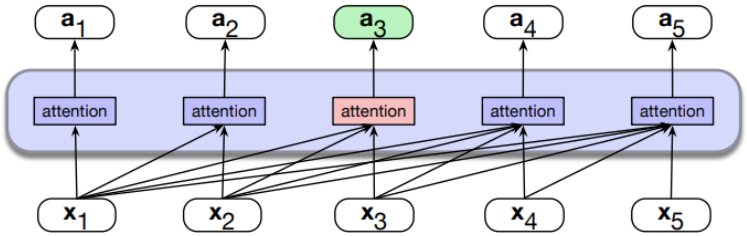
Masked Language Models

Professor Thiago de Paulo Faleiros

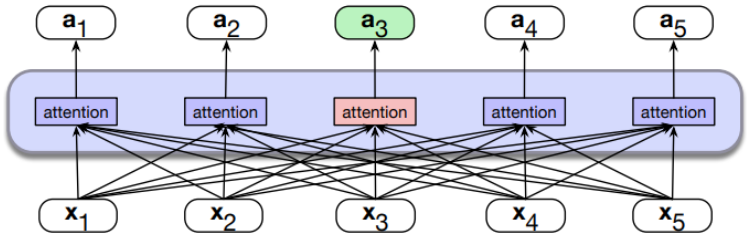
Masked Language Modeling

- Até agora vimos modelos autoregressivos (causal, left-to-right) LMs.
- Mas e quanto às tarefas que queremos poucos tokens futuros?
    - Especialmente para tarefas onde mapeamos cada token de entrada para um token de saída.
- Encoders Bidirecionais usam Masked self-attention para
    - Mapear sequência de embeddings de entrada $(x_1, \ldots, x_n)$
    - Para sequências de embeddings de saída de mesmo tamanho $(h_1, \ldots, h_n)$
    - Onde o vetor de saída foi contextualizado usando informações de toda sequência de entrada.

Bidirectional Self-Attention



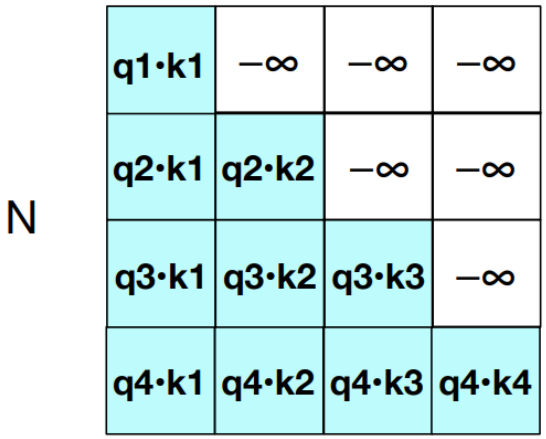
a) A causal self-attention layer



b) A bidirectional self-attention layer

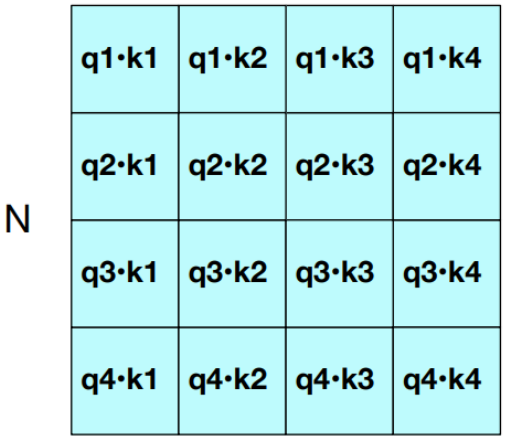
Fácil! Apenas remova a máscara

Casual self-attention



$$\textbf{head} = \text{softmax}\left(\text{mask}\left(\frac{\mathbf{QK}^\intercal}{\sqrt{d_k}}\right)\right)\mathbf{V}$$

Bidirectional self-attention



$$\textbf{head} = \text{softmax}\left(\frac{\mathbf{QK}^\intercal}{\sqrt{d_k}}\right)\mathbf{V}$$

BERT: Bidirectional Encoder Representation from Transformers

- BERT (Devlin et al., 2019)
  - 30.000 tokens apenas em inglês (WordPiece tokenizer)
  - Janela de contexto de entrada $N = 512$ tokens, em dimensionalidade $d = 768$
  - $L = 12$ camadas de blocos de atenção, cada com $A = 12$ camadas de atenção com múltiplas cabeças.
  - O modelo resultante tem aproximadamente 100M parâmetros
- XLM-RoBERTa (Conneau et al., 2020)
  - 250000 tokens multilínguas (SentencePiece Unigram LM tokenizer)
  - Janela de contexto de entrada $N = 512$ tokens, em dimensionalidade $d = 1024$
  - $L = 12$ camadas de blocos de atenção, com $A = 16$ camadas de atenção com múltiplas cabeças.
  - O modelo resultante tem aproximadamente 550M parâmetros

Intuição do treinamento do BERT

- Para LMs da esquerda-para-direita, o modelo tenta predizer a última palavra a partir das palavras anteriores.

