ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25feb1287

Modular Square One-Way Function & Square Root Algorithm (Part-2): AI practical approaches for applying the paper results in GPT

(Sampling Codec- Normalization- Tokenization- Clustering-Compression-Fine Tunning)

Ahmed Mohammed Al-Fahdi[1]

[1]Sultanate of Oman

[1]MSc communication engineering, University of Birmingham

Publication Date: 2025/04/28

Abstract: This paper is built upon a previous paper entitled as "Modular Square One-Way Function& Square Root Algorithm: Analyzing the algorithm for randomness, regularity schematic (codec system) and vector normalization ". In that paper the modular square one-way function was analyzed yields the quadratic residue pattern numeric analyzation in the result section. Analyzing the integer factorization results leads to un expected schematic regularity regarding the irrational part of the remainder (decimal expansion) of nonperfect square root. Such regularity was surprising as the expected results assumed to be random. Rounding such rational numbers and normalizing it yield to what is innovatively called modular factor symbol similar to Legendre symbol. Such codec pattern has characteristics of Hilbert envelope, skewness around perfect root pattern with Hann window. In GPU, such calculations could be computed fastly using IEEE-754 [1] standard for rounding irrational part of the nonperfect square (decimal expansion) with floating point as what mentioned in inverse square root [1]. All above, illuminating an idea of the statistical analyzation for the root mean square error (RMSE). RSME is a powerful estimator of the prediction models used in the artificial intelligence AI especially for the reinforcement learning (RL). As a new approach in AI Google DeepMind Researchers looking through regression analysis algorithm tuning and representing the numerical values as discrete tokens for large language model (LLM). Such data set tokenization and tuning algorithm are helpful for the speed and the predictability of the model as it hase been recognized in the Deep Seek.[4] .

Up on all above and considering AI as a new evaluation approach, this paper will discuss the implementation aspects of such innovative results in sampling, tokenizing, clustering and compressing the base model of the GPT a long with fine tuning Neural Network (NN) reasoning of the Reinforcement model.

Key words: *AI, tokenization, Riemann Integral, Hilbert Envelope, Hann Window, Skewness Generative Pre-Trained Transformer (GPT), Auto-Regressive Transformer, Base Model, Reward Modeling, RPE, Vision-Based Estimation, SL, Sequence-Based Estimation, Temporal Difference (TD), Root Mean Square Error (RMSE), Statistical Analysis ,Numerical Analysis, Regression Analysis, Pigeonhole Clustering, Matryoshka Structure, Reinforcement Learning RL, RSME Estimator, Floating Point IEEE-754,Auto-Regression Tuning, Parameter Tuning Of Decoder Unit, Vector Normalization, Normalized Token, Minimum Value Estimator (MVE), Statistical Estimation (Mean, Median),Histogram Based Regression.*

How to Cite: Ahmed Mohammed Al-Fahdi (2025), Modular Square One-Way Function & Square Root Algorithm (Part-2): AI practical approaches for applying the paper results in GPT (Sampling Codec- Normalization- Tokenization- Clustering Compression-Fine Tunning). *International Journal of Innovative Science and Research Technology,* 10(2), 2565-2571. https://doi.org/10.38124/ijisrt/25feb1287

I. INTRODUCTION

These days, there is a revolution in Artificial Intelligence (AI) especially in the field of generative pre-trained transformer (GPT) of large language model (LLM). The performance of such transformers is measured by the speed of response and the result precision. These parameters depend on the models' algorithm and the largeness of the data set of the pre- training corpus which may take long time. There are different approaches that researchers working on to minimize the time and data taking in concern, with the trade off the precession of the estimated value. One of the approaches is to have base model that has a collection of very massive large data known as corpus. Searching through such library consume effort and time, so it is practical to implement an efficient tokenization process stored in an effective way for easy faster search with no collision precession. This tedious number of tokens needs a tedious storage, so it is practical to use lossless compression. Training the model using such massive huge data needs lookup algorithm a long with fine tuning supervised estimator. Later on, reinforcement learning could be applied for such estimator result on redistilled model with fast parallel computation clusters. [1],[2],[3],[5] ,[6]

Estimator is defined to be a measure of given quantity based on the observed data. In general AI, it represents the precision of the transformer output in coordination of the true values. Such parameter is used in the feedback loop of the AI algorithm to make the output more precise within a specific interval. This parameter is called error signal in supervised learning (SL). In Reinforcement learning (RL) and generative transformer (GT), such parameter is called reward prediction error (RPE). It guide the process to maximize the future rewards of the output. There are different tools that used to estimate such error. Root mean square error RMSE is one of the estimators is one of the metrics widely used in different types of AI in predictive modeling and regression tasks but it does not used directly in the reinforcement learning (RL) as it used to evaluate the overall performance of the model. The usual way for such RL model is the temporal difference and send a signal to update the agent's policy. This methodology is known as tuning and a scalable sliding window over time is one of powerful tool for such process [1],[2],[3],[5] ,[6]

II. METHODOLOGY AND PAPER ORGANIZATION

Referring to previous paper "Modular Square One-Way Function & Square Root Algorithm: Analyzing the algorithm for randomness, regularity schematic (codec system) and vector normalization", which analyzed the expansion decimal of the irrational part of nonperfect square. Leading to an innovative factor named as normalized modular factor (M). The resulting codec has powerful tuning characteristics such as skewness, window, ascending pattern. RSME is also an analyzed tool with powerful computing trick of IEEE-754 standard with floating point for the rational part [1]. Throughout this paper such results analyzed for the base-model and reinforcement model of the LLM.[1],

