ROFORMER: ENHANCED TRANSFORMER WITH ROTARY POSITION EMBEDDING

Xi Liang

Biostatistics & Bioinformatics Department

November 1, 2023

Study Aim and Multi-label Classification Approach

- Predict the probability of future diagnoses using past Electronic Health Records (EHR).
- **Methodology:**
  - Approaching the task as a multi-label classification problem.
  - Predicting probabilities for each disease simultaneously.
- **Advantage:**
  - A single predictive model scales across a range of diseases.
  - Eliminates the need for individual models per disease.

EHR Data Structure in BEHRT Study

- **Patient Medical History:** Consists of a sequence of visits to GPs and hospitals.
- **Visit:** Each visit may include diagnoses, medications, measurements, etc. We only consider diagnosis.
- **Representation of EHR:** Denoted as $V_{pp} = \{\text{CLS}, v_{1p}, \text{SEP}, v_{2p}, \text{SEP}, \ldots, v_{np}, \text{SEP}\}$.
  - n_p: Number of visits in patient p's EHR.
  - v_{jp}: Contains the diagnoses in the jth visit, viewed as a list of m_j diagnoses (i.e., $v_{jp} = \{d_{1p}, \ldots, d_{mpj}\}$).
  - *CLS* : Start of medical history.
  - *SEP* : Space between visits.

Patient Profile and Embeddings in EHR Analysis

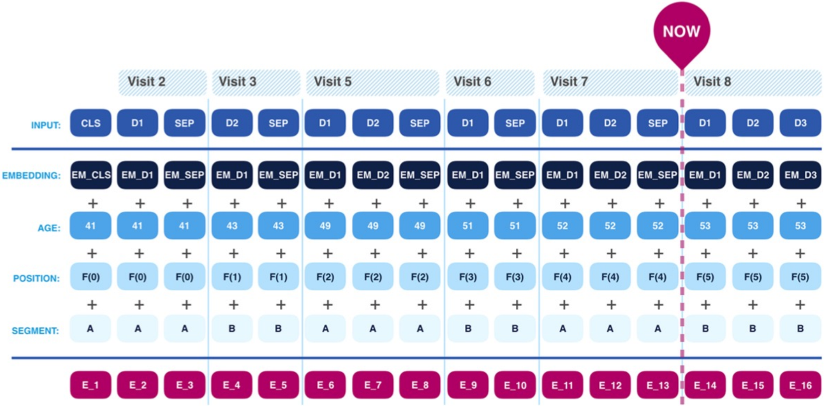
Patient Profile:

- Patients have a sequence of visits, each associated with medical concepts.

Four Key Embeddings:

1. **Events Embeddings:** Pretrained embeddings, similar to word2vec, for medical events.
2. **Age Embedding:** Provides a sense of time, offering universal epidemiological context.
3. **Positional Embeddings:** Determine relative positions of concepts in EHR, capturing disease interactions.
4. **Segment Embeddings:** Uses ’A’ or ’B’ symbols for encounter separation, ensuring order-invariance for intra-visit concepts.

Embeddings overview



NOW

		Visit 2		Visit 3		Visit 5			Visit 6		Visit 7			Visit 8		
INPUT:	CLS	D1	SEP	D2	SEP	D1	D2	SEP	D1	SEP	D1	D2	SEP	D1	D2	D3
EMBEDDING:	EM_CLS	EM_D1	EM_SEP	EM_D1	EM_SEP	EM_D1	EM_D2	EM_SEP	EM_D1	EM_SEP	EM_D1	EM_D2	EM_SEP	EM_D1	EM_D2	EM_D3
	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
AGE:	41	41	41	43	43	49	49	49	51	51	52	52	52	53	53	53
	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
POSITION:	F(0)	F(0)	F(0)	F(1)	F(1)	F(2)	F(2)	F(2)	F(3)	F(3)	F(4)	F(4)	F(4)	F(5)	F(5)	F(5)
	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
SEGMENT:	A	A	A	B	B	A	A	A	B	B	A	A	A	B	B	B
	E_1	E_2	E_3	E_4	E_5	E_6	E_7	E_8	E_9	E_10	E_11	E_12	E_13	E_14	E_15	E_16

Age2Vec: A Time Representation Model

Key Properties of Age2Vec:

1. **Periodicity and Non-Periodicity:** Captures both periodic (e.g., seasonal trends) and non-periodic patterns (e.g., age-related events).
2. **Invariance to Time Rescaling:** Effective across different time scales (days, hours, seconds), ensuring model consistency despite time unit changes.