  The water of Walden Pond is so beautifully _____

  - O treinamento tenta melhorar essa predição da última palavra
- Para bidirecional masked LM, o modelo tenta predizer uma ou mais palavras considerando todo o restante de palavras

  The _____ of Walden Pond _____ so beautifully blue

  - O modelo gera a distribuição de probabilidade sobre o vocabulário para cada token faltante
  - Usamos a entropia cruzada para cada predição do modelo controlando o processo de aprendizagem.

MLM training in BERT

15% dos tokens são aleatoriamente escolhidos para serem mascarados
Exemplo: “Lunch was delicious”, se a palavra “delicious” foi aleatoriamente escolhida:
Três possibilidades:

- 80% : Tokens é substituído pelo token especial [MASK]

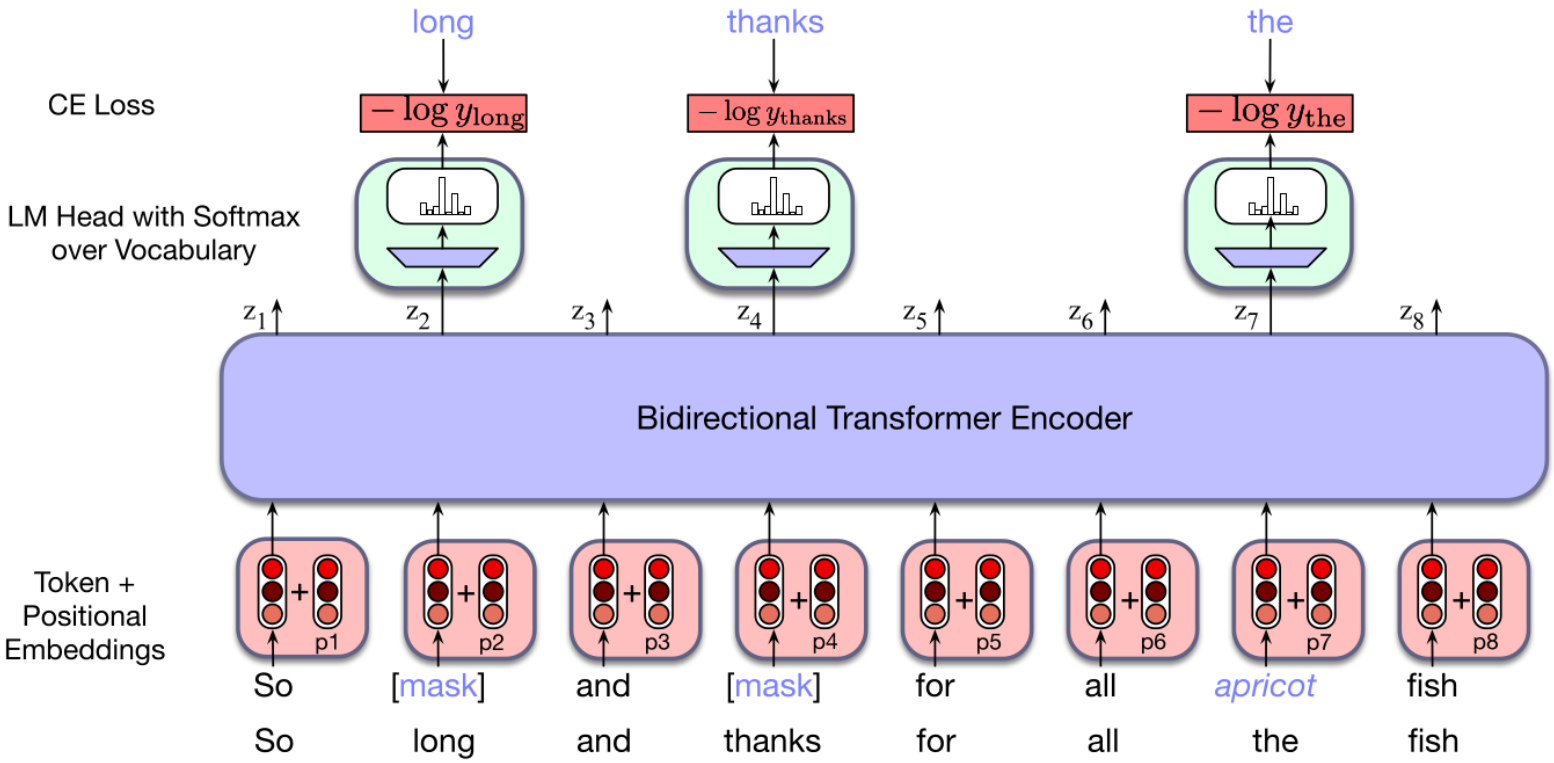
  Lunch was delicious –> Lunch was [MASK]

