SCC0633/5908 Processamento de Linguagem Natural

E a história avança...

Modelos distribucionais, Anotação de córpus

SCC5908 Introdução ao Processamento de Língua Natural
SCC0633 Processamento de Linguagem Natural

Prof. Thiago A. S. Pardo

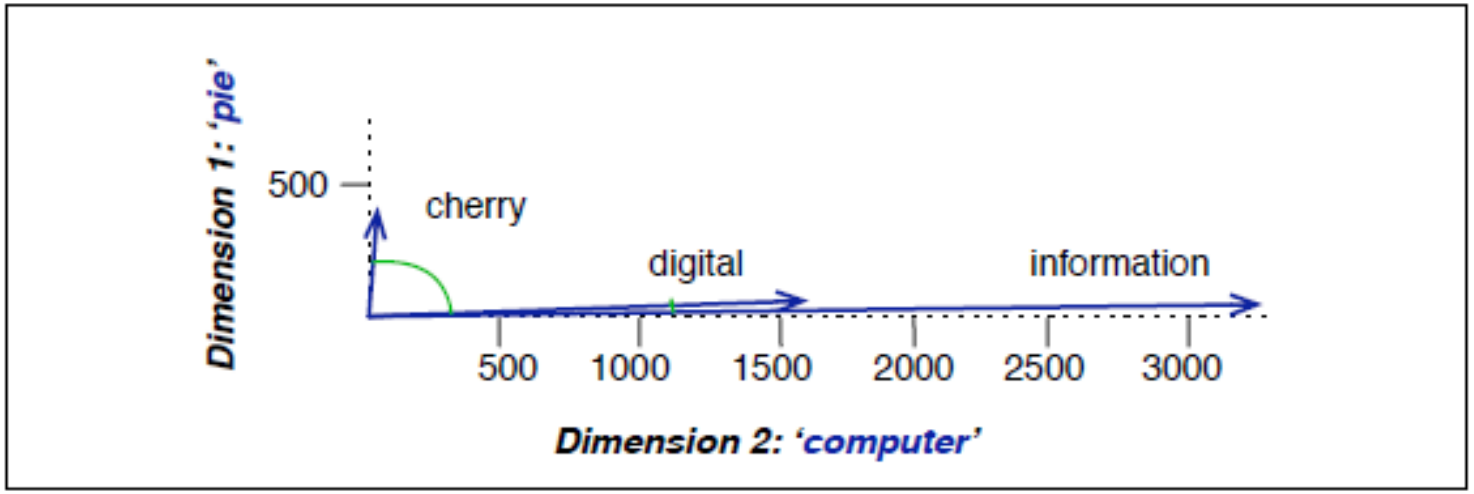
Relembrando

- Métodos de contagem e Zipf
- Estatística, Bayes e modelo *noisy-channel*
- Hipótese distribucional, semântica e vetores
- De representações *bag-of-words* aos *embeddings* modernos
  - Vetores esparsos vs. densos

Relembrando

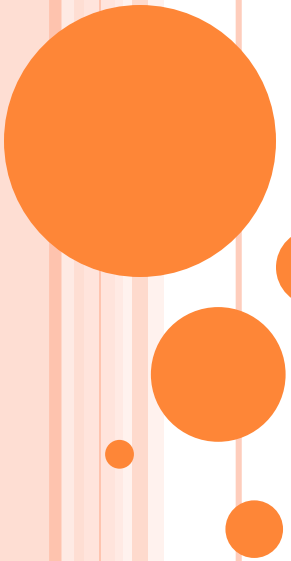
- Matriz termo-contexto

	aardvark	...	computer	data	result	pie	sugar	...
cherry	0	...	2	8	9	442	25	
strawberry	0	...	0	0	1	60	19	
digital	0	...	1670	1683	85	5	4	
information	0	...	3325	3982	378	5	13	



Relembrando

- Algumas abordagens já tradicionais (apesar de algumas serem bastante recentes)
  - SVD – *Singular Value Decomposition*
    - LSA (Deerwester et al., 1988)
  - Redes neurais (Bengio et al., 2003) e modelos preditivos
    - "Skip-gram" e "continuous bag of words" (Mikolov et al., 2013)
      - Métodos incorporados no pacote **word2vec**
  - Métodos baseados em contagem
    - GloVe (Pennington et al., 2014)
  - BERT (Devlin et al., 2019) e modelos contextuais
    - BERTimbau para o português (Souza et al., 2020)
- E muitas outras variações, para diferentes propósitos, inclusive
  - FastText, Wang2Vec, Doc2Vec, ELMo, RoBERTa, DeBERTa, Product2Vec, code2vec, etc.



Word2vec

WORD2VEC

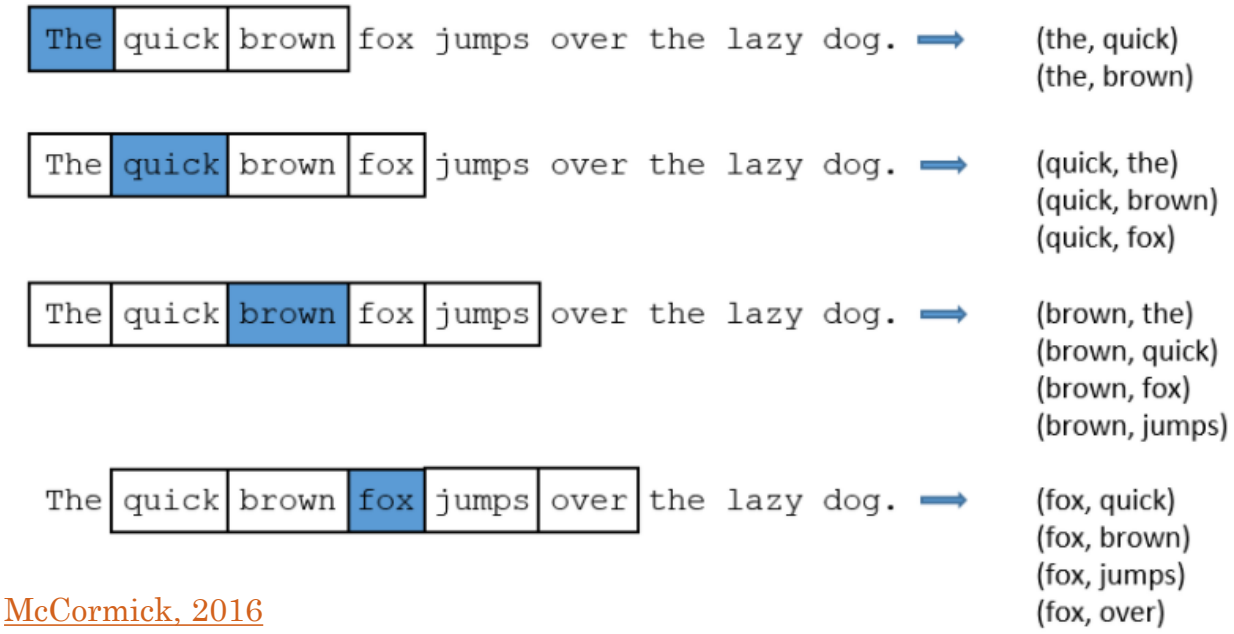
(MIKOLOV ET AL., 2013)

- Revolucionou a área ao apresentar um método relativamente "simples" para aprendizado de vetores densos
  - Com estratégias para otimizar o processo
- Os vetores provêm de "pesos" aprendidos por uma rede neural treinada para uma tarefa "fake"
  - Previsão de uma palavra dado seu contexto
- Na época, o autor principal trabalhava no Google
  - A importância das empresas no avanço recente do PLN!

Dados para aprendizado

- Previsão de palavras que ocorrem no mesmo contexto
  - Intuição: vetores de palavras que ocorrem com o mesmo contexto ("janela") tendem a convergir para valores próximos durante o aprendizado
  - Dados de treino facilmente acessíveis e abundantes!