III. RESULTS ANALYSIS AND APPLICATIONS

➢ *Overview*

This part of the paper handles the application of the innovative state-of-art results -paper part-1- to optimize the generative pre-trained model (GPT) of the large language model (LLM). Starting with the base-model, resulting innovative codec schematic of the modular factor could be used for sampling, tokenization, clustering and lossless compression for the pre-training corpus. Secondly, the resulting model tuned using a supervised fine tuning (SFT) for the feedback loop error signal which is here called the reward prediction error (RPE) for the Reinforcement Learning (RL). Root mean square error (RMSE) is a widely used metric tool in different types of AI. It is used for the accuracy of the predictive models in regression parts especially for supervised learning (SL) with different types of regressions. It is used for decline regression weight for supervised fine tuning (SFT). Similar to that applies with deep learning Regressions. It is used for smoothen the output of NLP (natural language processing) along with image depth or quality. In addition, RMSE used for tuning different models hyperparameters and engineering features of machine learning. [1],[2],[3]

➢ *Base model*

- *Codec (Sampling)*

Although the modular square algorithm has random modular, the rational part of such reminders has innovative kind of regularity. The innovative way here is by taking threshold (.25 & .75) and using the revert way of bit hacking used in (fast inv sqrt) - for approximating the irrational part which leads to that kind of skewing regularity codec with sampling rate equal to the perfect root. Such sampling codec could be used as Hilbert envelop codec.[1]

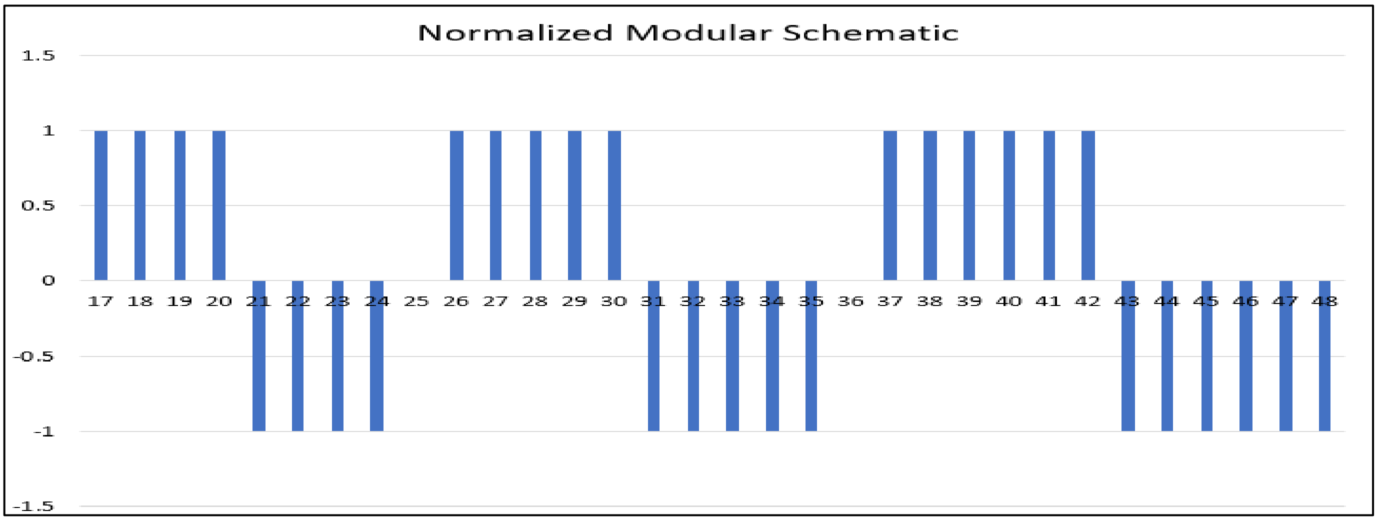


Fig 1 Mirrored skewing of ascending widow (temporal sliding) result from Quantized normalized approximated irrational root part for sequence squares (17-48) [1]