Time2Vec Representation:

- Defines time τ as a vector $t2v(\tau)$ of size $k + 1$.
- Components: $t2v(\tau)[i] = \begin{cases} \omega_i\tau + \phi_i & \text{if } i = 0 \\ F(\omega_i\tau + \phi_i) & \text{if } 1 \leq i \leq k \end{cases}$
- F is a periodic activation function (e.g., sine).
- Captures periodic behavior using parameters ω_i (frequency) and ϕ_i (phase-shift).
- Linear term for non-periodic patterns; demonstrates invariance to time rescaling.

Pre-training BEHRT with Masked Language Model (MLM)

- **Deep Bidirectional Model:**
  - BEHRT pre-trained using MLM, akin to original BERT.
  - Outperforms unidirectional or shallow bidirectional models.
- **Embedding Initialization and Training:**
  - Random initialization of disease, age, and segment embeddings. Positional encoding from pre-determined position encoding.
  - MLM training: 86.5% words unchanged to learn most of the real, unaltered language patterns, 12% to [mask] to learn how to predict these masked words based on the context, 1.5% random to increase the diversity and complexity of the training data, compelling the model to learn not only how to handle masked information but also how to recover correct information from incorrect or irrelevant inputs.
- **Evaluation of MLM Task:**
  - Precision score evaluation (true positive/predicted positive ratio).

Understanding Self-Attention Mechanism in NLP

Overview:

- The self-attention mechanism is pivotal for capturing complex relationships in natural language, applicable to various positions within an input sequence.

Sequences and Embeddings:

- Consider a sequence $SN = \{w_i\}_{i=1}^{N}$ of N input tokens, with corresponding d-dimensional word embeddings $EN = \{x_i\}_{i=1}^{N}$, lacking position information.

Self-Attention Mechanics:

- Position information is incorporated into embeddings, transformed into queries q_m, keys k_n, and values v_n.
- Attention weights $a_{m,n}$ are calculated from queries and keys, producing output as weighted sums of value vectors v_n.

Key Concept:

- Effective position encoding in transformers is crucial, impacting how the model forms the attention mechanism.

Absolute position embedding

Vaswani et al. (2017) introduced a method to generate position vectors p_i using sinusoidal functions in Transformer models. This approach encodes position information using sine and cosine functions.

A typical choice of Equation (1) is:

$$f_{t:t\in\{q,k,v\}}(x_i, i) := W_{t:t\in\{q,k,v\}}(x_i + p_i)$$

The sinusoidal functions are defined as:

$$p_{i,2t} = \sin(k/10000^{2t/d})$$

$$p_{i,2t+1} = \cos(k/10000^{2t/d})$$

Where $p_{i,2t}$ is the $2t^{th}$ element of the d-dimensional vector p_i, k is the position index, and d is the embedding dimension.

Absolute position embedding

- Position Information: Each position has a unique encoding, helping the model understand the position of this token in a sequence. However, it does not show the relative position between two tokens.
- Scalability: The periodic nature of sine and cosine functions allows for effective encoding of long sequences.
- No Additional Training Required: This position encoding is predefined and doesn't require learning through training, simplifying the model's training process. This is also one of its drawbacks; it means that they may not capture some nuances of positional information that a model could potentially learn with trainable parameters.

Relative Position Embedding - Introduction

Initial Approach (Shaw et al. [2018]):

$$f_q(x_m) := W_q x_m$$
$$f_k(x_n, n) := W_k(x_n + \tilde{p}^{r}_{kr})$$
$$f_v(x_n, n) := W_v(x_n + \tilde{p}^{r}_{vr})$$

$\tilde{p}_{kr}, p_{vr} \in \mathbb{R}^d$ are trainable relative position embeddings.Relative distance r is clipped between $r_{\min}$ and $r_{\max}$.