Exemplo: geração de dados de treino considerando uma janela de +- 2 palavras

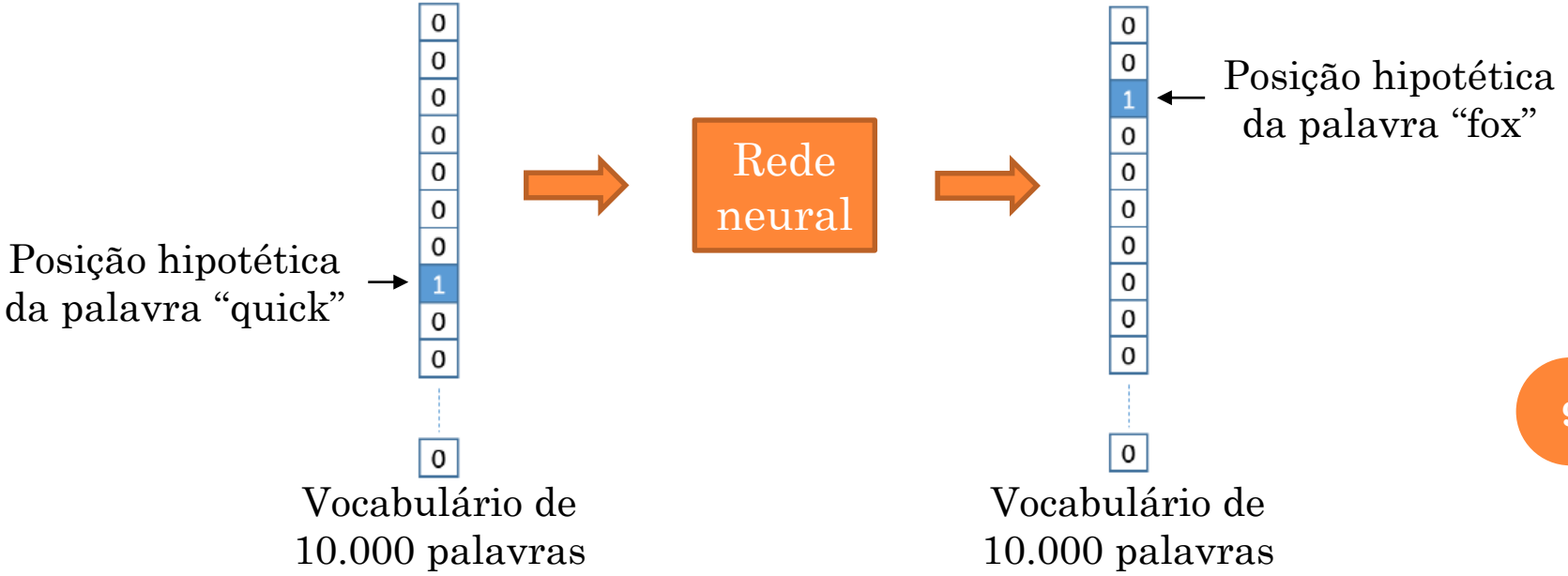


McCormick, 2016

Representação de entrada e saída

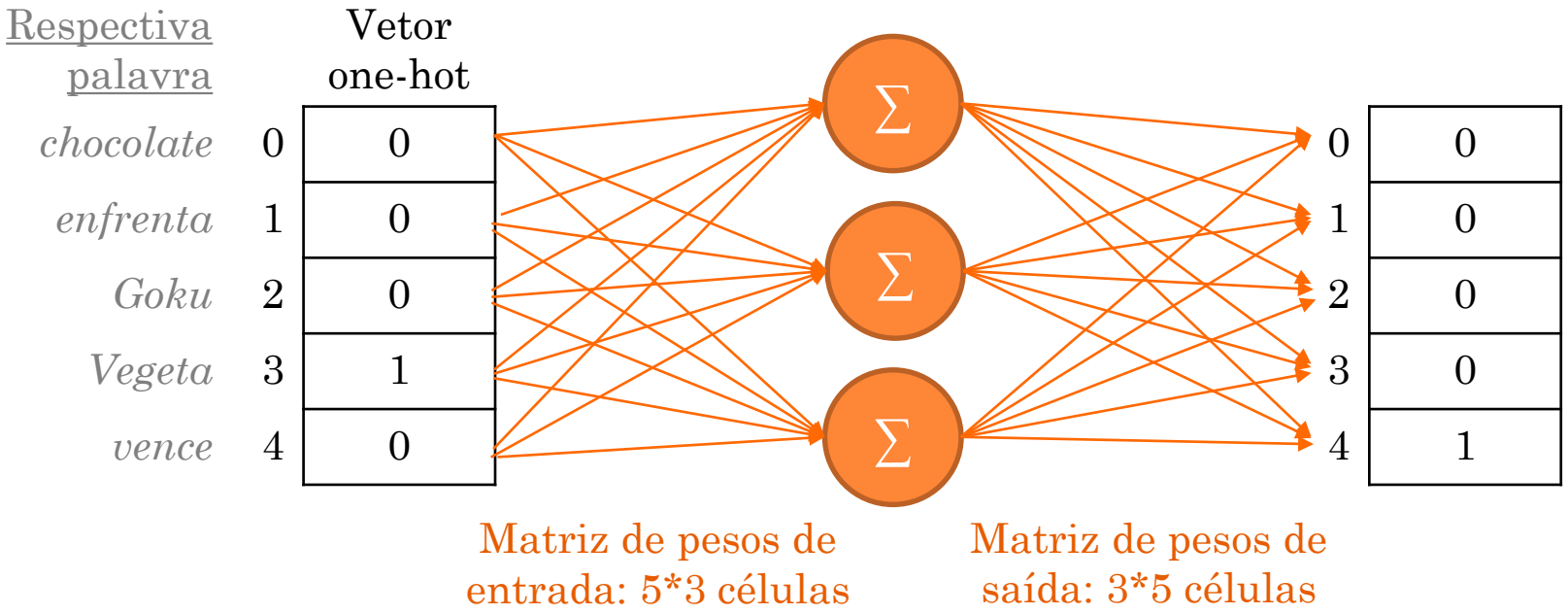
- Vetores one-hot para vocabulário
  - Vetor de cada palavra tem o tamanho do vocabulário
  - A palavra de interesse é marcada com ‘1’, enquanto as demais com ‘0’

- Idealmente, supondo o aprendizado do par (quick, fox) em um vocabulário de 10.000 palavras

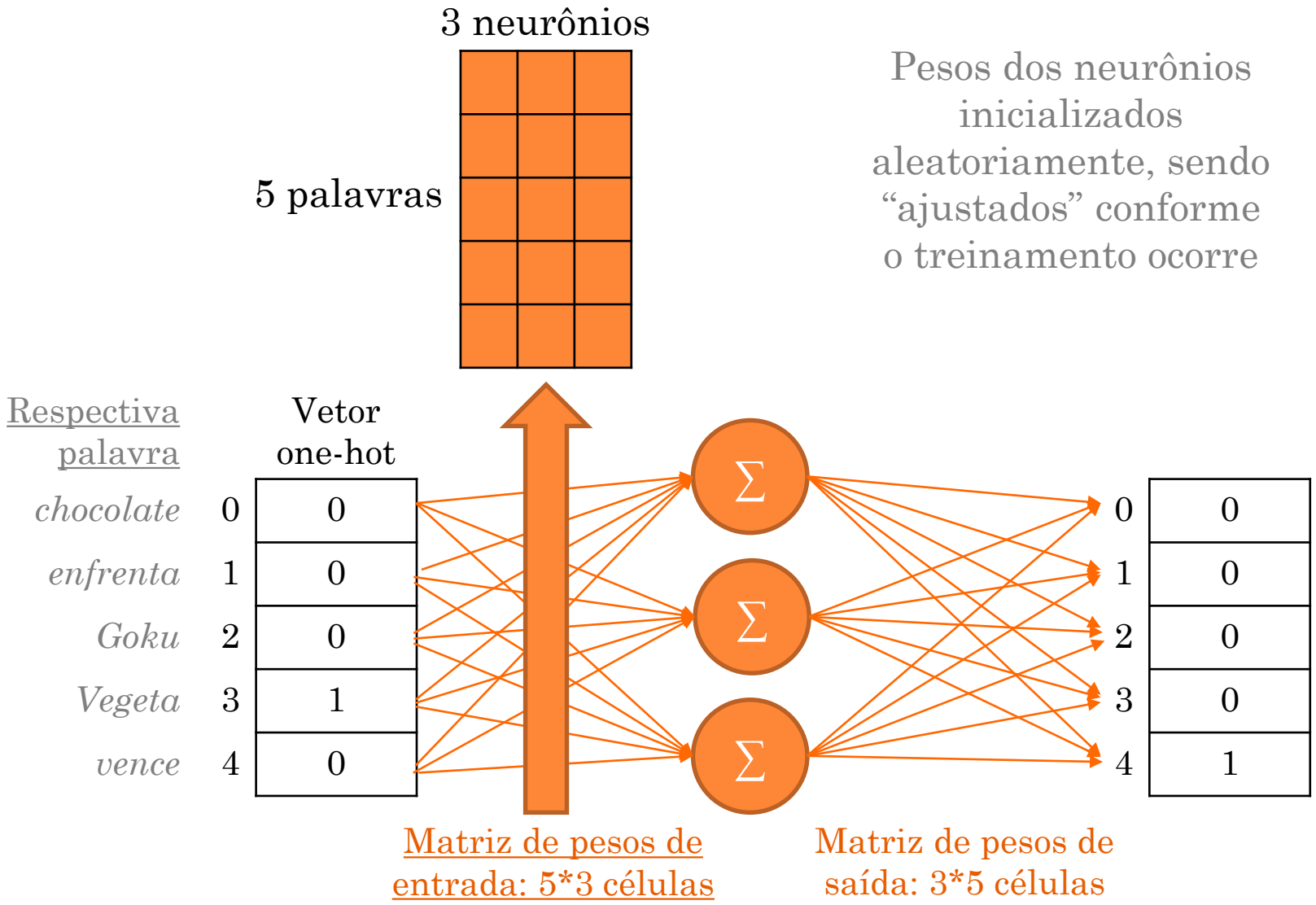


"Abrindo" a rede neural

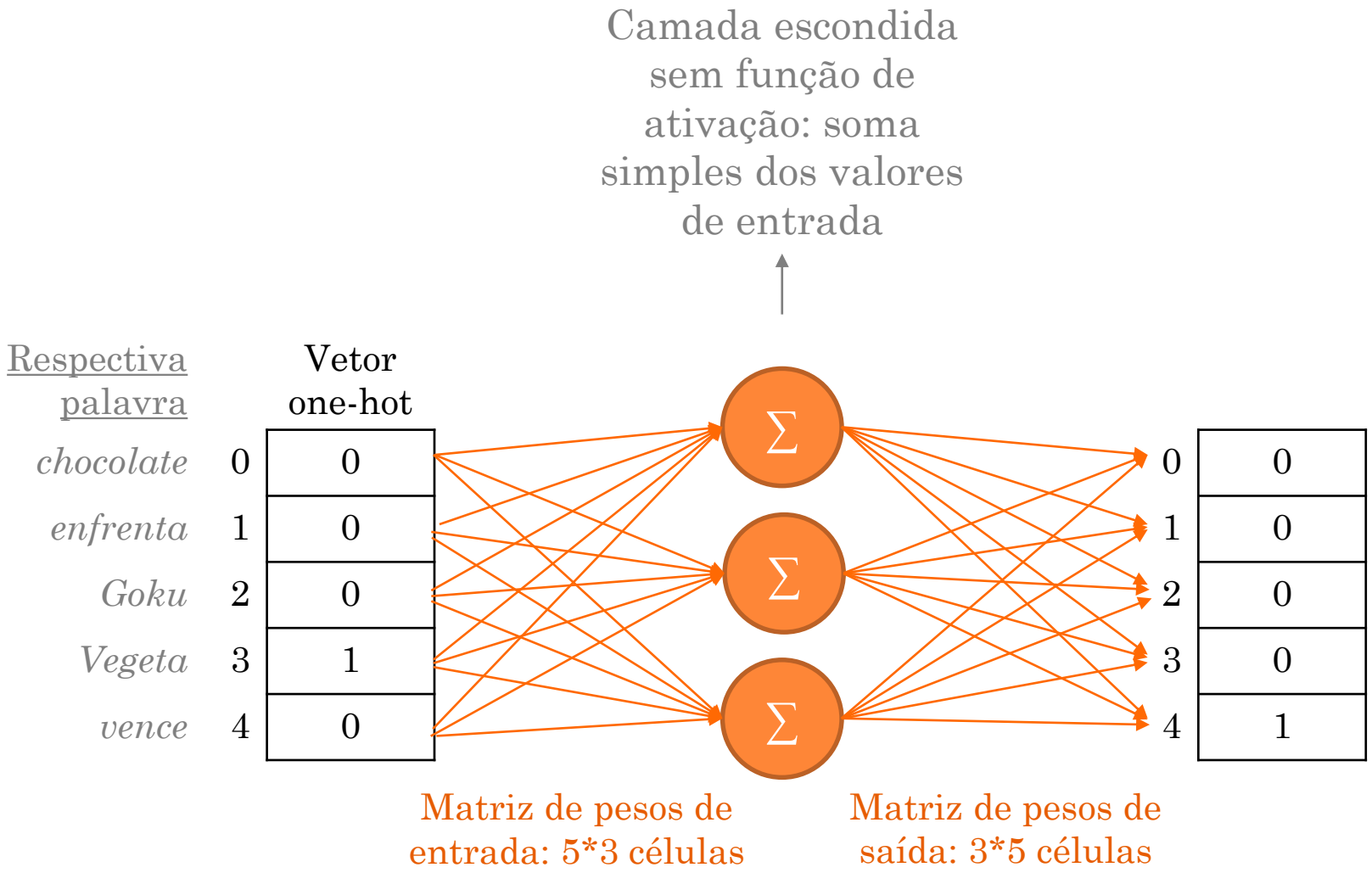
- Exemplo hipotético simples (totalmente irreal, apenas para fins didáticos)
  - Língua com vocabulário de 5 palavras
  - 3 neurônios na camada escondida da rede