- 10% : Token é substituído por um token aleatório (amostrado por uma distribuição de unigramas)

  Lunch was delicious –> Lunch was gasp

- 10% : Token não é alterado

  Lunch was delicious –> Lunch was delicious


long
thanks
the
CE Loss
$-\log y_{\text{long}}$
$-\log y_{\text{thanks}}$
$-\log y_{\text{the}}$
LM Head with Softmax over Vocabulary
z_1 z_2 z_3 z_4 z_5 z_6 z_7 z_8
Bidirectional Transformer Encoder
Token + Positional Embeddings
p1 p2 p3 p4 p5 p6 p7 p8
So [mask] and [mask] for all apricot fish
So long and thanks for all the fish

MLM loss

O Modelo de Língua tem como saída a camada final do transformer L, multiplica isso pela camada de "unembedding" e cria as probabilidades

$$u_i = h_i^L E^T$$
$$y_i = softmax(u_i)$$

Para o x_i correspondente a "long", o LOSS é a probabilidade da palavra correta "long", dada a saída h_i^L):

$$L_{MLM}(x_i) = -\log P(x_i|h_i^L)$$

Obtemos os gradientes calculando a média dessa perda no batch

$$L_{MLM} = -\frac{1}{|M|}\sum_{i \in M} \log P(x_i|h_i^L)$$

Predição da próxima sentença

Dadas duas sentenças o modelo prediz se os pares de sentenças são adjacentes (do corpus de treinamento) ou pares não relacionados. BERT introduz dois caracteres especiais

CLS é adicionado no começo da do par de sentenças

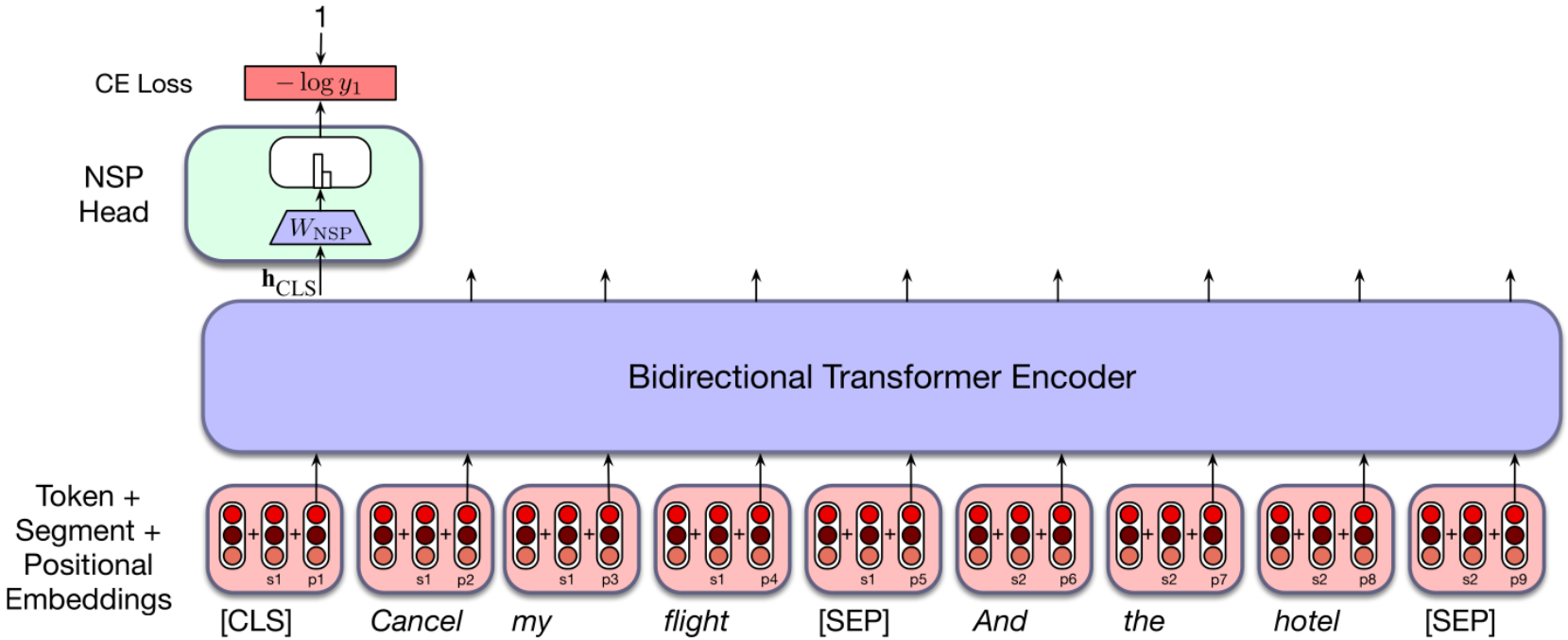
SEP é adicionado entre as sentenças e no final da segunda sentença