Decomposition of Self-Attention (Dai et al. [2019]):

- Decomposes $q_m^T {}_* K_n$ as
- $x_m^T W_q W_k x_n + x_m^T W_q W_k p_n + p_m^T W_q W_k x_n + p_m^T W_q W_k p_n$.
  Introduces sinusoid-encoded relative positions and independent vectors u and v.

Relative Position Embedding - Further Developments

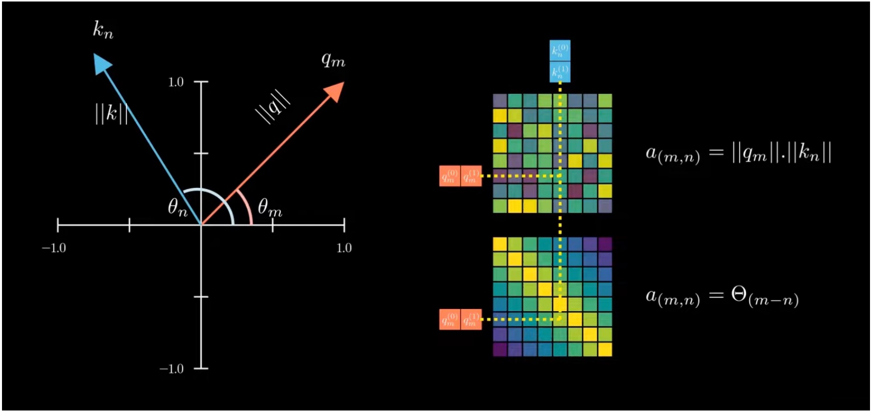
Further Refinements:

- Raffel et al. [2020]: Simplifies to $q_{m} {}_{K n} = x_m^T W_q W_k x_n + b_{i,j}$.
- Ke et al. [2020]: Investigates correlations between absolute positions and words.
- He et al. [2020]: Argues for modeling relative positions using middle two terms of Dai et al.'s formulation.

Comparative Analysis:

- Radford and Narasimhan [2018]: Finds the approach similar to He et al. [2020] most efficient.
- Common goal: Directly add position information to context representations.

Visualize the query and the key



Transformer-based Language Modeling

In order to incorporate relative position information in the embeddings, we require the inner product of query q_m and key k_n to be formulated by a function g, which takes only the word embeddings x_m, x_n, and their relative position $m - n$ as input variables. In other words, we hope that the inner product encodes position information only in the relative form:

$$\langle f_q(x_m, m), f_k(x_n, n)\rangle = g(x_m, x_n, m - n).$$

Rotary Position Embedding

Specifically, incorporating the relative position embedding is straightforward: simply rotate the affine-transformed word embedding vector by amount of angle multiples of its position index and thus interprets the intuition behind Rotary Position Embedding.

- **A 2D case**
  - We begin with a simple case with a dimension $d = 2$. We want $q_m^T k_n$ be a function (g) of only word embeddingsx_m, x_n and their relative position $m - n$:

$$f_q(x_m, m) = (W_q x_m)e^{im\theta}$$
$$f_k(x_n, n) = (W_k x_n)e^{in\theta}$$

  - $g(x_m, x_n, m - n) = \mathrm{Re}[(W_q x_m)(W_k x_n)^* e^{i(m-n)\theta}]$
  - Re[·] is the real part of a complex number
  - $(W_k x_n)^*$ represents the conjugate complex number of $(W_k x_n)$.
  - $\vartheta \in \mathbb{R}$ is a preset non-zero constant.

Rotary Position Embedding (Contd.)

- **A 2D case (Continued)** with Euler's formular:$e^{im\theta} = \cos(m\vartheta) + \sin(m\vartheta)$, the matrix form of $e^{im\theta}$ is

$$\begin{matrix} \cos(m\vartheta) & -\sin(m\vartheta) \\ \sin(m\vartheta) & \cos(m\vartheta) \end{matrix}$$

- We can further write $f_{q,k}$ in a multiplication matrix:

$$\begin{bmatrix} \cos(m\vartheta) & -\sin(m\vartheta) \\ \sin(m\vartheta) & \cos(m\vartheta) \end{bmatrix} \begin{bmatrix} W_{q,k}^{(1,1)} & W_{q,k}^{(1,2)} \\ W_{q,k}^{(2,1)} & W_{q,k}^{(2,2)} \end{bmatrix} \begin{bmatrix} x_m^{(1)} \\ x_n^{(2)} \end{bmatrix}$$

where (x_m, y_m) is x_m expressed in the 2D coordinates.