"Abrindo" a rede neural

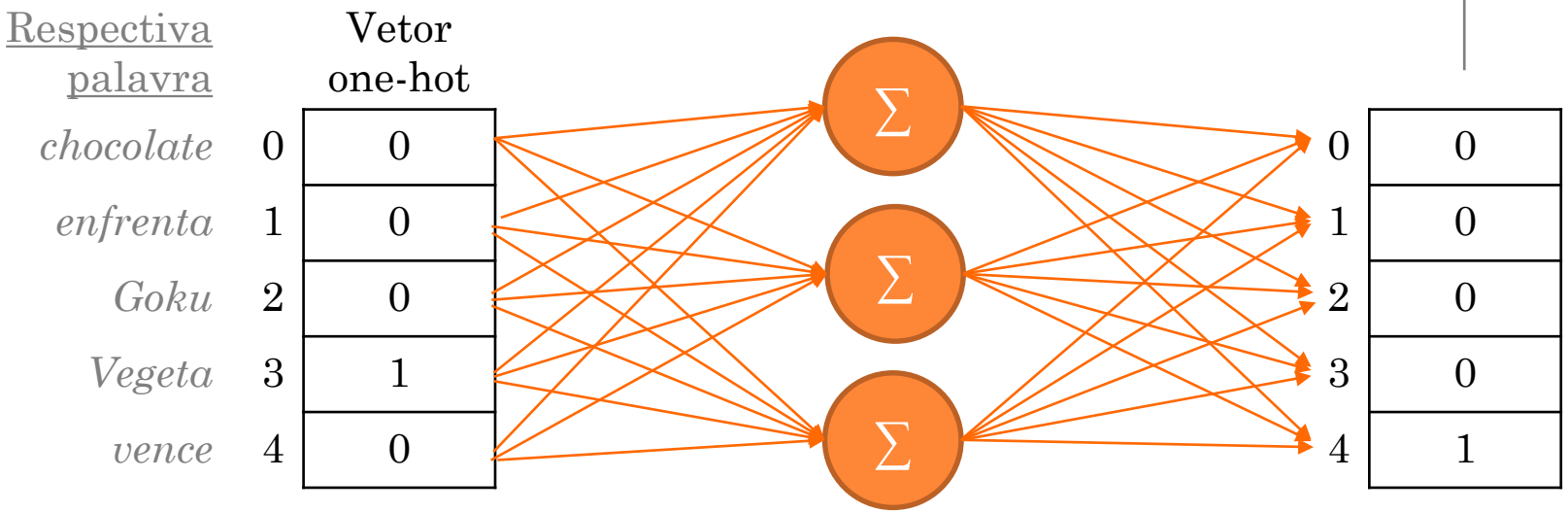


"Abrindo" a rede neural



"Abrindo" a rede neural

Também consiste em uma camada de neurônios com função softmax, que indica a probabilidade de cada palavra de saída



Matriz de pesos de entrada: 5*3 células

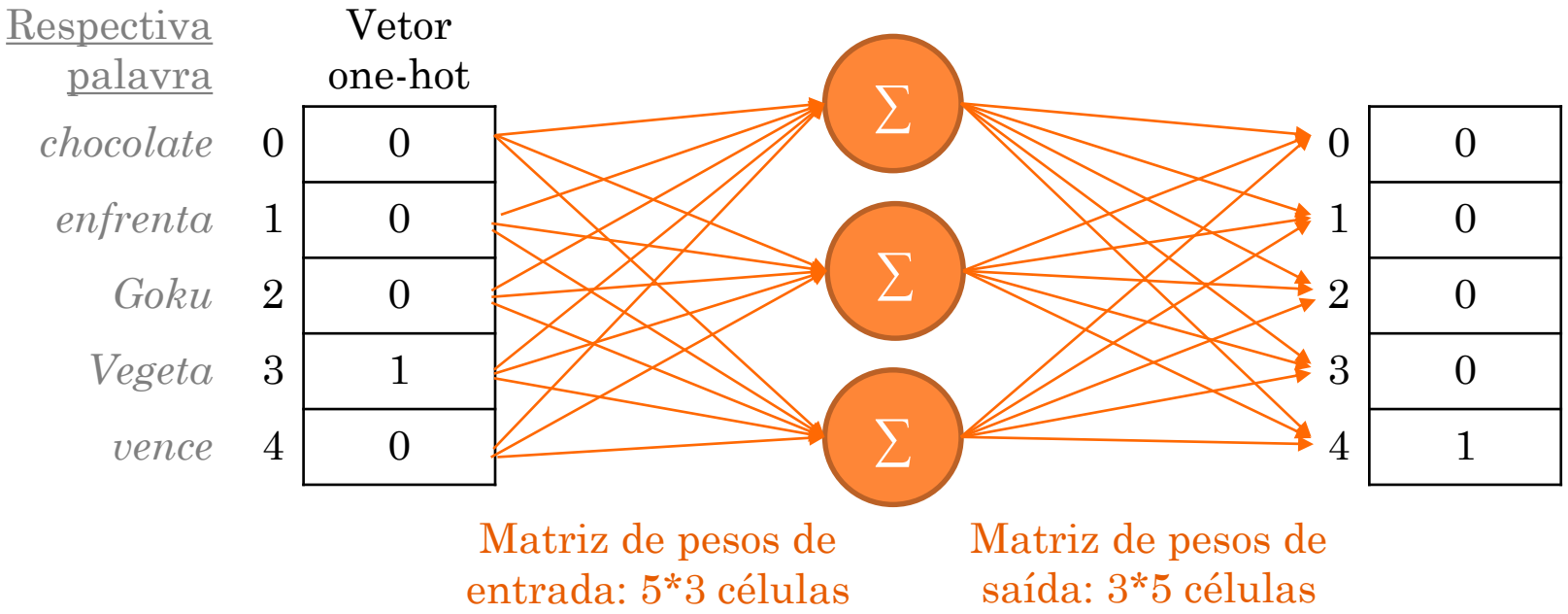
Matriz de pesos de saída: 3*5 células

"Abrindo" a rede neural

Intuição para o aprendizado: palavras que ocorrem nos mesmos contextos devem "forçar a rede" a aprender essa similaridade entre elas.

Por exemplo: tanto "**Goku**" quanto "**Vegeta**" ocorrem frequentemente com os termos "**enfrenta**" e "**vence**"

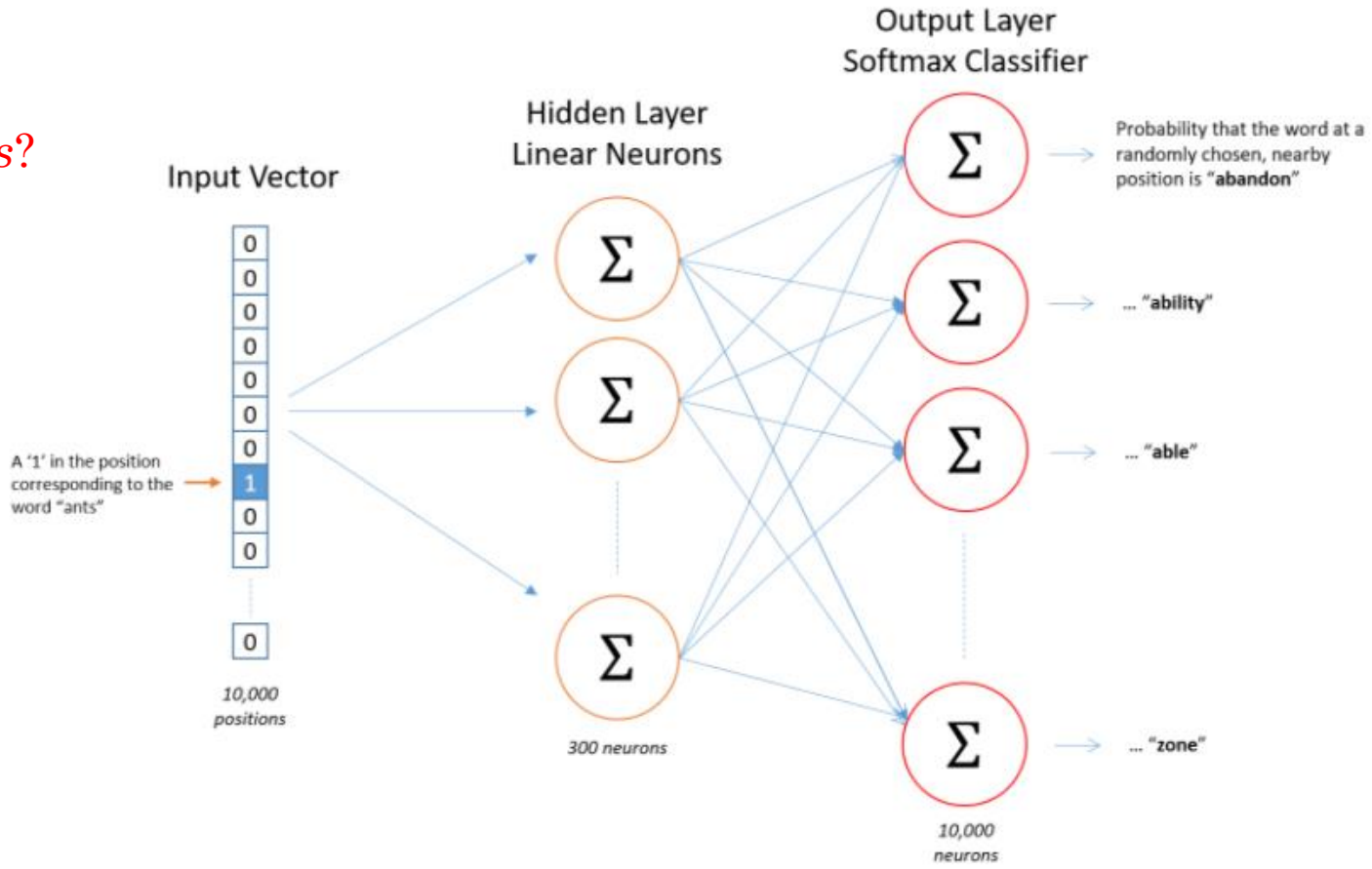
→ portanto, "Goku" e "Vegeta" têm alguma similaridade