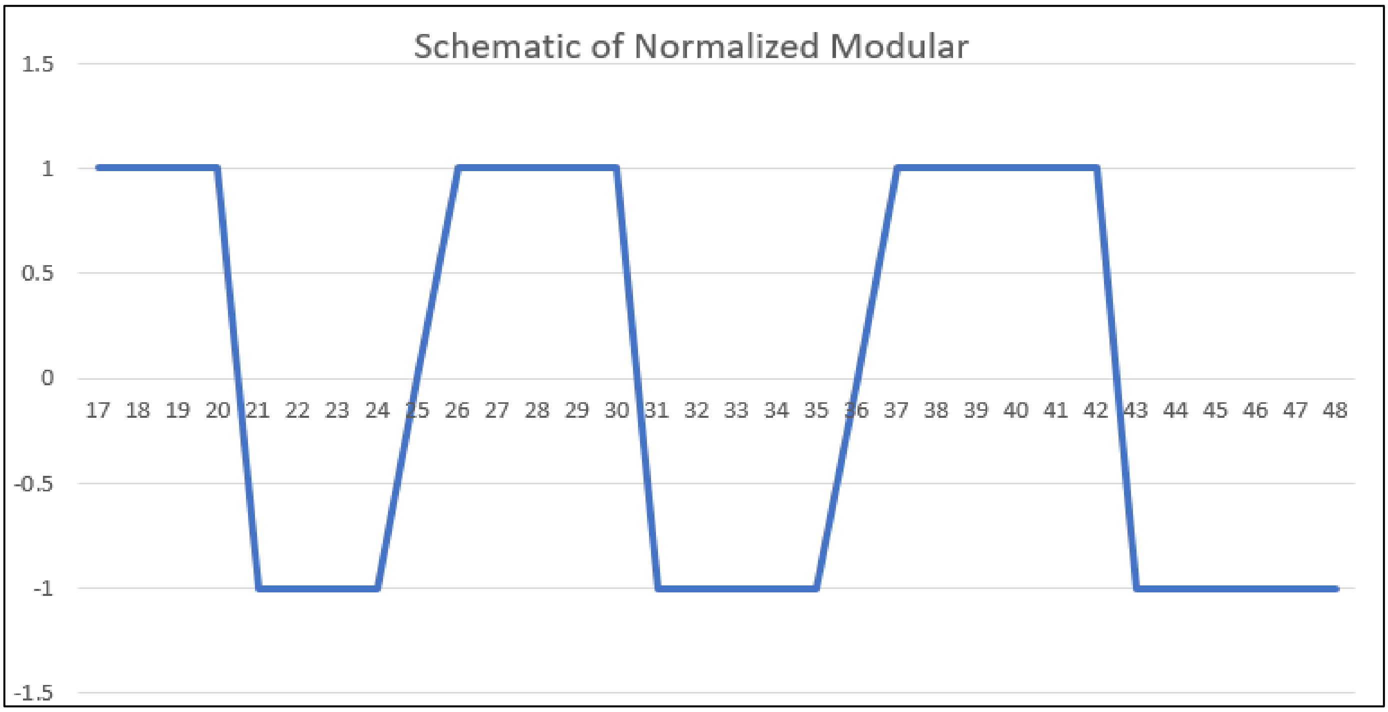


Fig 2 skewing signal of normalized approximated irrational root part for sequence squares (17-48) [1]

- *Tokenization*

uext corpus model is sampled using byte pair encoding (BPE) to get the model format. The name-value pair arguments of the sampled corpus could tokenized using the discrete form of the quantized skewing signal (FFT) for the base model Figure(3) [1].Such kind of novel quantization is useful to handle the complexity of texts and other diverse forms and store in cell array .

Similar to what happen with Riemann integral method for definite integral could followed up to dividing the area into very small rectangle which could be achieved rising the sampling rate exponentially by square [8]. which leads to highly efficient approximation at large exponents and repeating it until precision of target size Figure(4) [1].

Similar to that this codec pattern used with audio/video tokenization with extra advantage of the Skewness feature as well as the stream cipher codec with IEEE754 [1] standard bit shift floating point in similar way of LFSR.[1], [2],[3]

