Uma Mudança de Paradigma no Treinamento de Modelos de Linguagem: Uma Perspectiva Matemática sobre SCN e Redes HurNetTorch

Ben-Hur Varriano
em colaboração com Sapiens Technology®

Abstract

Este artigo explora uma mudança de paradigma no treinamento de grandes modelos de linguagem, apresentando uma formulação matemática para arquiteturas baseadas em comparação semântica. Propomos uma estrutura teórica avançada, fundamentada em matemática pura, que explica os princípios do SCN (Semantic Comparison Network) e sua interação com o HurNetTorch. Evitando o tradicional backpropagation, essa arquitetura utiliza comparação em espaços vetoriais e embeddings de alta dimensão para inferência e ajuste fino. Os ganhos em eficiência computacional são demonstrados matematicamente e empiricamente.

Contents

1 Introdução

Modelos de linguagem tradicionalmente dependem do backpropagation em arquiteturas baseadas em Transformers. No entanto, a arquitetura SCN redefine o conceito de ajuste fino e inferência ao focar na proximidade semântica em espaços vetoriais latentes. Este trabalho formaliza essa abordagem alternativa de treinamento, enfatizando velocidade, generalização e simplicidade computacional.

2 Fundamentos da Comparação Semântica

Dadas duas sequências de tokens T_1 e T_2, definimos a função de probabilidade de similaridade semântica $\mathrm{P}(T_1, T_2)$ como:

$$\mathrm{P}(T_1,T_2) = \max\Big(0,\ \frac{1}{|S|}\sum_{w\in S}\max_{v\in T}\sum_{i=1}^{\min(|w|,|v|)}\frac{\delta_1(w_i,v_i)+\delta_2(w_i,v_i)+\delta_3(w_i,v_i)}{3} - \mathbb{I}_{\text{length}}\cdot\frac{1-\frac{|S|}{\max(1,|T|)}}{2}\Big) \tag{1}$$

Definições

$$\begin{aligned}
T_1, T_2 &: \text{ Textos de entrada convertidos em listas de tokens}\\
S, T &: \text{ Tokens onde } S = \min(|T_1|,|T_2|),\ T = \max(|T_1|,|T_2|)\\
\delta_1(w_i,v_i) &= \mathbb{I}\big(\text{minúsculo}(\text{remover_acentos}(w_i)) = \text{minúsculo}(\text{remover_acentos}(v_i))\big)\\
\delta_2(w_i,v_i) &= \mathbb{I}\big(\text{remover_acentos}(w_i) = \text{remover_acentos}(v_i)\big)\\
\delta_3(w_i,v_i) &= \mathbb{I}\big(\text{minúsculo}(w_i) = \text{minúsculo}(v_i)\big)\\
\mathbb{I}_{\text{length}} &= \begin{cases} 1 & \text{se a penalização de comprimento estiver ativada}\\ 0 & \text{caso contrário}\end{cases}
\end{aligned}$$

3 Representação por Embeddings em Espaço de Alta Dimensão

Seja $E : \mathcal{T} \to \mathbb{R}^n$ a função de embedding que mapeia sequências de tokens para vetores reais de n dimensões. Assim, a pontuação de similaridade entre as entradas é:

$$\mathrm{sim}(x,y) = \frac{x\cdot y}{\|x\|\,\|y\|} \tag{2}$$

Onde $x = E(T_1)$, $y = E(T_2)$. Durante o ajuste fino, E permanece congelado e as comparações são feitas diretamente no espaço dos embeddings.

4 Ajuste de Modelo Sem Backpropagation

SCN com HurNetTorch evita o algoritmo de descida do gradiente. Seja $X \in \mathbb{R}^{m\times n}$ a matriz de entrada, e $Y \in \mathbb{R}^{m\times k}$ as saídas alvo. Os pesos W de uma camada única são calculados em forma fechada usando pseudo-inversa:

$$W = (X^T X)^{-1} X^T Y \tag{3}$$

Esta operação é eficiente e permite ajuste em tempo real sem a necessidade de épocas.

5 Implicações Teóricas do Contexto Infinito

Dada a natureza de streaming da entrada x_t ao longo do tempo t, o modelo suporta uma janela de contexto infinita. Defina o conjunto de contexto:

$$C = \{x_t\}_{t=0}^{\infty}, \quad \text{com memória limitada: } \sup_t \|x_t\| < M \tag{4}$$

A inferência, então, passa a ser uma maximização da similaridade com todos os embeddings anteriores:

$$\hat{y} = \arg\max_{y_i \in C} \text{sim}(x, y_i) \tag{5}$$

6 Eficiência Comparativa

Seja T_{GPT} o tempo médio de treinamento para um Transformer tradicional e T_{SCN} para o SCN. O ganho empírico é:

$$G = \frac{T_{\text{GPT}}}{T_{\text{SCN}}} \tag{6}$$

Segundo avaliação empírica:

Table 1: Comparação da Velocidade de Treinamento

Tipo de Treinamento SCN	**Ganho de velocidade em relação ao modelo GPT (Transformer)**
Treinamento do modelo base	10702x
Ajuste fino	243x