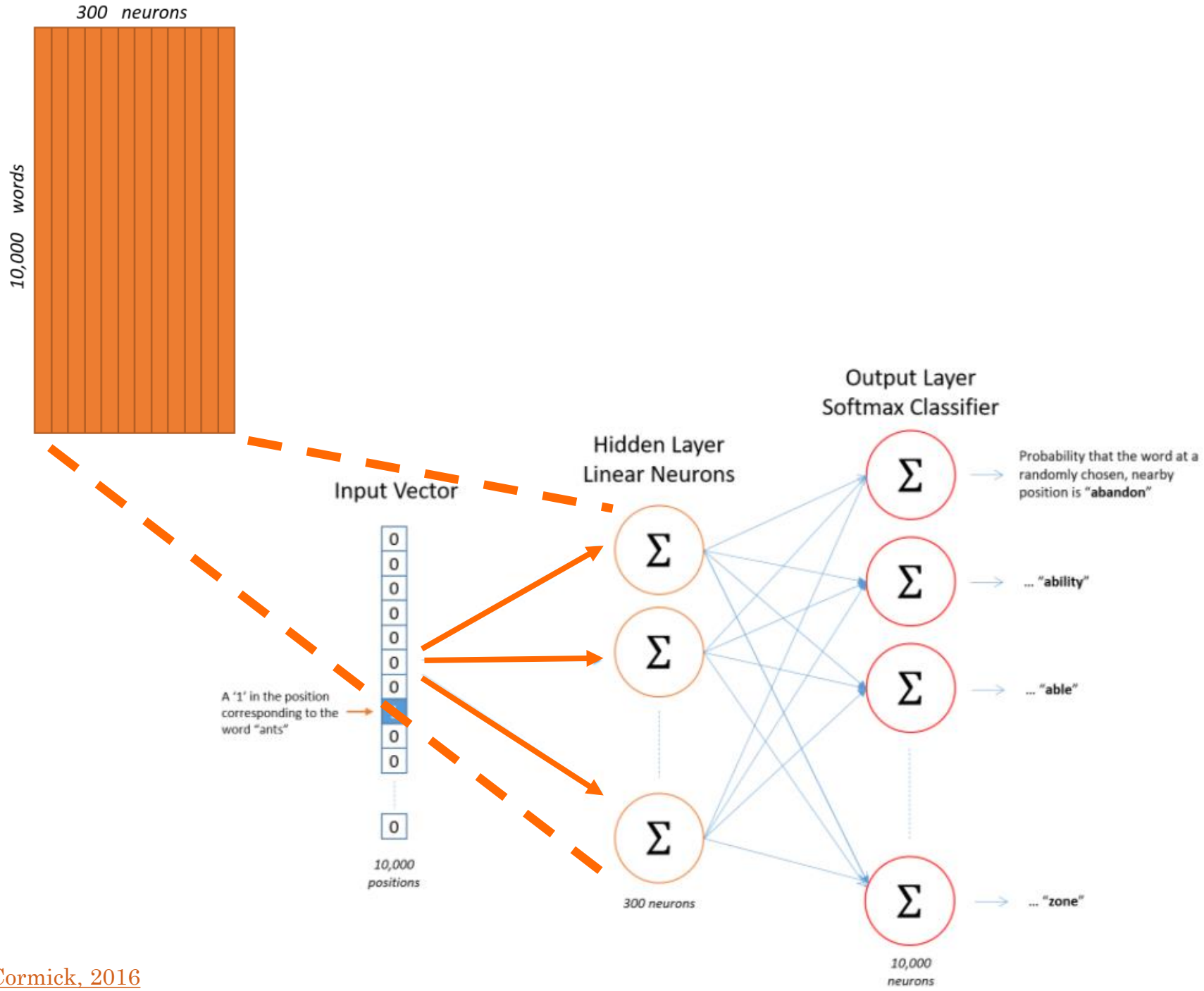
GENERALIZANDO

- Situação mais real
  - Vocabulário de 10.000 palavras e 300 neurônios na camada escondida

Onde estão os *embeddings*?



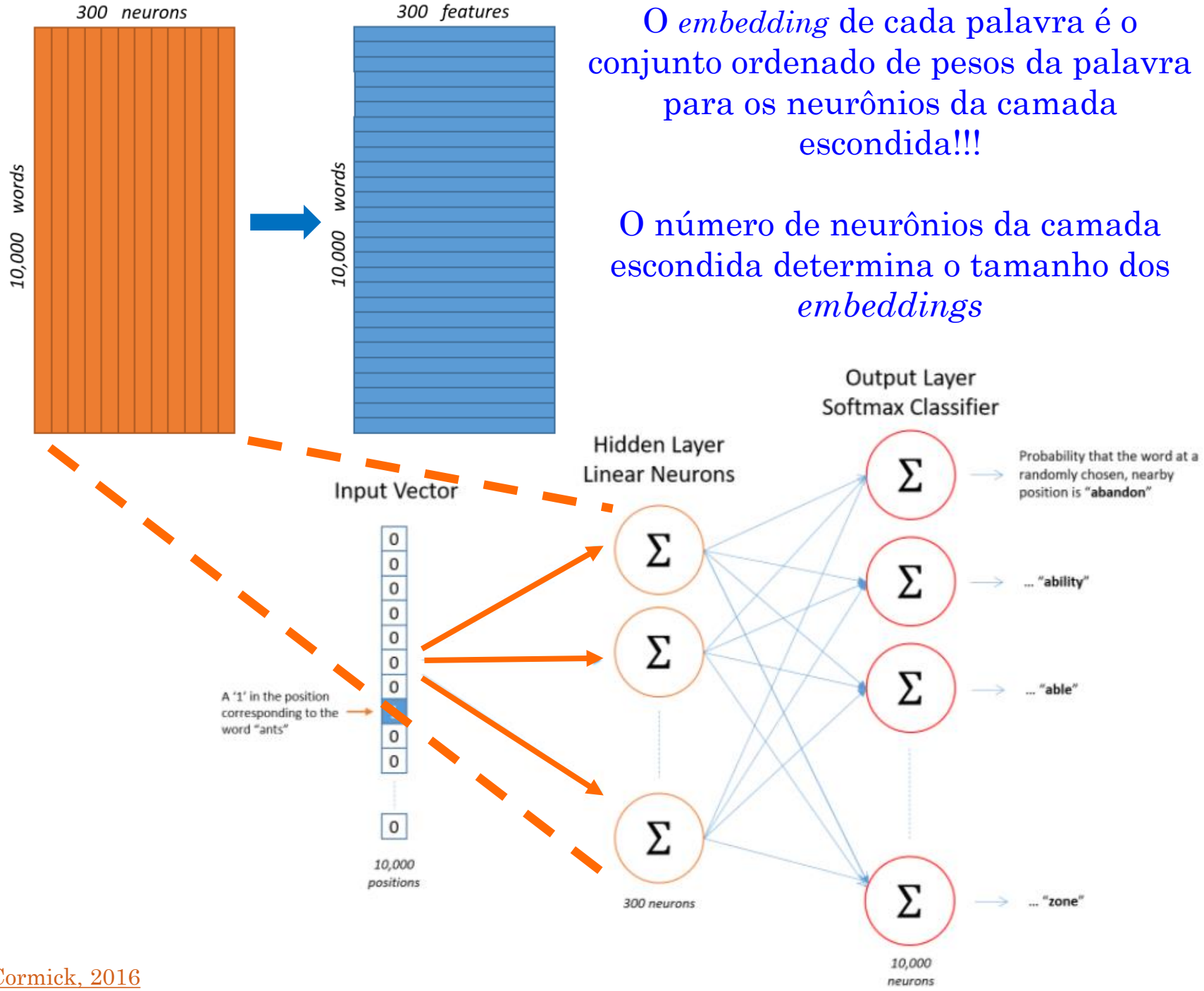
McCormick, 2016



McCormick, 2016

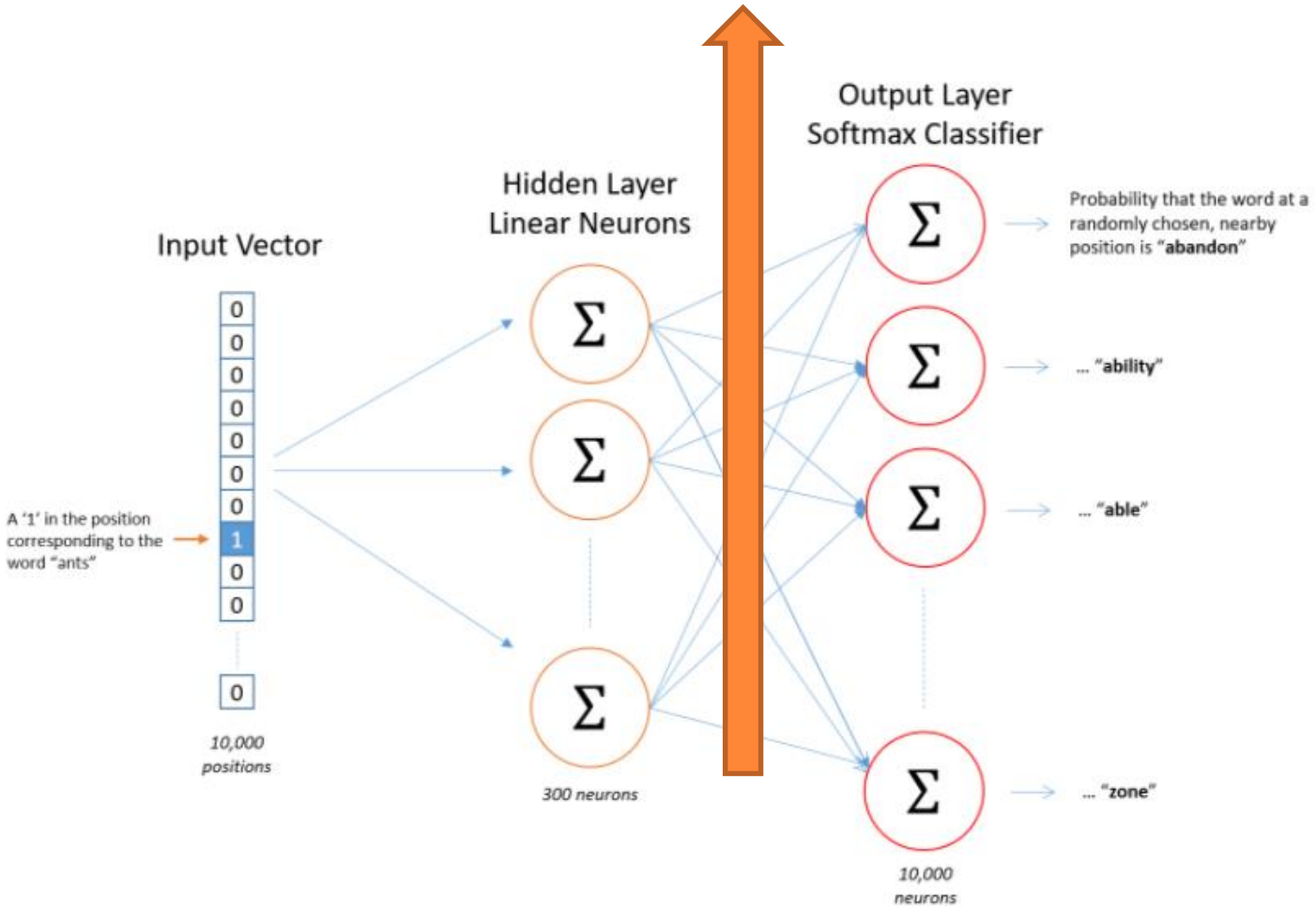
O *embedding* de cada palavra é o conjunto ordenado de pesos da palavra para os neurônios da camada escondida!!!