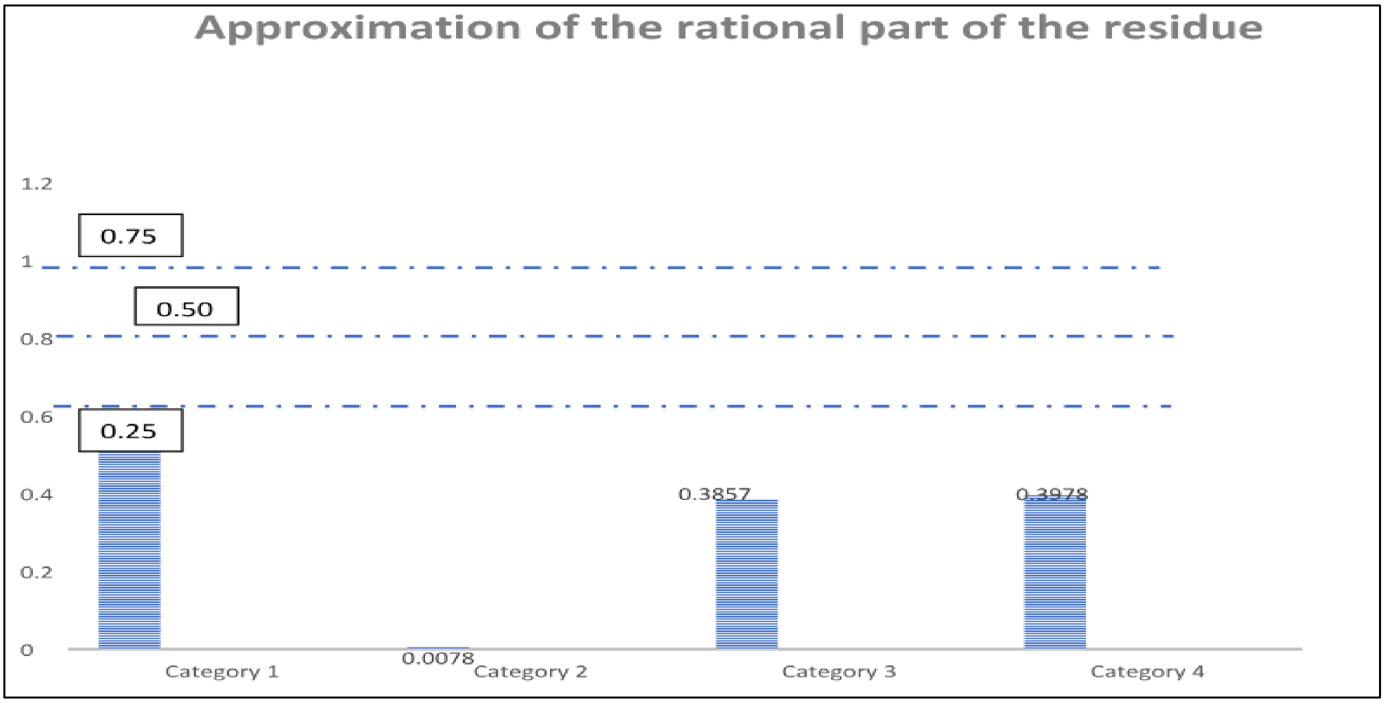


Fig 3 approximation procedure for the irrational part of roots of random nonperfect square [1]

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25feb1287

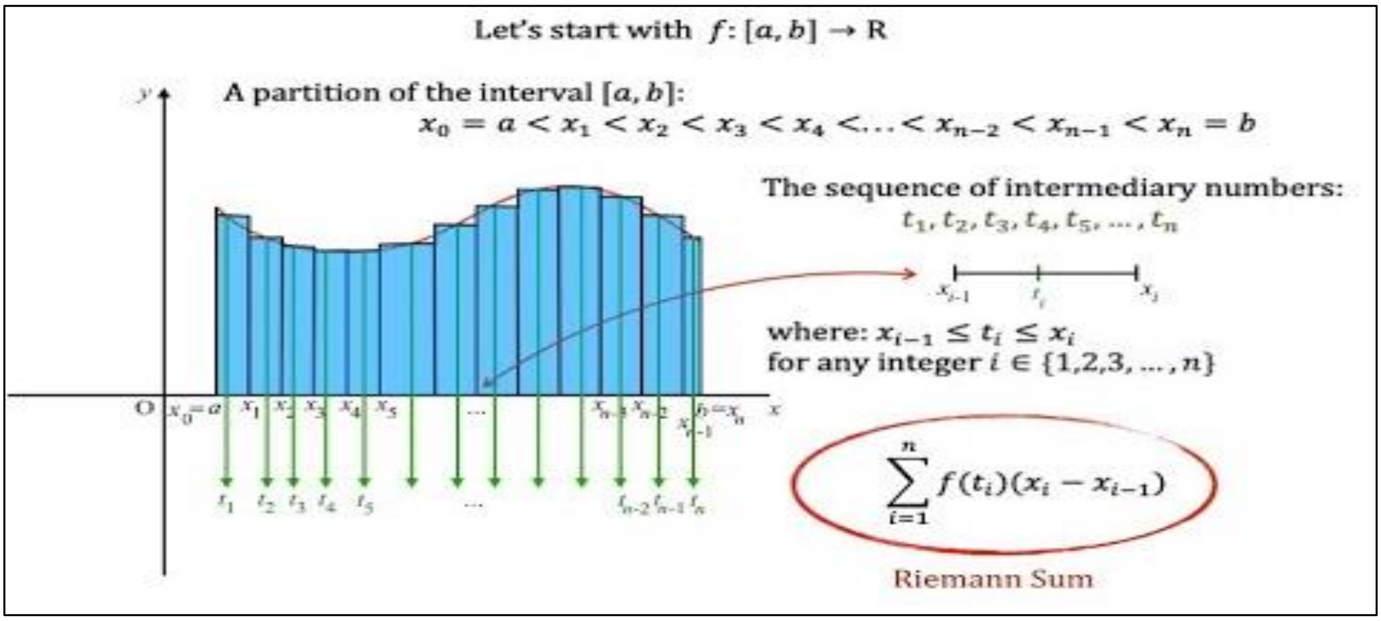

Let's start with $f: [a,b] \to \mathrm{R}$
A partition of the interval $[a,b]$:
$x_0 = a < x_1 < x_2 < x_3 < x_4 < \ldots < x_{n-2} < x_{n-1} < x_n = b$
The sequence of intermediary numbers:
$t_1, t_2, t_3, t_4, t_5, \ldots, t_n$
x_{i-1} t_i x_i
where: $x_{i-1} \le t_i \le x_i$
for any integer $i \in \{1,2,3,\ldots,n\}$
$\sum_{i=1}^{n} f(t_i)(x_i - x_{i-1})$
Riemann Sum
y
O
x
$x_0=a$ x_1 x_2 x_3 x_4 x_5 ... x_{n-3} x_{n-2} x_{n-1} $b=x_n$
t_1 t_2 t_3 t_4 t_5 ... t_{n-2} t_{n-1} t_n

Fig 4 Riemann integral method for continuous data signal quantization [1],[8]

- *Mirrored skewing of ascending widow (sliding)*

As it shown in bellow figure (5), the modular factor extracted from the irrational decimal remainder of square root (paper part-1) characterize gradual ascending widow with skewness property. Such characteristic is useful for extracting different kind of features from data especially that evolving temporal dynamic pattern like audio and video. In addition, this sliding window over time could be used for tuning the feed forward network (FFNN)-like that used with Deep Seek - as well recurrent neural network (RNN) after calculating the (mean, max, min ,standard deviation, etc.) of each window and feed it to the temporal process (FFNN, RNN).As an innovative way for deep integration of visual and textual multimodal, such kind of skewing sliding window could be used to patch image and tokenize text to combine the emending into unified representation space. The temporal sliding feature of such window could also be used for dynamic frame rate tuning - frames per second FPS[4]. Such dynamic frame rate is useful for video processing in term of monitoring performance and adjusting frame rate. That is mean it allows fine-grained over dynamic frame rate [1],[2],[3],[4]