E mais dois tokens especiais

- (1^a sentença) e (2^a sentença)
- Esses embeddings são adicionados ao embeding de entrada e ao embedding posicional

h^L_{CLS} da camada final do token [CLS] é a entrada da cabeça de classificação (pesos de W_{NSP} que prediz duas classes:

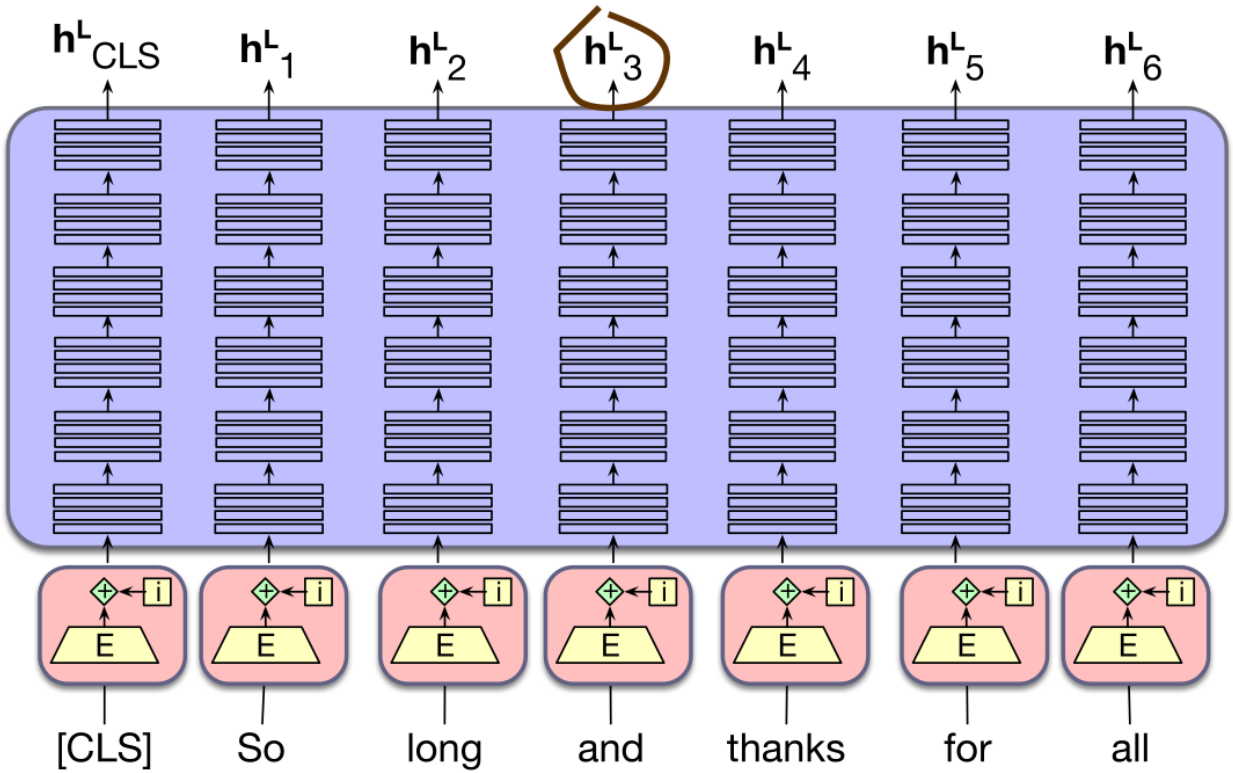
$$y_i = softmax(h^L_{CLS} W_{NSP})$$


1
CE Loss
$-\log y_1$
NSP
Head
W_{NSP}
$\mathbf{h}_{\text{CLS}}$
Bidirectional Transformer Encoder
Token +
Segment +
Positional
Embeddings
s1 p1
s1 p2
s1 p3
s1 p4
s1 p5
s2 p6
s2 p7
s2 p8
s2 p9
[CLS]
Cancel
my
flight
[SEP]
And
the
hotel
[SEP]