O número de neurônios da camada escondida determina o tamanho dos *embeddings*



McCormick, 2016

No final, essa matriz da camada de saída é irrelevante, pois foi utilizada para simular a tarefa “fake”



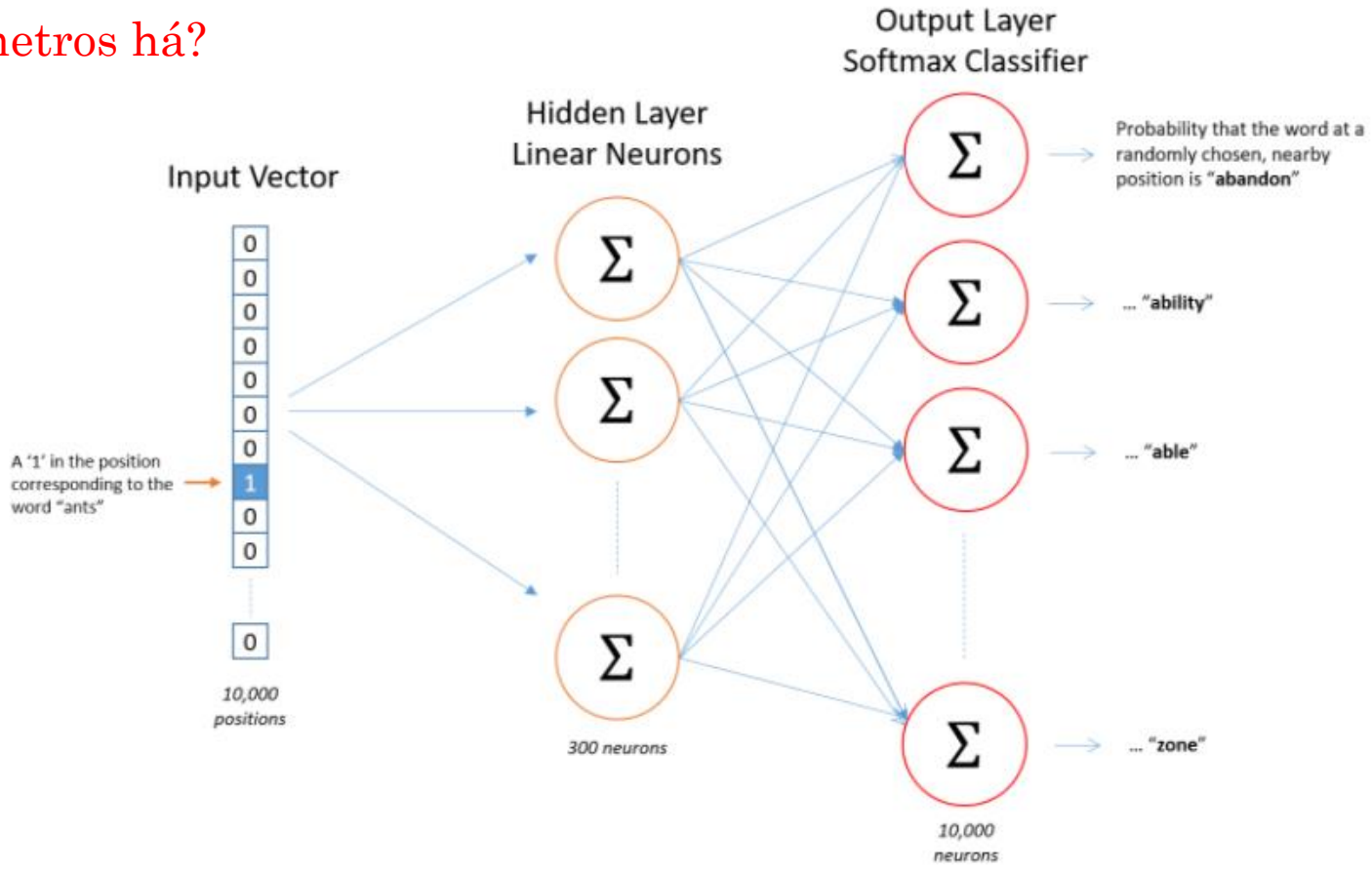
McCormick, 2016

GENERALIZANDO

- Situação mais real
  - Vocabulário de 10.000 palavras e 300 neurônios na camada escondida

Quantos parâmetros há?

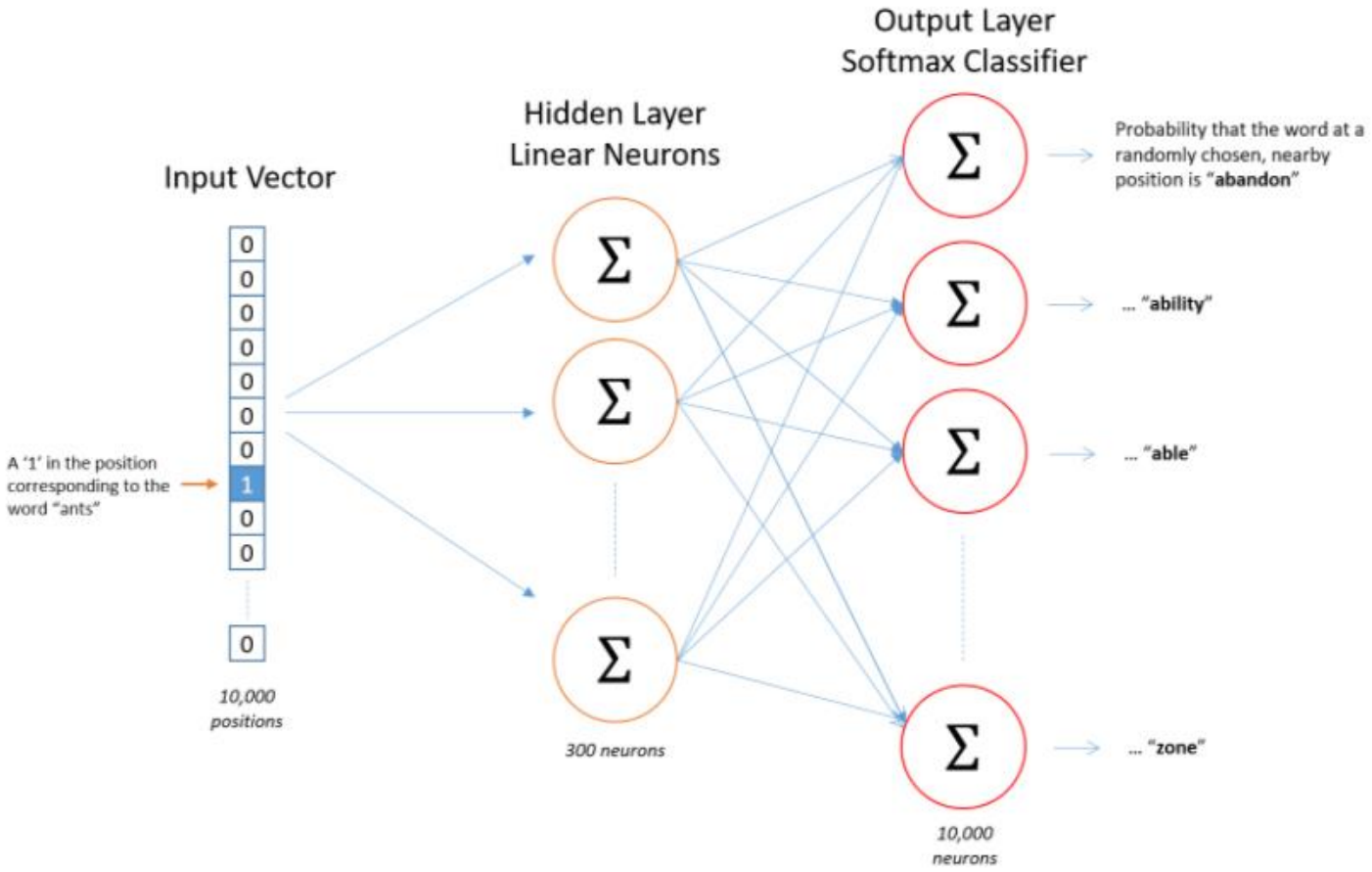
6.000.000!!!



McCormick, 2016

O TREINAMENTO

- Na camada escondida, somente os pesos da respectiva palavra de entrada são processados
  - Por quê?



McCormick, 2016

Recuperando nosso exemplo simples

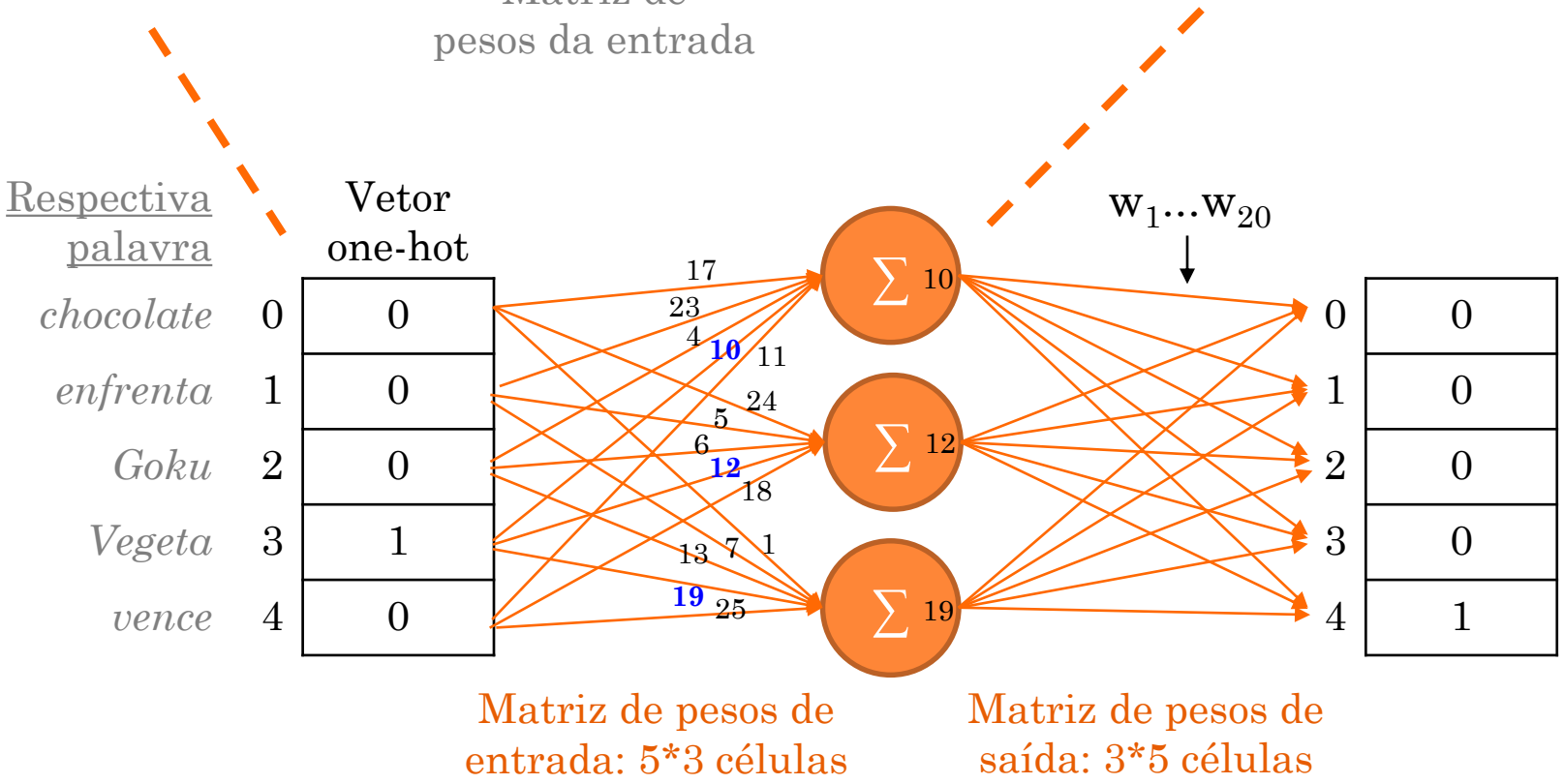
$$[0 \quad 0 \quad 0 \quad 1 \quad 0] \times \begin{bmatrix} 17 & 24 & 1 \\ 23 & 5 & 7 \\ 4 & 6 & 13 \\ 10 & 12 & 19 \\ 11 & 18 & 25 \end{bmatrix} = [10 \quad 12 \quad 19]$$