✓ *Notice:*

Recently (20 Feb 2025) Qwen2.5 using similar concept for their decoder!! for unified space Figure (5)

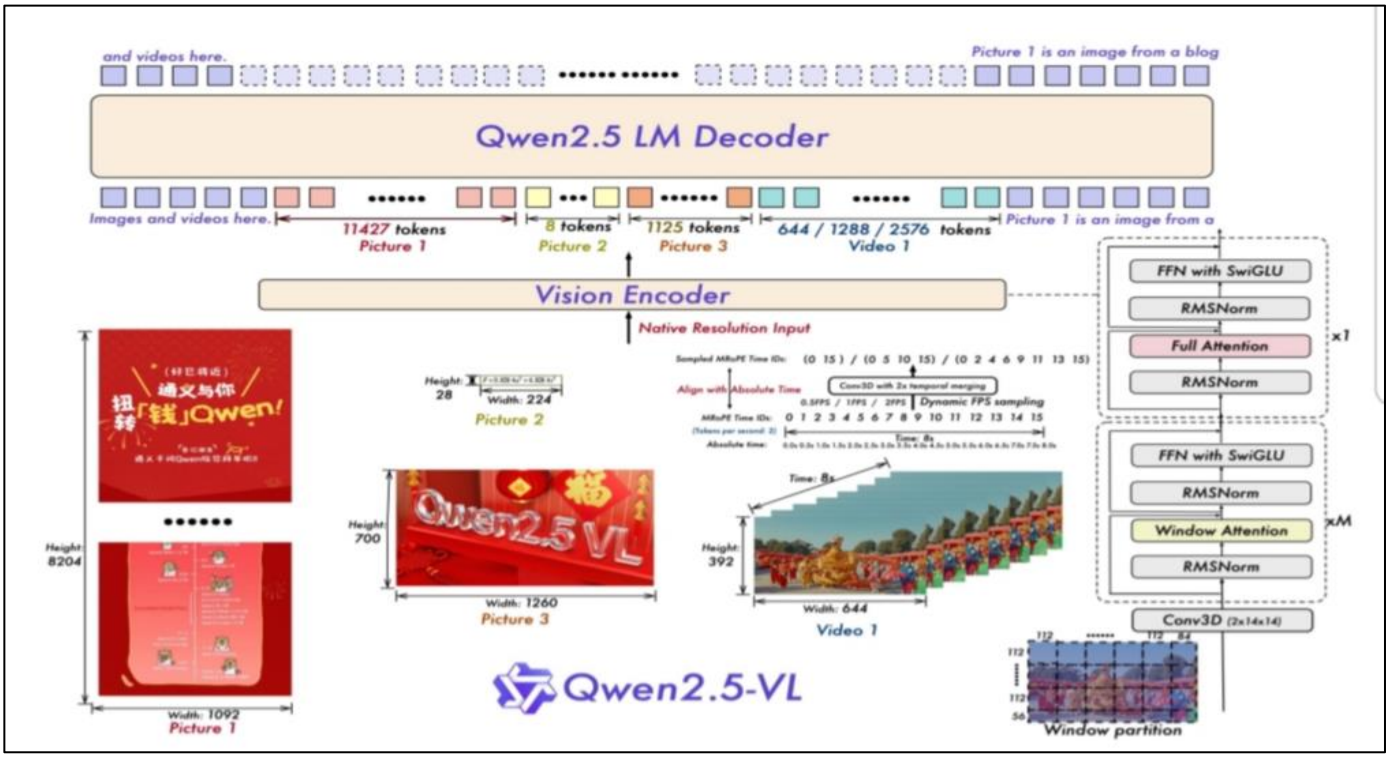


Fig 5 Qwen 2.5 Decoder using sliding window and RMS Norm.[4]

- *Distilled Clustering and pigeon principle*

There is an essential common property of all regular languages. It is described by what is called the pumping lemma for regular languages. It says that there is a middle section of the string repeated an arbitrary number of times for all sufficiently long strings in a regular language. This means that the language is not regular if and only if it does not satisfy the pumping lemma. The mathematical representation that make pumping lemma practical in the field of AI is called the pigeon principle.

The general form of pigeon principle says that, if you have m pigeonholes rack and you want to distribute n pigeons uniformly inside it with probability of $\frac{1}{m}$ [7], then at least one pigeonhole will hold more than one pigeon with probability of :

$$P = 1 - \frac{(m)_n}{m^n} \qquad (1)$$

Referring to the fact that the square root irrational part (decimal expansion) are nonrepeating infinite, tokenization lead to rational approximation where the pigeonhole plays significant role for clustering.[1]

After the tokenization, clustering in lossless pigeon principle based-array leading to distilled low space and reduce the searching time and avoid collision of the hashed (tokenized data). This mean it is powerful in term of lower cash size.[6]

Such clustering is different from matryoshka, a head, clustering MRL (Matryoshka Representation Learning) which is used usually for the main nested structure.[6]

- *Lossless compression and Sense disambiguation*

As mentioned above, the pigeon principle useful in clustering algorithms, grouping the similar, but not identical which the case for to hashed tokens avoiding collision, data point together. However, it has many other practical aspects such as lossless compression for text and image and disambiguation sense for words context, object detection and motion planning. In addition to that it could be used for searching algorithm and resource allocation-similar to google pigeon although the referred naming differences!!!