Mais detalhes

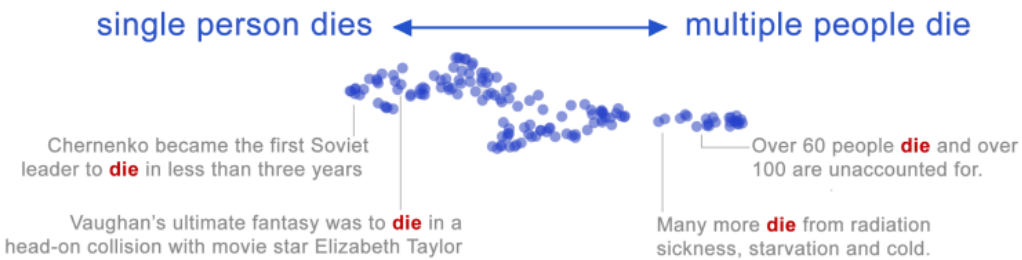
- O modelo original foi treinado com 40 passos sobre o conjunto de treinamento
- Alguns modelos (com o RoBERTa) remove o NSP Loss
- O tokenizador para modelos multilínguas é treinado para amostras estratificadas de línguas.
- Modelos Multilínguas são melhores que modelos monolíngua com pequeno número de linguagens.
  - Com grandes números de linguagens, modelos monolínguas naquela lingua pode ser melhor
  - A "maldição da multilinguagem"

Contextual Embeddings para representar palavras



Embeddings estáticos vs contextual

Embedding estáticos representam os “tipos das palavras” (entradas em dicionários)
Embedding de contexto representam instâncias das palavras (uma para cada vez que a palavra ocorre em qualquer contexto/sentença)



Word Sense

Palavras são ambíguas

O sentido das palavras é uma representação discreta do significado de uma aspecto

mouse.^{1} : ... a mouse controlling a computer system in 1968.

mouse.^{2} : ... a quiet animal like a mouse

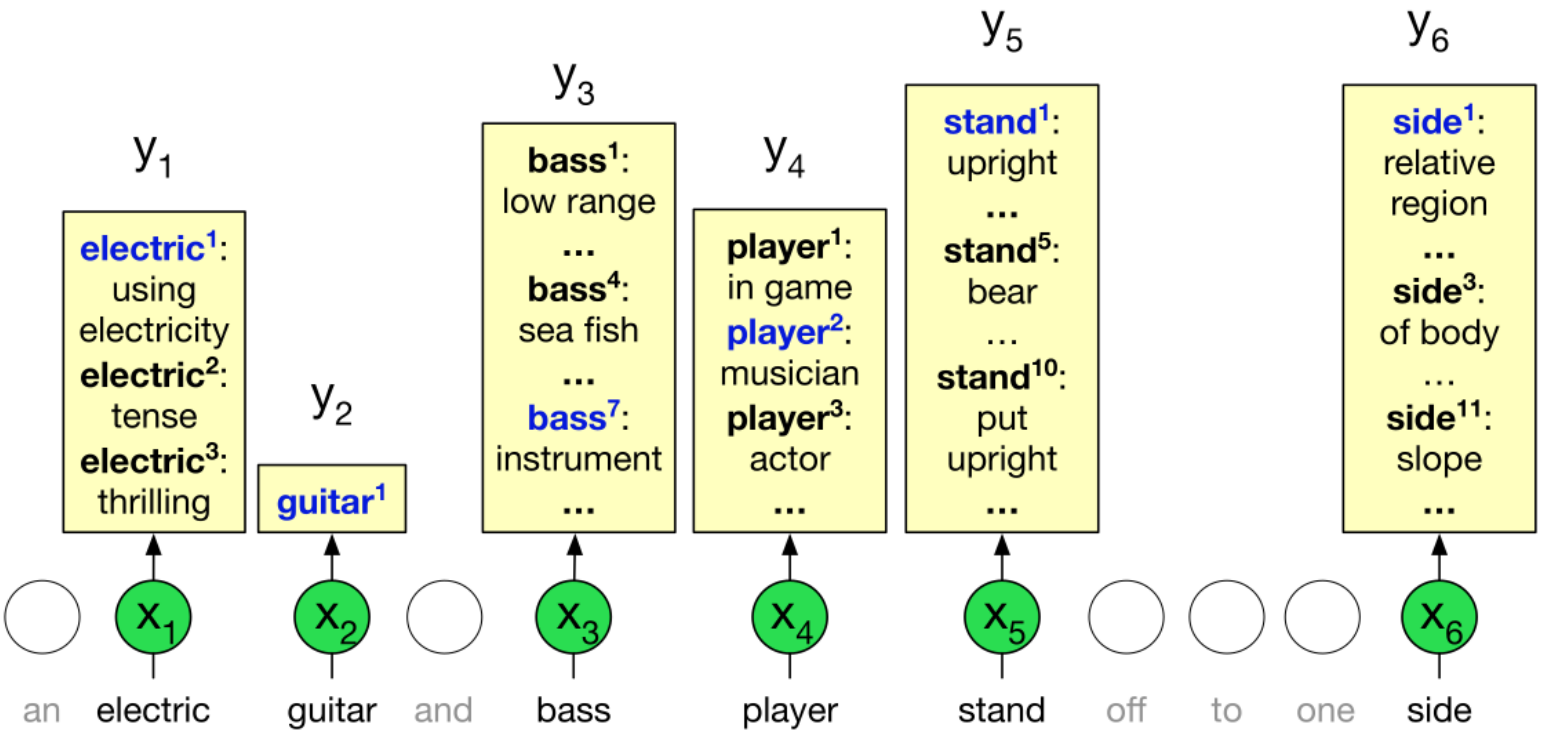
bank.^{1} : ... a bank can hold the investments in a custodial account...

