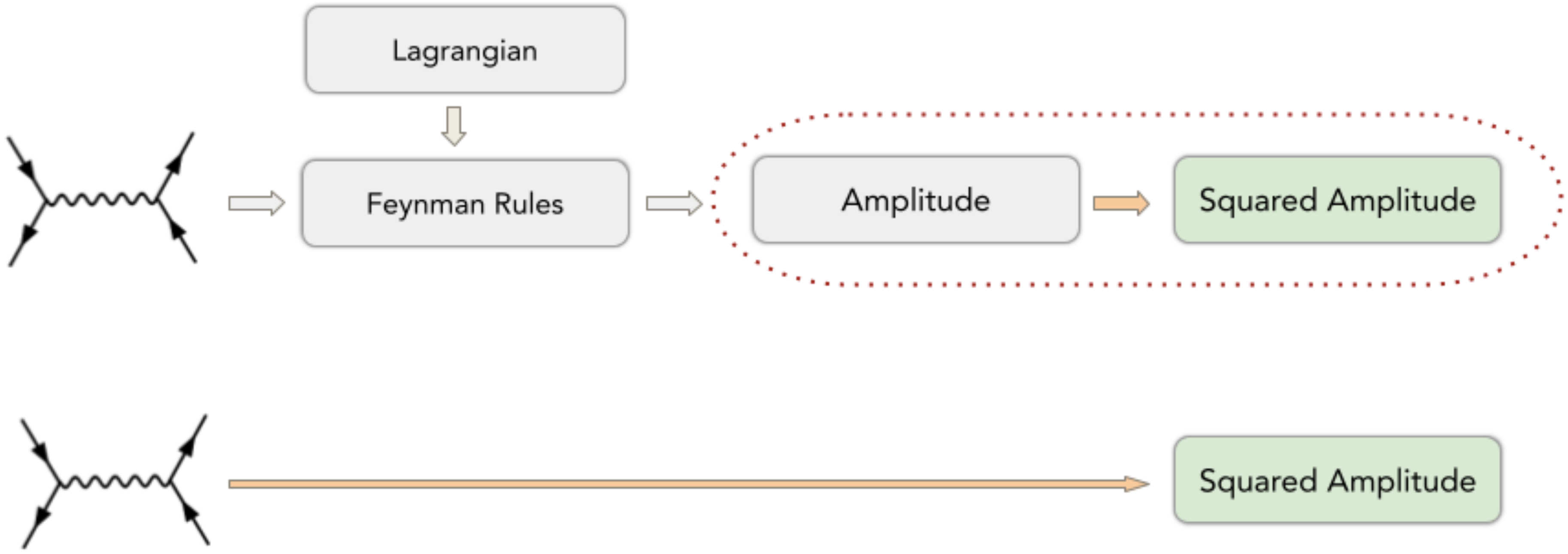
Squared amplitude and cross section:



$$i\mathcal{M} \longrightarrow |\mathcal{M}|^2$$

Polarization Averaging
Lorentz index
Color factors
Matrix multiplication
Dirac algebra
Traces
Integration

Cross section $\boxed{\sigma \sim |\mathcal{M}|^2}$


Lagrangian
Feynman Rules
Amplitude
Squared Amplitude
Squared Amplitude

Dataset:

- Generate 2→2 processes in all Standard Model at tree-level and compute their squared amplitudes using MARTY
- Generate 2→3 processes in QED and QCD at tree-level and compute their squared amplitudes using MARTY

MARTY

Pairs of two types:

(amplitude, squared amplitude)

(Feynman diagram, squared amplitude)

Data Preparation:

Simplifying squared amplitude:

Tokenization:

- The amplitudes are tokenized by operator (tensor) and its indices
- The squared amplitudes are tokenized by each mass, product of momenta and numerical factor
- Exclude expressions longer than 264 tokens

Training:

- The model has 6 layers and 8 attention-heads, with 512 embedding dimensions
- We use sparse categorical *crossentropy* as the loss function, the Adam optimizer with a learning rate of 0.0001 and a batch size of 64.
- The training was performed for 50-100 epochs on two CASCADE-NVIDIA V100 GPUs which took about 12-24 hours.

Accuracy metrics:

We assessed the accuracy of the trained model in the standard way, namely by applying the model to a test dataset consisting of 500 amplitudes that were not used to train the model. Three measures of accuracy were considered:

1. **Sequence Accuracy:**
   The relative number of squared amplitudes correctly and exactly predicted.