Vetor one-hot deitado

Matriz de pesos da entrada

Saídas dos neurônios da camada escondida

Valores hipotéticos

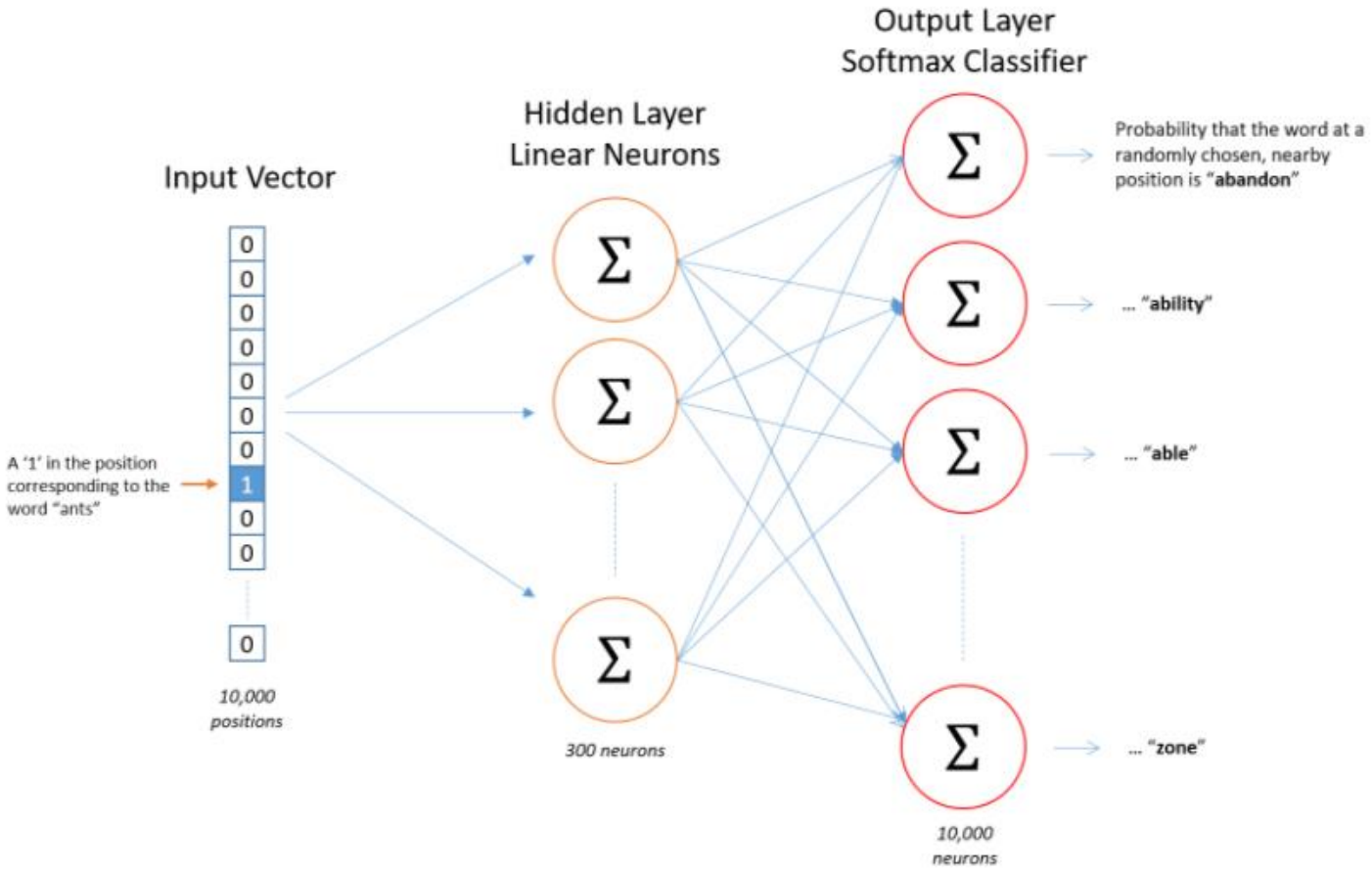


Matriz de pesos de entrada: 5*3 células

Matriz de pesos de saída: 3*5 células

O TREINAMENTO

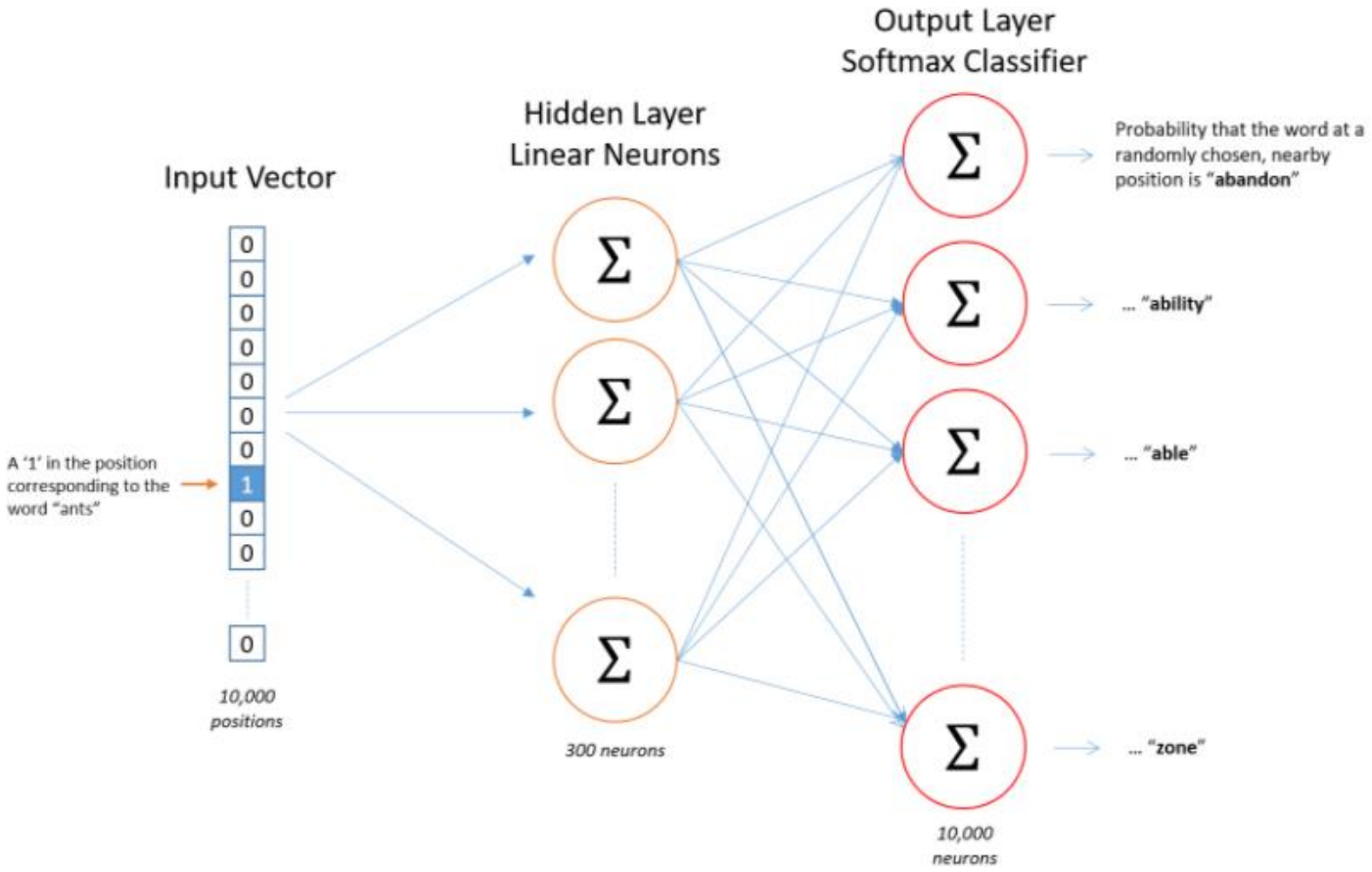
- Ajuste dos pesos via algoritmo *backpropagation*