➢ *Reinforcement Learning (RL) Fine tuning*

Training the model using the corpus of tokens to learn the general features and patterns from the massive data will results in some offset. In the reinforcement learning (RL), this offset feedback error signal is called reward prediction error (RPE). The root means square error (RMSE) apply a powerful predictive tool for measuring the weight of the feedback RPE. Hence, it is used as a metric of the decline regression weight of rewards leading to highly precise outputs. In following sections, such parameter is determined and correlated to the square root floating point methodology with IEEE754 [1] standard and the proper normalization method resulted from paper part-1. [1],[3],[4]

- *RMSE*

In order to fine-tuning the pre-training model (Base model 3.1) of LLM, it is practical to measure the weight of the feedback error signal or what is called reward prediction error (RPE) for reinforcement learning. Such weights need to be underestimate using feedback regression algorithm RMSE.

Mean Square Error (MSE) also called Mean Square Deviation (MSD) define in statistics as the average squared difference between the estimated values and the true values. It is used as an estimator for n values with the following mathematics [1],[9]:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - Y_i^{\wedge})^2) \qquad (2)$$

For matrix representation

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(e_i)^2 = \frac{1}{n}\boldsymbol{e}^{\boldsymbol{T}}\,\boldsymbol{e} \qquad (3)$$

Where $\boldsymbol{e}$ is n×1 vector and T represent the transpose.

From above, the MSE could be used as a predictor to extend the model for further q data points that have been newly obtained in a process known as cross-validation:

$$MSE = \frac{1}{q}\sum_{i=n+1}^{n+q}(Y_i - Y_i^{\wedge})^2 \qquad (4)$$

- *MSE as an Estimator*

Having an estimator $\theta^{\wedge}$ then with respect to an unknown θ is:

$$MSE\ (\theta) = E[(\theta^{\wedge} - \theta)^2] \qquad (5)$$

Expanding the square brackets as follow:

$$MSE\ (\theta) = E\left[\left(\theta^{\wedge 2} - \theta^2 - 2\theta^{\wedge 2}\theta\right)^2\right]$$
$$= E\left[\left(\theta^{\wedge 2} - \theta^2 - 2\theta^{\wedge 2}\theta\right)\ \right]$$
$$= Var_{\theta}\left(\theta^{\wedge 2}\right) + Bias_{\theta}^{2}(\theta^{\wedge},\theta) \qquad (6)$$

- *RMSE function*

From the above sections root mean square error (RMSE) or (RMSD) is defined as follow:

$$RMSE\ (\theta^{\wedge}) = \sqrt{MSE\ (\theta^{\wedge}) =} \sqrt{E[(\theta^{\wedge} - \theta)^2]} \qquad (7)$$

This could be applied for samples as follow:

$$RMSE\ = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(X_i - x_0\ )^2}$$

Applying above with time t continuous variable or regression's dependent variable over period of time T:

$$RMSE \ = \frac{\sqrt{\Sigma_{t=1}^{T}\left(X_i - x_0\ \right)^2}}{T} \qquad (8)$$

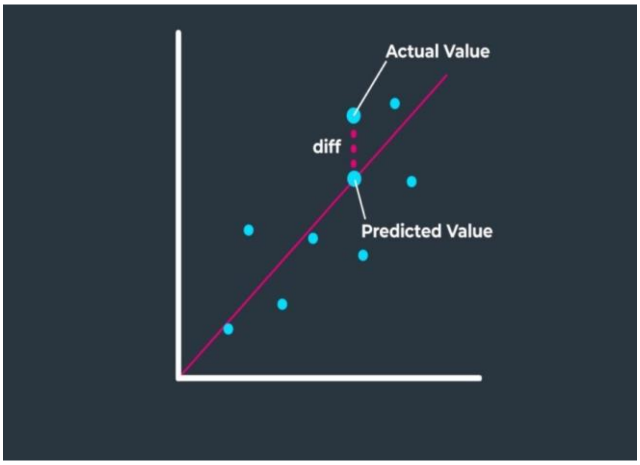


Fig 6 Root Mean Square Error Estimator (RMSE).[11]

- *RMSE normalization (RMS Norm)*

In mathematics, normalizing the root mean square error (NRMSE) is written as follow: [1]

$$NRMSE = \frac{RMSE}{y_{max} - y_{min}} \qquad (9)$$

$$= \frac{RMSE}{y^{-}}$$

Where y^{-} is the mean value of y

Percent of RMS is NRMSE expressed as a percentage and is also called ***Coefficient of Variation*** as follow:

$$CV(RMSD) = \frac{RMSD}{y^{-}} \qquad (10)$$

✓ *Notice:*

The resulting RMS Norm could be used to weight the input tokens to be tuned with the neural network (NN) out put as it shown in figure (6) for Deep seek v2 as example.