2. **Token Score:**
   The relative number of tokens (i.e., symbols) that were predicted correctly in the right place in the sequence:

$$Token\ Score = \frac{n_c - n_{ex}}{n_{act}} \quad \%,$$

Accuracy metrics(cont.):

3. **Numerical Error:**

   Random numbers between {10, 100} are plugged into the variables (momenta) in the squared amplitude and we compare the predicted numerical value of the squared amplitude with the actual numerical value:

$$Numerical\ Error = \frac{x_{act} - x_{pred}}{x_{act}},$$

Results:

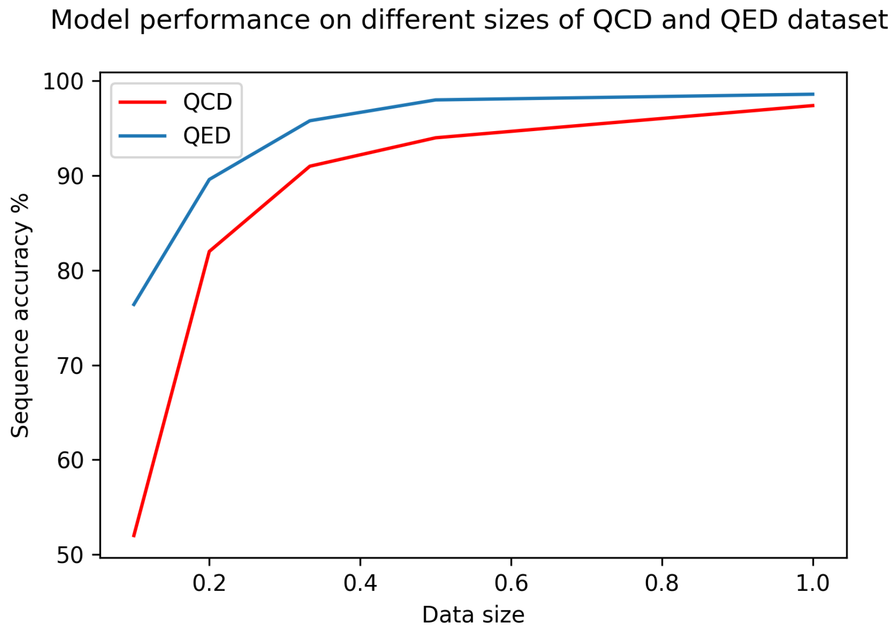
Alnuqaydan *et al* 2023 *Mach. Learn.: Sci. Technol.* **4** 015007

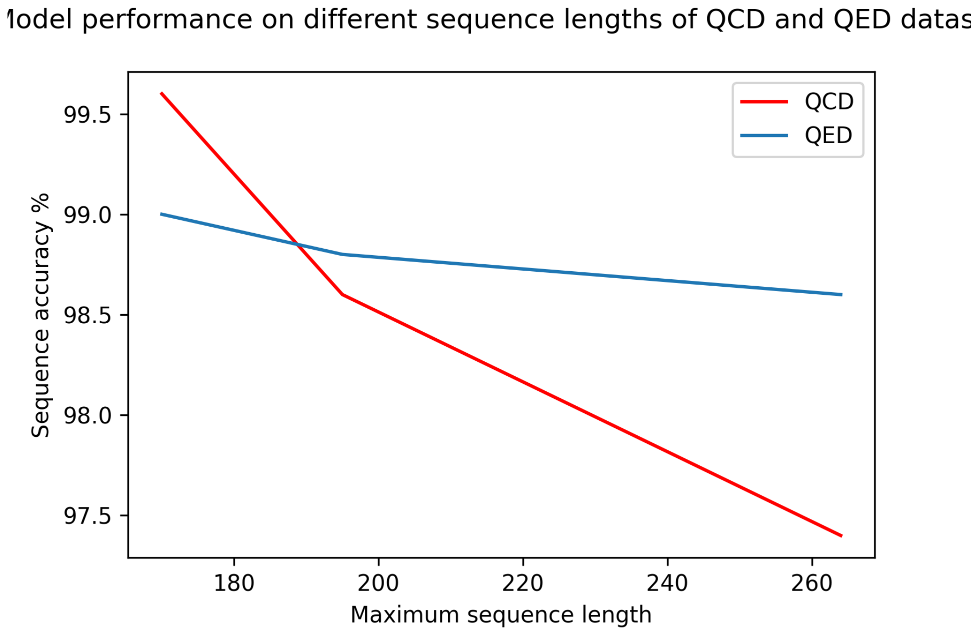
	Data size	Sequence Accuracy
QED (amplitude)	251K	**99%**
QCD (amplitude)	140K	**97%**
EW 2to2 (amplitude)	285K	**90%**
QED (diagram)	258K	**99%**
QCD (diagram)	142K	**73%**
EW 2to2(diagram)	259K	**93%**

Average time of inference $\leq$ 2 sec

Remarks:

- High dependency on: **sequence length** and **data size**



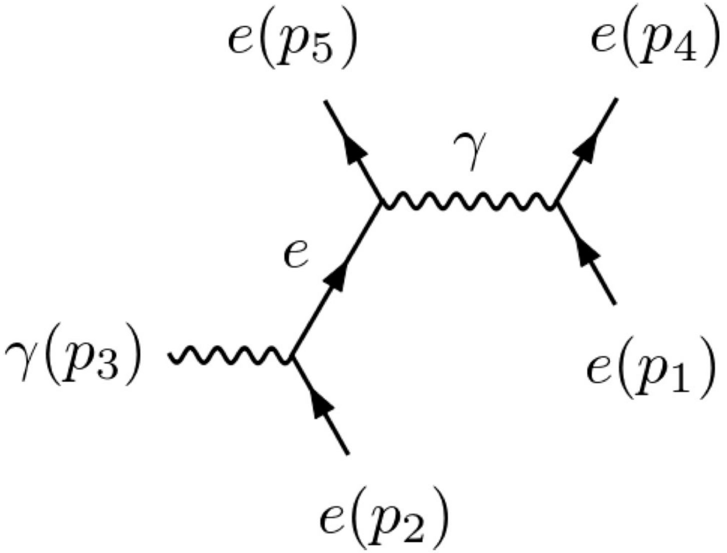


Future:

- Improve decoding (beam search, dimensional decoding
- Uncertainty
- Include more theories, more final states and higher orders
- Transformer for long sequence

Thank you ..

Back-up



- ***The amplitude*** $(e\ e \to e\ e\ \gamma)$:

$$i\mathcal{M} = \frac{\frac{1}{2}\, ie^3\big(p_{3\rho}\,\gamma_\epsilon^\rho\,\gamma_{\rho\eta}\,A_j^{\rho *}(p_5)\,\mathbf{e}_{i\eta}^*(p_4)\,\mathbf{e}_{l\epsilon}^*(p_3)\,\mathbf{e}_{k\delta}(p_2)\,\mathbf{e}_{i\delta}(p_1)\ -\ \frac{1}{2}\,p_{5\sigma}\,\gamma_{\rho\epsilon}\,\gamma_\epsilon^\rho\,\gamma_{\rho\eta}\,\gamma_\epsilon^\sigma\,A_j^{\rho *}(p_5)\mathbf{e}_{i\eta}^*(p_4)\,\mathbf{e}_{l\epsilon}^*(p_3)\,\mathbf{e}_{k\delta}(p_2)\,\mathbf{e}_{i\delta}(p_1)\big)}{((m_e^2 - \vec{p_2}\,.\vec{p_4})\ *\ \vec{p_3}.\,\vec{p_5})}$$

- ***The squared amplitude*** $(e\ e \to e\ e\ \gamma)$:

$$|\mathcal{M}|^2 = -\frac{e^6}{((\vec{p_3}.\vec{p_5})^2 * (m_e^2 - \vec{p_2}.\vec{p_4})^2)}(2m_e^6 + m_e^4 * (-\vec{p_1}.\vec{p_3} - \vec{p_1}.\vec{p_5} - \vec{p_2}.\vec{p_4} + 2\vec{p_3}.\vec{p_5}) + m_e^2 * (\vec{p_1}.\vec{p_2} * \vec{p_3}.\vec{p_4} +$$
$$\vec{p_1}.\vec{p_2} * \vec{p_4}.\vec{p_5} + \vec{p_1}.\vec{p_4} * \vec{p_2}.\vec{p_3} + \vec{p_1}.\vec{p_4} * \vec{p_2}.\vec{p_5} + \vec{p_1}.\vec{p_5} * \vec{p_3}.\vec{p_4} - \vec{p_2}.\vec{p_4} * \vec{p_3}.\vec{p_5}) - \vec{p_1}.\vec{p_2} * \vec{p_3}.\vec{p_5} * \vec{p_4}.\vec{p_5} - \vec{p_1}.\vec{p_4} * \vec{p_2}.\vec{p_5} * \vec{p_3}.\vec{p_5})$$

Results:

Table 1: Amplitude-squared amplitude Model results

	Training Size	Sequence Acc.	Token Score	RMSE
QED (sequence)	251K	98.6%	99.7%	1.3×10^{-3}
QCD (sequence)	140K	97.4%	98.9%	8.8×10^{-3}
(QED + QCD) on QED	391K	**99.0%**	99.4%	2.5×10^{-3}
(QED + QCD) on QCD	391K	**97.6%**	98.8%	6.8×10^{-3}
QED (diagram)	258K	**99.0%**	99.7%	9.3×10^{-4}
QCD (diagram)	142K	73.4%	82.0%	0.3