McCormick, 2016

O TREINAMENTO

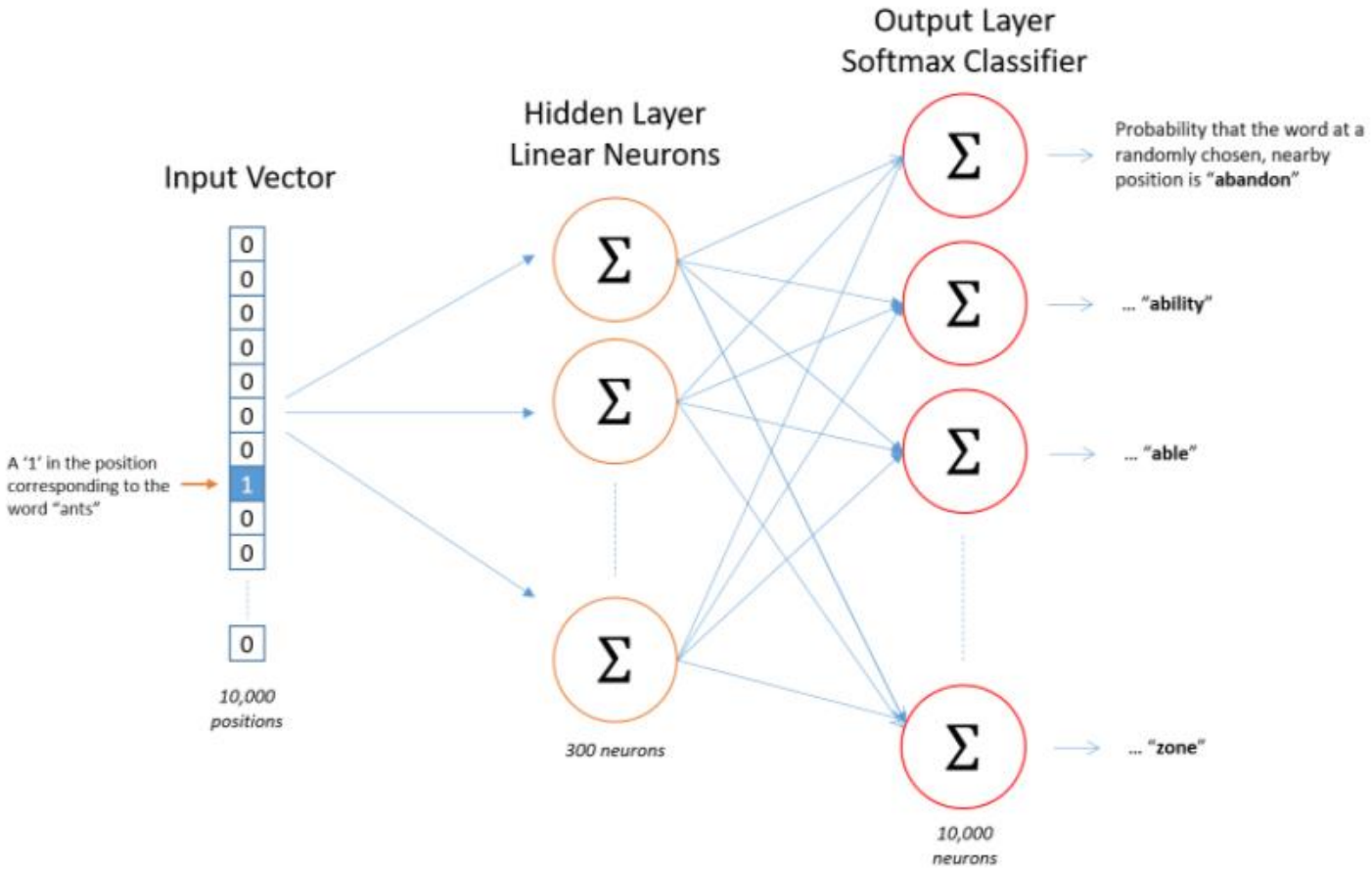
- Estratégias
  - Sub-amostragem de palavras muito frequentes (por exemplo, artigos e preposições se combinam com quase qualquer coisa)



McCormick, 2016

O TREINAMENTO

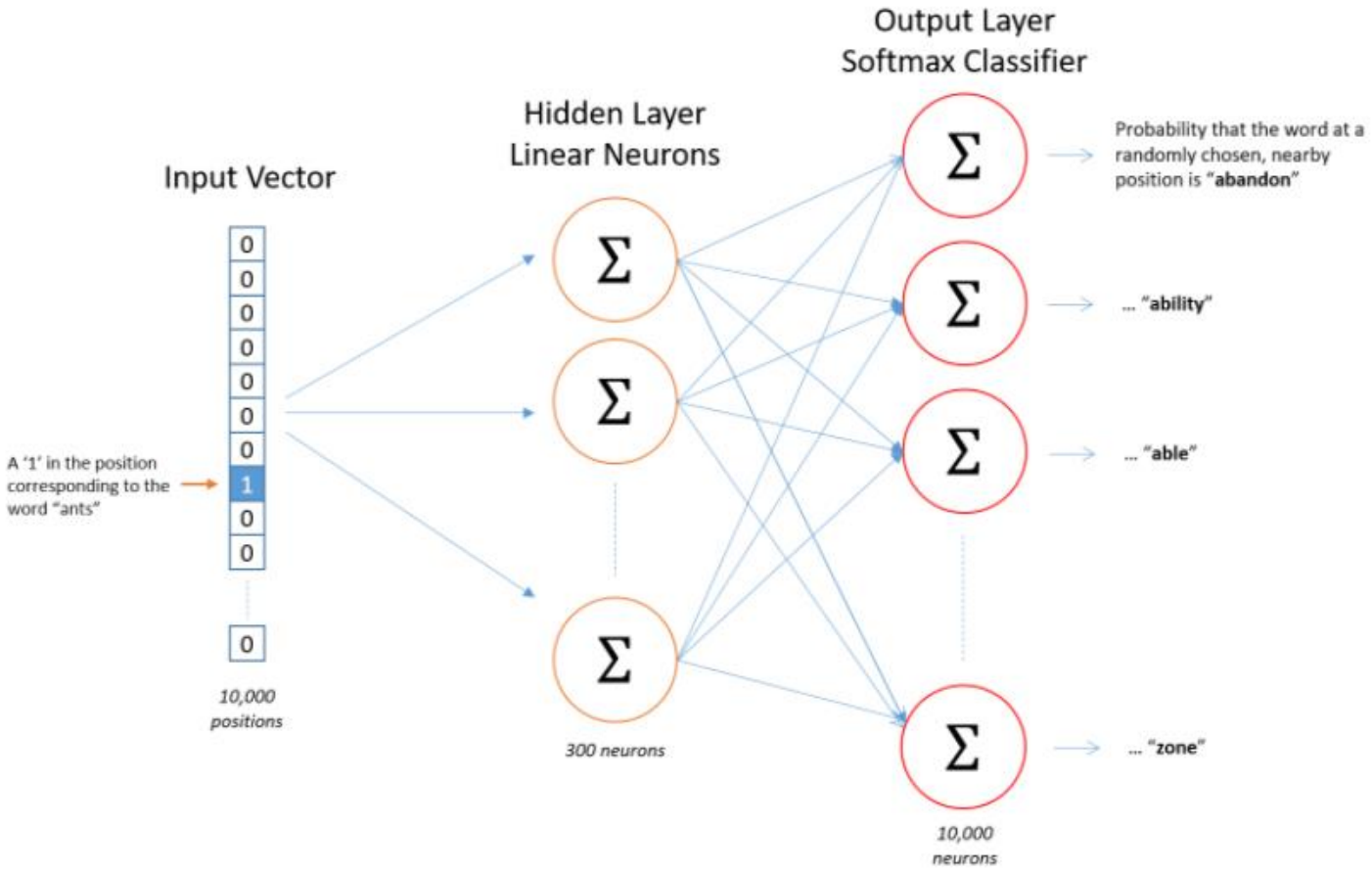
- Estratégias
  - Contextos de tamanhos variáveis, de 1 a N palavras (palavras mais próximas acabam tendo algum destaque, pois contextos maiores também incluem os contextos menores)



McCormick, 2016

O TREINAMENTO

- Estratégias
  - Amostragem negativa: para não treinar todos os pesos para cada instância, uma amostragem de exemplos negativos é utilizada para cada exemplo positivo (assim, apenas parte dos pesos é ajustada)