bank.^{1} : ... as agriculture burgeons on the east bank, the river...

Embeddings contextual oferecem um modelo de alta dimensionalidade contínuo sobre o significado que é mais refinado que o discreto

Word sense disambiguation (WSD)

Tarefa de selecionar o sentido correta de uma palavra



Algoritmo de 1-vizinho mais próximo para WSD

No momento de treinamento, pegue um corpus de significado de palavras (como o SEMCOR)
Execute o treinamento do BERT nesse corpus para pegar os embedding de contexto para cada token

- Unindo as representações das últimas 4 camadas do BERT

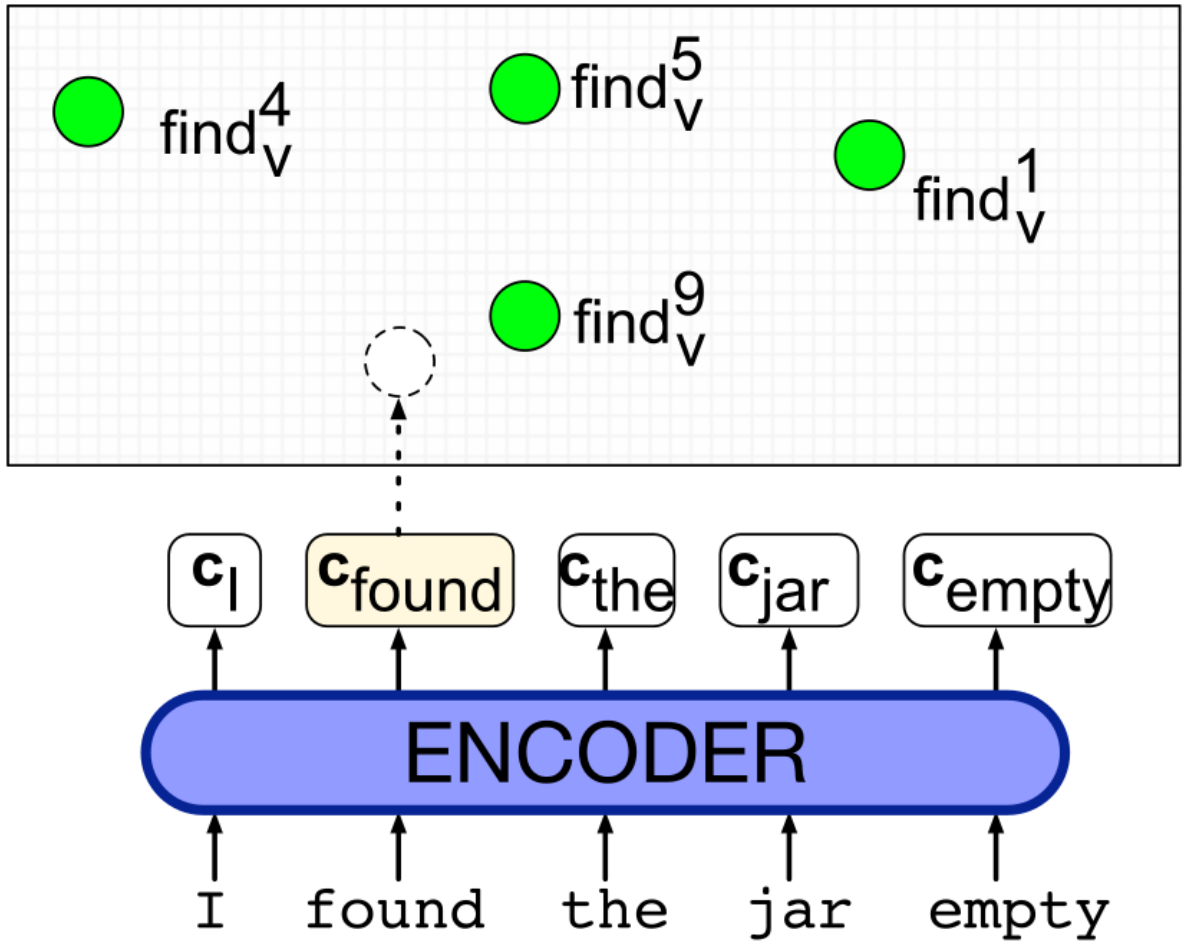
Assim, para cada significado s da palavra w e para os n tokens daquele significado, una os embeddings:

$$v_S = \frac{1}{n} \sum_i v_i \qquad \forall v_i \in tokens(s)$$

No momento do teste, dado um token de uma palavra alvo t, calcule o embedding contextual de t e escolha o significado dos vizinhos mais próximos da base de treinamento

$$sense(t) = argmax_{s \in senses(t)} cosseno(t, v_S)$$

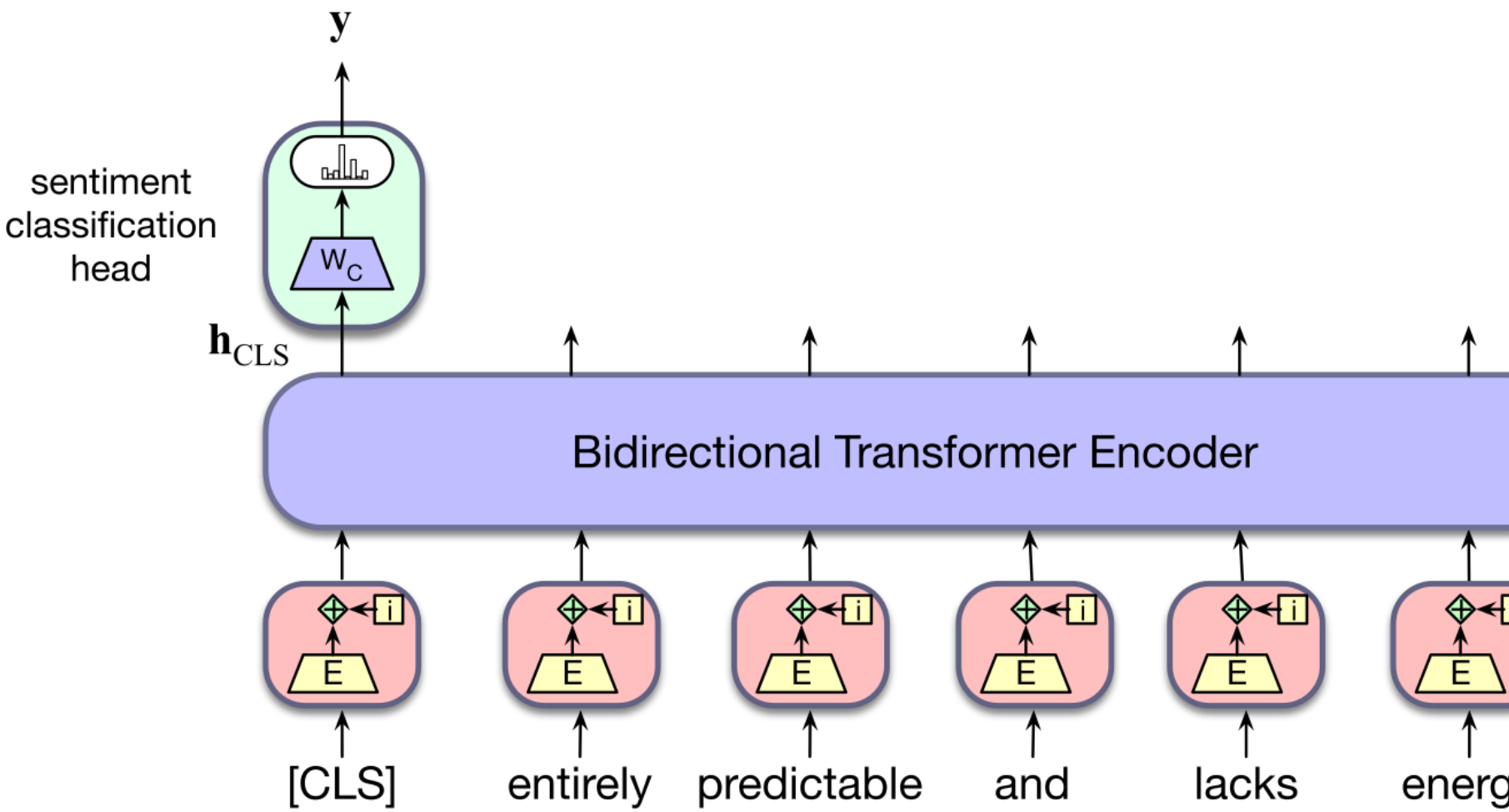
Algoritmo de 1-vizinho mais próximo para WSD