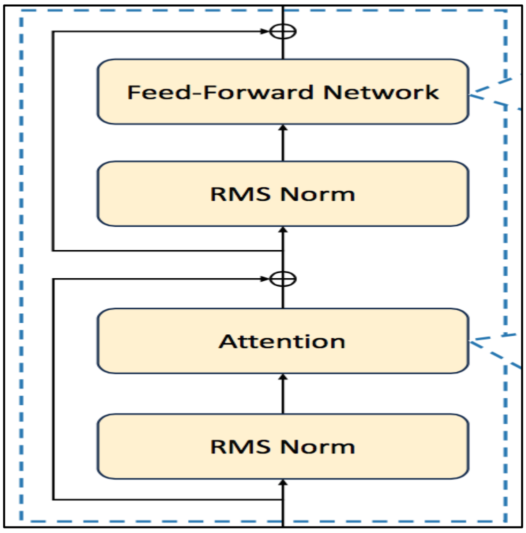


Fig 7 DeepSeek V2 tuned reinforcement learning (RL) block diagram with RMS Norm

- *MSE in MoE model*

MoE refer to mixture of experts is used in LLM that used by transformer models such as that with GPT of many transformers. MSE is used for training the model by performing gradient descent on the MSE loss. The training is done by mixture of experts with time delayed neural network.[3]

- *Matryoshka (MRL) & RMSE gradient*

Matryoshka representation learning (MRL) is nested structure-similar to the nested dolls- that is used as technique of embedding models for flexible computation resources and task requirements. While Matryoshka policy gradient is kind of entropy regularization that is used for the agent model which out of our focus in this paper, RMSE is used in combination with such structure as measure trading off between efficiency and accuracy.[6]

IV. CONCLUSION

In conclusion, through this paper and referring to the paper-part1, the resulting codec schematic, IEEE754 floating point normalization methodology and related RMSE were evaluated for the practical applications of the generative pre-trained GPT model of the coming wave of the AI revolution.

This paper was mainly split in to two parts that plays significant and growing role in generative models which are the base model and the reinforcement learning. The base model, as it's named, represents the foundational structure for pretraining the collected massive data (language, image, video, codes, math, etc). All these data are sampled and tokenized using different codecs and algorithms. This paper analyzed how could using skewing adaptive sliding window for tokenizing different type of data such as image patch and text to combine them into unified representation space. In addition, video could be framed with scalable frame rate (FPS) benefitting from the temporal sliding schematic. Such tokenization has an added value of lossless compression. As a result, the pigeon whole principle offers a practical way of clustering array. Such methodology minimizes the speed and the cash space for arrays searching process

For the reinforcement learning part, the regression feedback plays a critical role to refine and align the base model with the desired outcomes. Root mean square root (RMSE) parameter that could be fastly measured using the method of float point bit iteration IEEE754 [1] rather than regular division could lead to innovative fine tuning.

above process Leads state of art structure of nested matryoshka with pigeon clustering which could result in distilled small models with fast reasoning capability.

REFERENCES

[1] Ahmed ALfahdi, "Modular Square one-way function and square root algorithm" , June 2024

[2] DeepSeek-AI " DeepSeek-V2: A Strong, Economical, and Efficient Mixture-of-Experts Language Model, June 2024

[3] DeepSeek-AI “Incentivizing Reasoning Capability in LLMs via Reinforcement Learning” ,Jan 2025

[4] Qwen Team, Alibaba Group “Qwen 2.5-VL Technical Report” Qwen, China ,Feb 2025

[5] Google DeepMind “Decoding-based Regression” Song, Bahri :Equal cont , 31 Jan 2025.

[6] Google DeepMind “Matryoshka Quantization” Nair, Datta, Dean,Jian,Kusupati :Equal cont, Feb 2025.

[7] Stanford University”The pigeonhole principle lecures 7&8” https//webs.stanford.edu/class/archive/

[8] UC Davis Math “The Riemann integral “ https: // www.math.ucdavis.edu & Riemann Sum and Riemann Integral Explained ,Jan 2020.