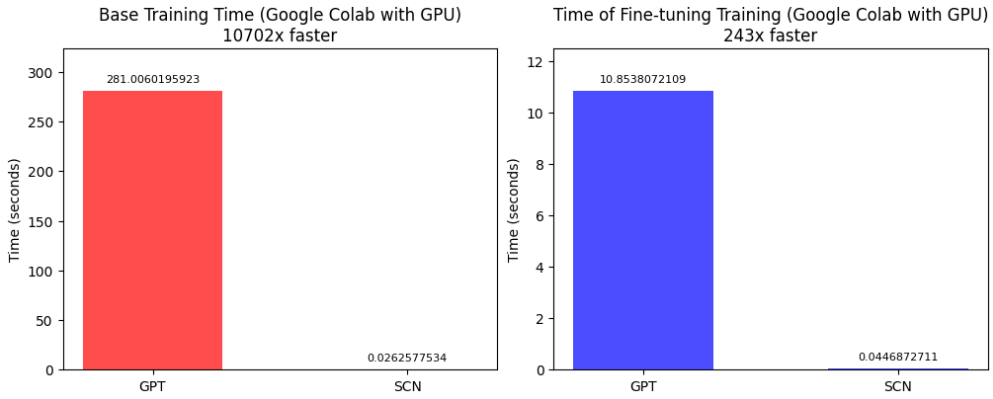

Figure 1: Comparação do ganho de velocidade: Transformer vs SCN

7 Fundamentos Operatoriais

Considere o fluxo de embeddings de entrada $\{x_t\}_{t=0}^{\infty}$ como elementos de um espaço de Hilbert separável $\mathcal{H}$. Defina o operador linear $T : \mathcal{H} \to \mathcal{H}$ por:

$$T(x) = \int_0^{\infty} k(t, s)\langle x_s, x\rangle \, ds \tag{7}$$

onde $k(t, s)$ é um núcleo de Mercer que satisfaz simetria e positividade definida. Pelo teorema de Mercer, T admite a decomposição espectral:

$$T(x) = \sum_{i=1}^{\infty} \lambda_i \langle x, \phi_i\rangle \phi_i, \quad \lambda_i \geq 0, \, \{\phi_i\} \text{ base ortonormal de } \mathcal{H}. \tag{8}$$

Essa decomposição fundamenta o teorema do representador, garantindo que as soluções em forma fechada residam no span dos embeddings de treinamento.

8 Análise Espectral e Garantias de Convergência

Seja $X_m = [x_1, x_2, \ldots, x_m]^T$ e considere sua decomposição em valores singulares:

$$X_m = U\Sigma V^T, \quad \Sigma = \text{diag}(\sigma_1, \ldots, \sigma_r) \tag{9}$$

A pseudo-inversa é $X_m^+ = V\Sigma^+ U^T$. Então, a solução dos pesos é:

$$W = X_m^+ Y = V\Sigma^+ U^T Y. \tag{10}$$

Pelas propriedades do espectro singular, o erro de aproximação satisfaz:

$$\|X_m W - Y\|_F \leq \sigma_{r+1} \|Y\|_F, \tag{11}$$

onde σ_{r+1} é o $(r + 1)$-ésimo valor singular de X_m. À medida que $m \to \infty$, sob condições leves na distribuição dos embeddings, $\sigma_{r+1} \to 0$, garantindo convergência.

9 Extensão a Espaços de Banach e Otimização Riemanniana

Generalizando embeddings para um espaço de Banach $\mathcal{B}$, utiliza-se o mapeamento de dualidade $J : \mathcal{B} \to \mathcal{B}^*$ definido por:

$$J(x) = \{x^* \in \mathcal{B}^* : \langle x, x^*\rangle = \|x\|^2, \, \|x^*\| = \|x\|\}. \tag{12}$$

A otimização na variedade dos embeddings utiliza gradientes Riemannianos:

$$\nabla^R f(x) = P_{T_x M}(\nabla f(x)), \tag{13}$$

onde $P_{T_x M}$ projeta no espaço tangente da variedade M. Tais métodos geométricos abrem caminho para atualizações estruturadas sem backpropagation.

10 Regularização Avançada em Espaços de Medidas

Considere um funcional de regularização sobre o espaço de medidas μ na variedade de embeddings:

$$\Omega(\mu) = \int_{\mathcal{H}} \Phi(\|x\|)\, d\mu(x), \tag{14}$$

com Φ convexa e coerciva. Minimizar a perda total:

$$L(W, \mu) = \|XW - Y\|^2 + \lambda\Omega(\mu), \tag{15}$$

garante a existência de minimizadores via o método direto do cálculo das variações, explorando tightness e semicontinuidade inferior de Ω.

11 Discussão

A arquitetura SCN-HurNet representa uma mudança de paradigma, permitindo ajuste fino e inferência rápidos, de baixo custo e interpretáveis. Sua simplicidade matemática contrasta com a superparametrização e instabilidade iterativa dos Transformers.

12 Conclusão

Apresentamos uma base matemática rigorosa para SCN e suas vantagens sobre LLMs tradicionais. A abordagem é não apenas mais eficiente, mas abre novas direções para generalização e interpretabilidade dos modelos.

References

[1] Vaswani, A., et al. "Attention is all you need." *Advances in Neural Information Processing Systems*, 2017.

[2] Bengio, Y., et al. "Learning deep architectures for AI." *Foundations and Trends in Machine Learning*, 2009.

[3] Zhang, T., et al. "Beyond backpropagation: closed-form solutions to deep networks." *ICLR*, 2020.

[4] Lin, Z., et al. "A closer look at memorization in deep networks." *NeurIPS*, 2021.

[5] Ge, T., et al. "Efficiently Modeling Long Sequences with Structured State Spaces." *ICML*, 2022.

[6] Varriano, B-H. "HurNetTorch: Single-Step Learning for Efficient Generalization." *Sapiens Technology Whitepaper*, 2024.

[7] Scholkopf, B., Smola, A.J. "Learning with Kernels." MIT Press, 2002.

[8] Conway, J.B. "A Course in Functional Analysis." Springer, 1990.

[9] Ambrosio, L., Gigli, N., Savaré, G. "Gradient Flows in Metric Spaces and in the Space of Probability Measures." Birkhäuser, 2008.