General Form

In order to generalize our results in 2D to any $x_i \in \mathbb{R}^d$ where d is even, we divide the d-dimension space into $\frac{d}{2}$ sub-spaces and combine them in the merit of the linearity of the inner product, turning $f_{q,k}$ into:

$$f_{q,k}(x_m, m) = R_{d\Theta,m} W_{q,k} x_m$$

where

$$R_{d\Theta,m} = \begin{bmatrix} \cos(m\vartheta_1) & -\sin(m\vartheta_1) & 0 & \cdots & 0 & 0 \\ \sin(m\vartheta_1) & \cos(m\vartheta_1) & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cos(m\vartheta_2) & -\sin(m\vartheta_2) & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & . \\ 0 & 0 & \cdots & 0 & \sin(m\vartheta_{\frac{d}{2}}) & \cos(m\vartheta_{\frac{d}{2}}) \end{bmatrix}$$

with $\Theta = \{\vartheta_i = 10000^{-2(i-1)/d}, \ i \in [1, 2, \dots \ ]\}$

$$\boldsymbol{q}_m^\intercal \boldsymbol{k}_n = (\boldsymbol{R}^d_{\Theta,m} \boldsymbol{W}_q \boldsymbol{x}_m)^\intercal (\boldsymbol{R}^d_{\Theta,n} \boldsymbol{W}_k \boldsymbol{x}_n) = \boldsymbol{x}^\intercal \boldsymbol{W}_q R^d_{\Theta,n-m} \boldsymbol{W}_k \boldsymbol{x}_n$$

Visualization of RoPE2

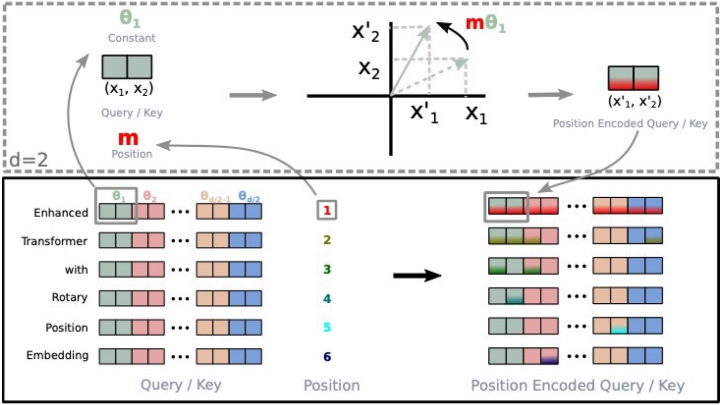


Figure 1: Implementation of Rotary Position Embedding(RoPE).

Figure: Detailed view of the RoPE2 model.

Absolute to Rotary

Starting from absolute embeddings in the standard transformer:

$$q_m^T k_n = [W_q(x_m + PE(m))]^T W_k(x_n + PE(n))$$
$$q_m^T k_n = [W_q x_m]^T W_k x_n$$

We removed the absolute positional encoding and then multiplied by R. In this way, we apply RoPE to self attention

$$\begin{aligned} q_m^T k_n &= [R_{d\Theta,m} W_{q,k}]^T R_{d\Theta,n} W_{q,k} \\ &= W_{q,k}^T x_n^T [R_{d\Theta,m} R_{d\Theta,n}] W_{q,k} x_n \\ &= W_{q,k}^T x_n^T [R_{d\Theta,m-n}] W_{q,k} x_n \end{aligned}$$

3.3 Properties of RoPE

Long-term Decay:

- Setting $\vartheta_i = 10000^{-2i/d}$ offers a long-term decay property, meaning the inner-product decays as relative position increases. This aligns with the intuition that distant tokens connect less.

RoPE with Linear Attention:

- Incorporates RoPE by multiplying rotation matrix with non-negative function outputs, enhancing linear attention models.
- Maintains the computation efficiency while embedding positional information.