Similaridade e embedding de contexto

- Nós geralmente usamos cosseno
- Mas existem alguns problemas:
  - Embeddings de contexto tendem a ser anisotrópicos: todos apontam aproximadamente na mesma direção, portanto têm altos valores de cossenos (Ethayarajh 2019)
  - Medida cosseno é dominada por um pequeno número dimensões "confusas" e com altos valores. (Timkey and van Shijndel 2021)
  - Cosseno tende a subestimar julgamentos humanos sobre similaridade de significado de palavras para palavras muito frequentes (Zhou et al, 2022)

Fine Tuning para Classificação



Sequence-Pair classification

Atribuindo um rótulo para pares de sequência

- Detecção de paráfrases (duas sentenças se parafraseam?)
- Implicação Lógica (A sentença A implica logicamente na sentença B)
- Coerência de discurso (O quanto a sentença B se relaciona com a sentença A)

Inferência em Linguagem Natural

Pares de sentenças são dados 3 rótulos

- Neutro
  1. *Jon walked back to the town to the smithy.*
  2. *Jon traveled back to his hometown.*
- Contraditórias
  1. *Escritório de Informações Turísticas pode ser muito útil*
  2. *Escritório de Informações Turísticas nunca é útil*
- Implicações
  1. *Eu estou confuso*
  2. *Nada disso está claro para mim*

Algoritmo: Passe os pares de premissa/hipótese pelo Encoder Bidirecional e use o vetor de saída do token [CLS] como entrada de uma cabeça de classificação.

Fine-Tuning para rotulação de sequência

Atribua um rótulo para cada token na sequência

- Named Entity Recognition
- Part of Speech Tagging

Reconhecimento de Entidades Nomeadas

Uma entidade nomeada é qualquer coisa que pode ser referenciado com um nome próprio: Uma pessoa, uma Localização, uma Organização

Reconhecimento de entidades Nomeadas (NER): encontre partes do texto que constituem nomes próprios em rotule o tipo de entidade.

Type	Tag	Sample Categories	Example sentences
People	PER	people, characters	**Turing** is a giant of computer science.
Organization	ORG	companies, sports teams	The **IPCC** warned about the cyclone.
Location	LOC	regions, mountains, seas	**Mt. Sanitas** is in **Sunshine Canyon**.
Geo-Political Entity	GPE	countries, states	**Palo Alto** is raising the fees for parking.

Reconhecimento de Entidades Nomeadas

Citing high fuel prices, [ORG **United Airlines**] said [TIME **Friday**] it has increased fares by [MONEY **$6**] per round trip on flights to some cities also served by lower-cost carriers. [ORG **American Airlines**], a unit of [ORG **AMR Corp.**], immediately matched the move, spokesman [PER **Tim Wagner**] said. [ORG **United**], a unit of [ORG **UAL Corp.**], said the increase took effect [TIME **Thursday**] and applies to most routes where it competes against discount carriers, such as [LOC **Chicago**] to [LOC **Dallas**] and [LOC **Denver**] to [LOC **San Francisco**].

BIO Tagging

Um método que nos permiter transformar uma tarefa de segmentação (encontrar limites das entidades) em uma tarefa de classificação.

[PER **Jane Villanueva**] of [ORG **United**] , a unit of [ORG **United Airlines Holding**] , said the fare applies to the [LOC **Chicago**] route.

Words	**IO Label**	**BIO Label**	**BIOES Label**
Jane	I-PER	B-PER	B-PER
Villanueva	I-PER	I-PER	E-PER
of	O	O	O
United	I-ORG	B-ORG	B-ORG
Airlines	I-ORG	I-ORG	I-ORG
Holding	I-ORG	I-ORG	E-ORG
discussed	O	O	O
the	O	O	O
Chicago	I-LOC	B-LOC	S-LOC
route	O	O	O
.	O	O	O

Rotulagem de sequência

$$\mathbf{y_i} = \text{softmax}(\mathbf{h_i^L W_K})$$
$$\mathbf{t_i} = \text{argmax}_k(\mathbf{y}_i)$$