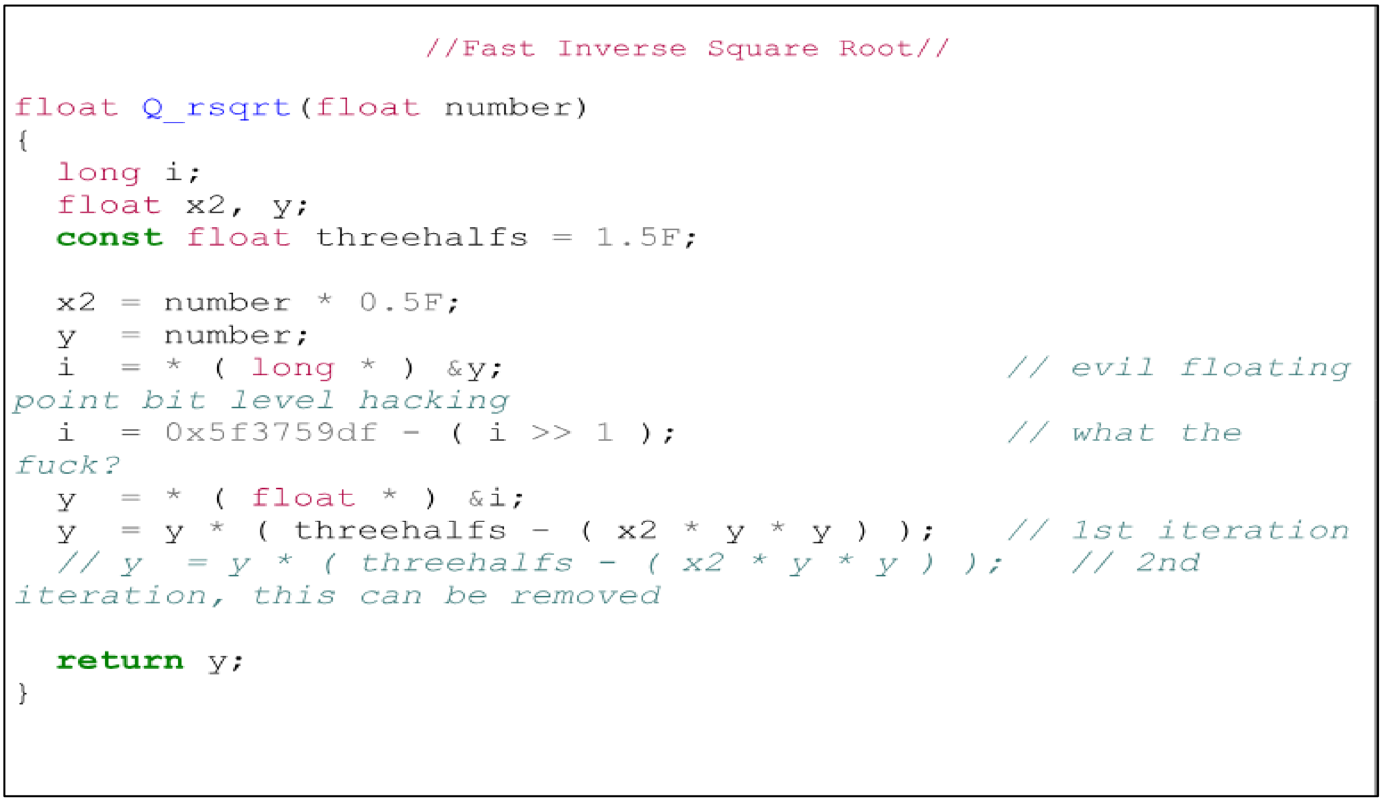
[9] Chai ,Draxler “Root mean square error (RMSE) or mean absolute error (MAE) argument a gainst avoiding RMSE in the literature” May 2014.

[10] Andrei Seymour-Howell,Fast inverse square-root program,2021

[11] Olumide “Root Mean Square Error (RMSE) in AI: What You Need To Know” August 2023.

APPENDIX:

[1] Fast Inverse Square Root [10]



```
                    //Fast Inverse Square Root//

float Q_rsqrt(float number)
{
  long i;
  float x2, y;
  const float threehalfs = 1.5F;

  x2 = number * 0.5F;
  y  = number;
  i  = * ( long * ) &y;                          // evil floating
point bit level hacking
  i  = 0x5f3759df - ( i >> 1 );                  // what the
fuck?
  y  = * ( float * ) &i;
  y  = y * ( threehalfs - ( x2 * y * y ) );      // 1st iteration
  // y  = y * ( threehalfs - ( x2 * y * y ) );   // 2nd
iteration, this can be removed

  return y;
}
```

Fig 8 Fast Inverse Square Root

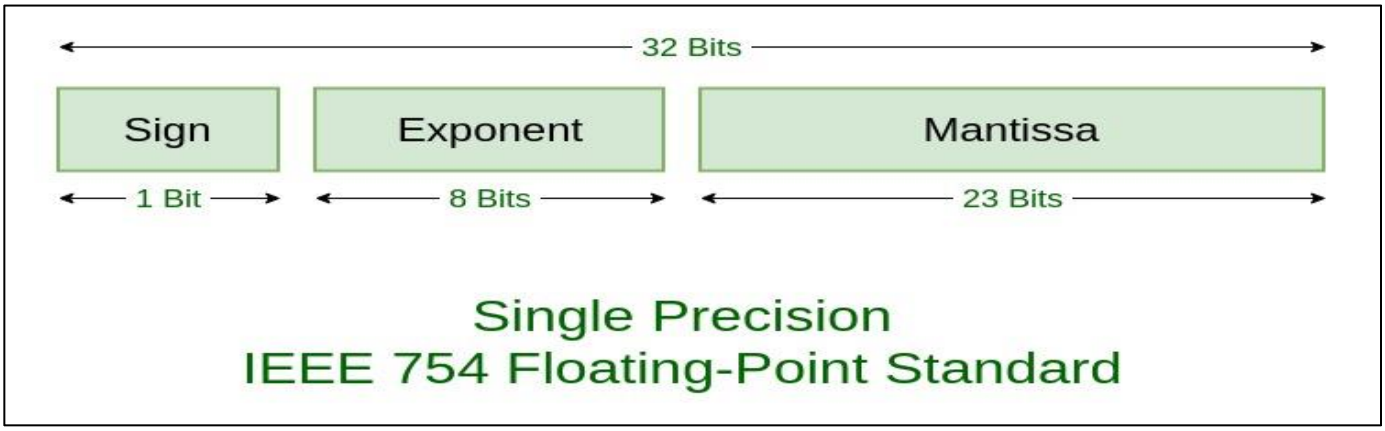


Fig 9 Single Precision IEEE 754 Floating-Point Standard

Quote…

“If you couldn’t can’t explain it simply, you don’t understand it well enough “Albert Einstein…