Efficiency and Effectiveness:

- RoPE integrates seamlessly with linear attention, preserving the norm of hidden representations.
- Allows efficient computation without losing the capacity to model the importance of values in attention.

Relative distance

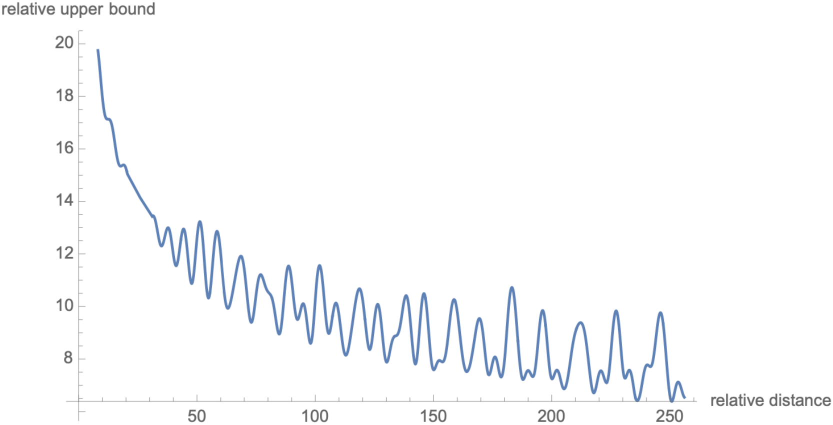


Figure 2: Long-term decay of RoPE.

Experiments and Evaluation - Overview

Overview:

- Evaluation of RoFormer across various NLP tasks.
- Experiments include machine translation, pre-training language modeling, fine-tuning on GLUE tasks, and evaluation with linear attention and on Chinese data.
- All experiments conducted on cloud servers with 4 x V100 GPUs.

4.1 Machine Translation:

- Dataset: WMT 2014 English-German with 4.5 million sentence pairs.
- Comparison with Transformer-base (Vaswani et al., 2017).
- RoFormer shows improved BLEU scores: Transformer-base (27.3), RoFormer (27.5).
- Implementation includes modifications to self-attention layer and use of BPE with 37k vocabulary.

Experiments and Evaluation - Continuation

4.2 Pre-training Language Modeling:

- Replacement of BERT’s sinusoidal position encoding with RoPE.
- Training on BookCorpus and Wikipedia Corpus.
- RoFormer shows faster convergence than BERT.

4.3 Fine-tuning on GLUE tasks:

- RoFormer fine-tuned on several GLUE datasets.
- Significant improvements in three out of six datasets.

4.5 Evaluation on Chinese Data:

- Modifications to WoBERT for Chinese data.
- Experiments on long documents exceeding 512 characters.
- RoFormer performs better with increased input text length.

Experiments and Evaluation

Table 1: The proposed RoFormer gives better BLEU scores compared to its baseline alternative Vaswani et al. [2017] on the WMT 2014 English-to-German translation taskBojar et al. [2014].

Model	BLEU
Transformer-baseVaswani et al. [2017]	27.3
RoFormer	**27.5**

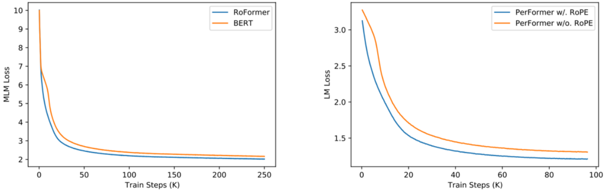


Figure 3: Evaluation of RoPE in language modeling pre-training. **Left**: training loss for BERT and RoFormer. **Right**: training loss for PerFormer with and without RoPE.

Conclusion: RoPE Model

- **Innovative Encoding**: RoPE introduces a novel way of incorporating positional information into the self-attention mechanism, enhancing the model's ability to understand sequential data.
- **Improved Performance**: Demonstrates superior performance in various NLP tasks, especially those requiring nuanced understanding of sequence and context.
- **Efficient and Scalable**: Offers a more computationally efficient approach compared to traditional positional encoding methods, making it suitable for larger datasets and models.
- **Versatile Application**: Can be integrated seamlessly with existing Transformer architectures, broadening its applicability across different domains.