McCormick, 2016

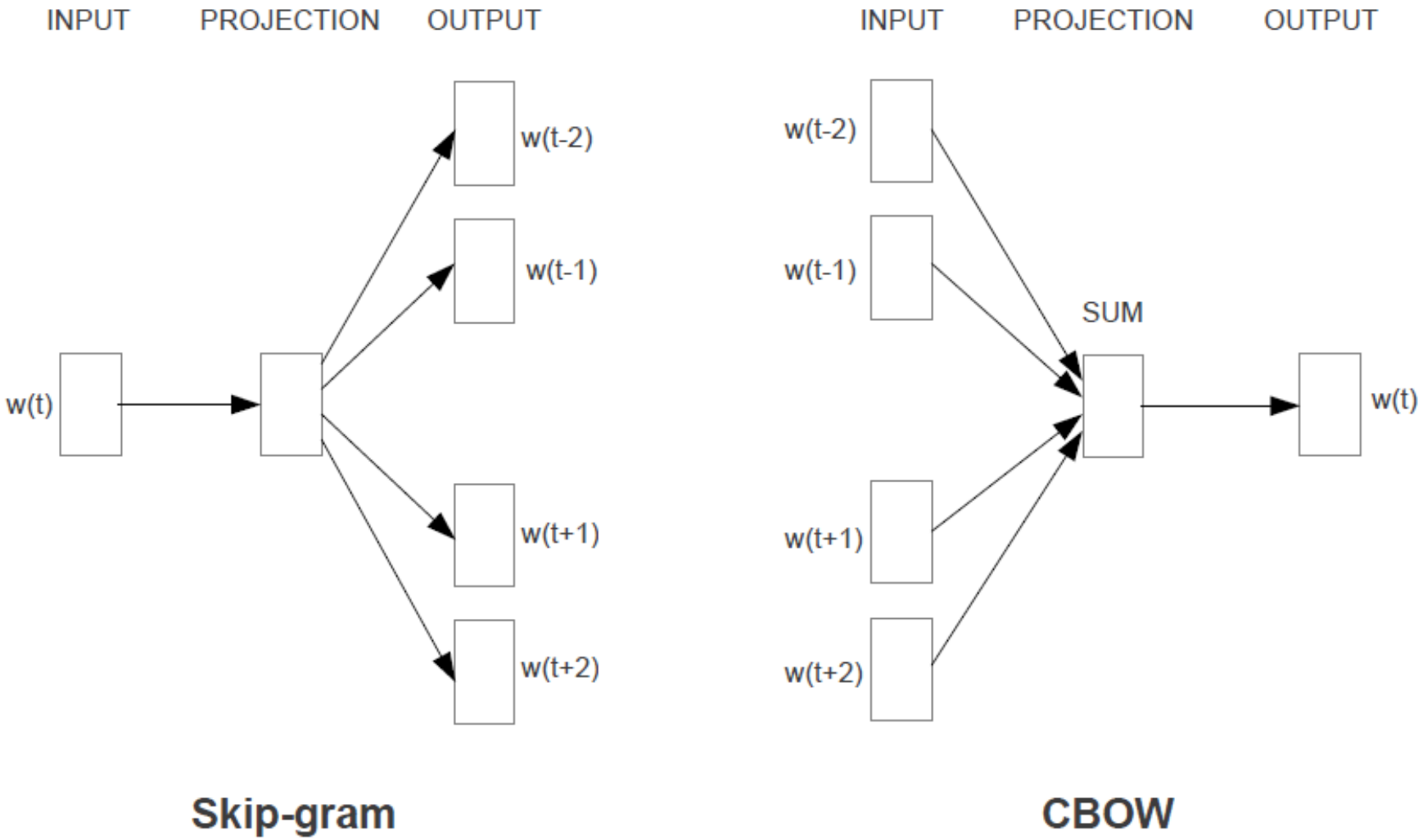
WORD2VEC

- Até agora, modelo *skip-gram*
  - Uma palavra prevê palavras de seu contexto

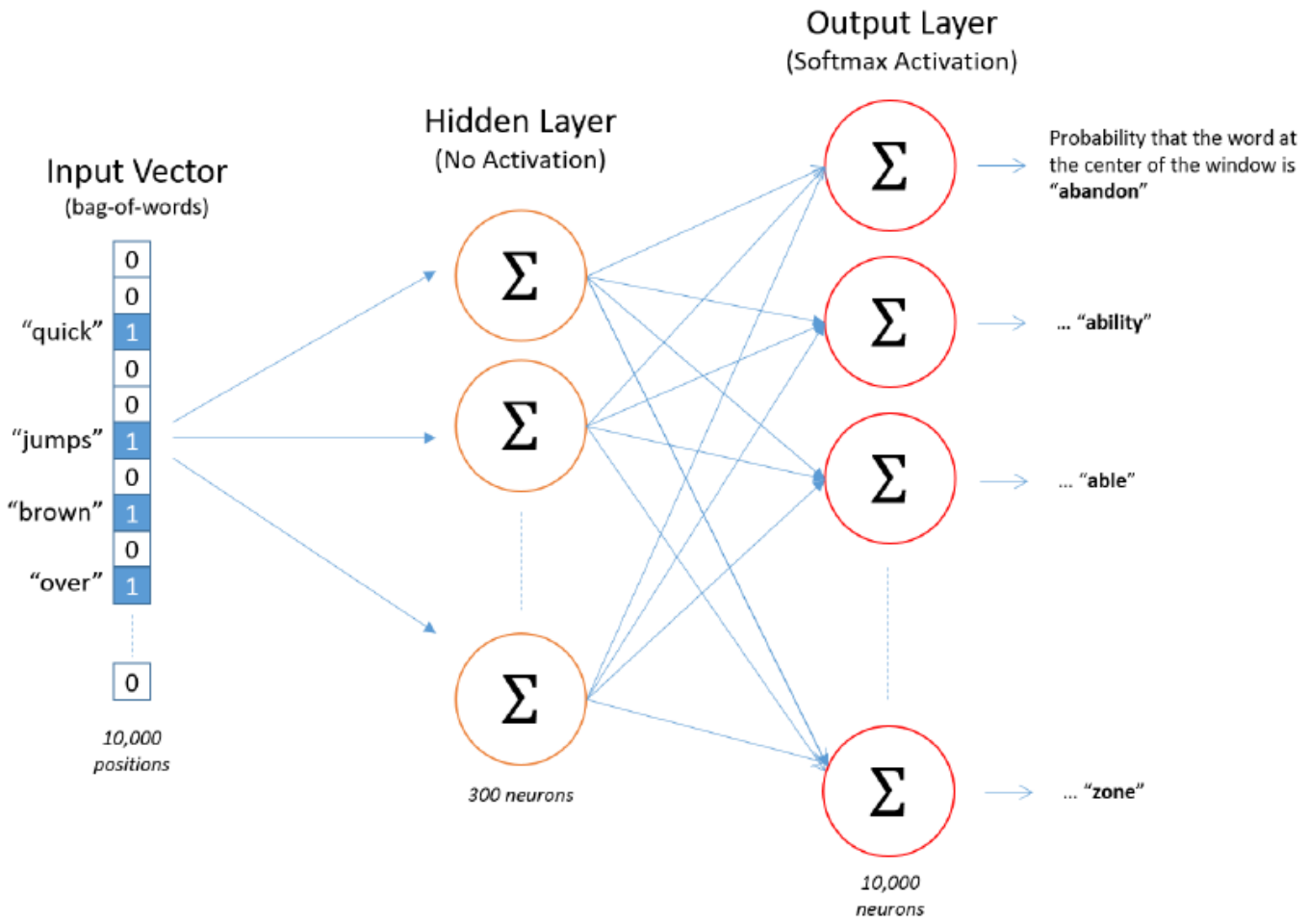
- Mas há também o *Continuous Bag-Of-Words* (CBOW)
  - O contexto é utilizado para prever uma palavra

Para pesquisar em casa: de onde surgiram esses nomes?

Mikolov et al. (2013)



CONTINOUS BAG-OF-WORDS (CBOW)



WORD2VEC

- Camada escondida também chamada de "camada de projeção"
- Número de dimensões (tamanho das *embeddings*) é determinado empiricamente, normalmente, e depende da tarefa em vista
- Não há dominância entre skip-gram e cbow nas tarefas
  - Cada caso é um caso e precisa ser analisado
- Variações
  - Sub-palavras: *FastText* (Bojanowski et al., 2016)
    - Tratamento de palavras fora do vocabulário e termos raros
  - Expressões multipalavra
    - Como perceber que o termo "Nova York" deve aparecer junto e não como duas palavras diferentes no treinamento?
  - Como lidar com sentenças, parágrafos e documentos inteiros? Há embeddings para eles?

EXEMPLOS

- Mikolov et al., 2013

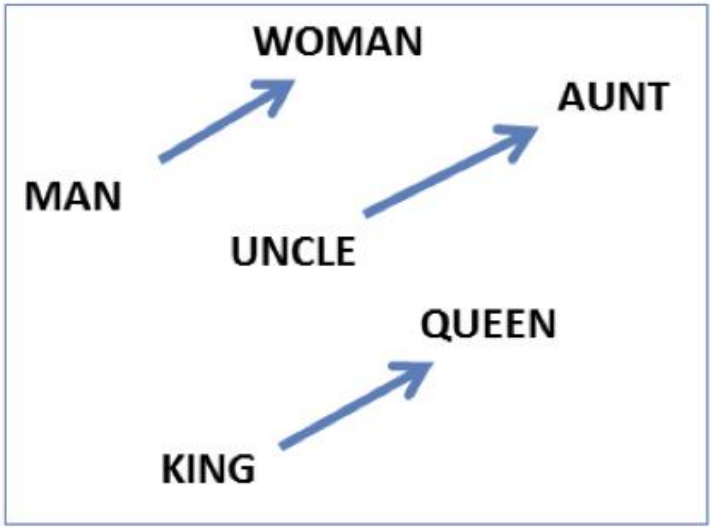
target:	Redmond	Havel	ninjutsu	graffiti	capitulate
	Redmond Wash.	Vaclav Havel	ninja	spray paint	capitulation
	Redmond Washington	president Vaclav Havel	martial arts	grafitti	capitulated
	Microsoft	Velvet Revolution	swordsmanship	taggers	capitulating

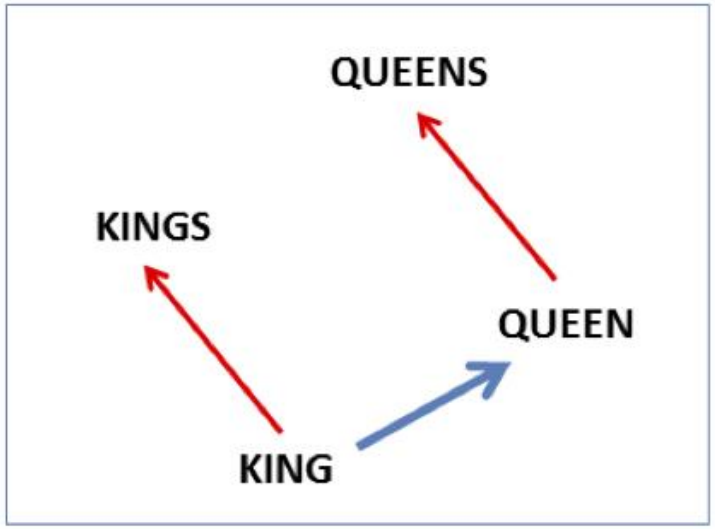
Exemplos famosos

- vector('king') - vector('man') + vector('woman') ≈ vector('queen')
- vector('Paris') - vector('France') + vector('Italy') ≈ vector('Rome')
- vector('Germany') + vector('capital') ≈ vector('Berlin')

Exemplos

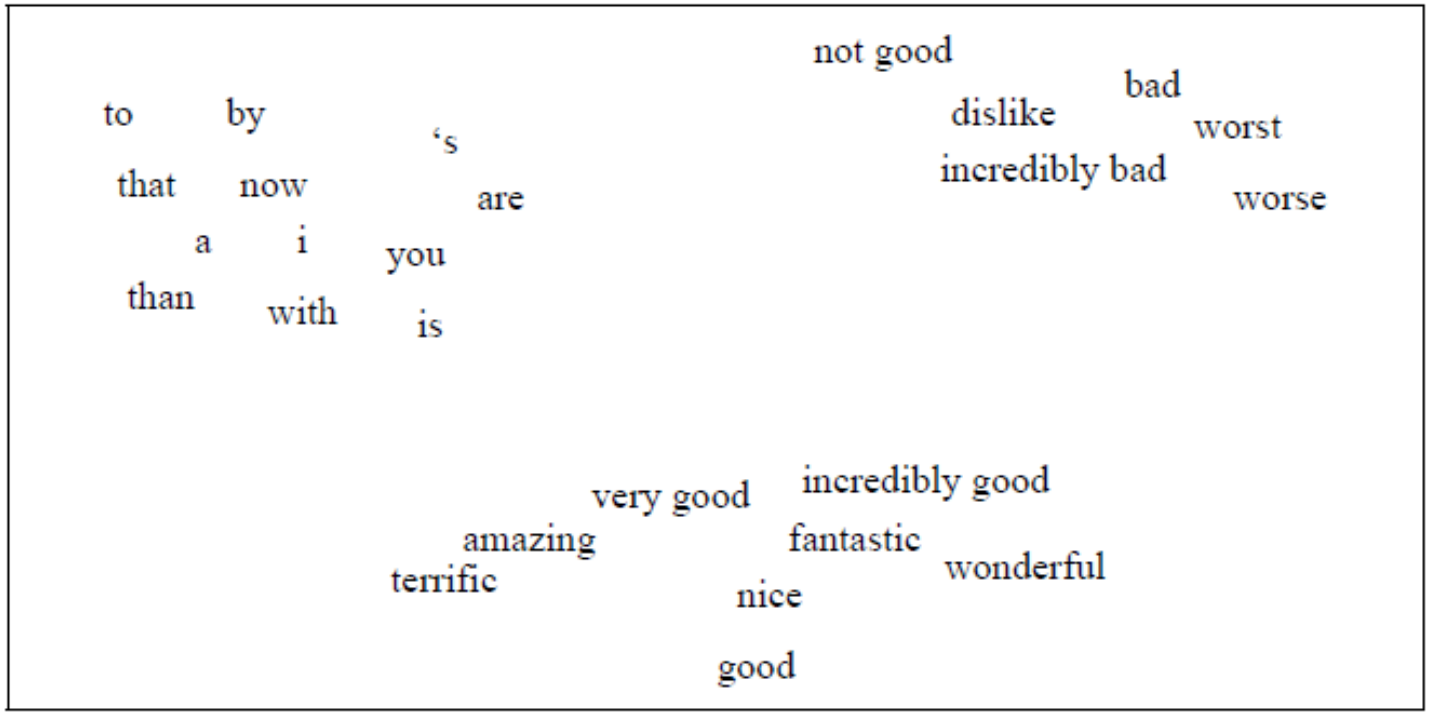
- Significado relacional





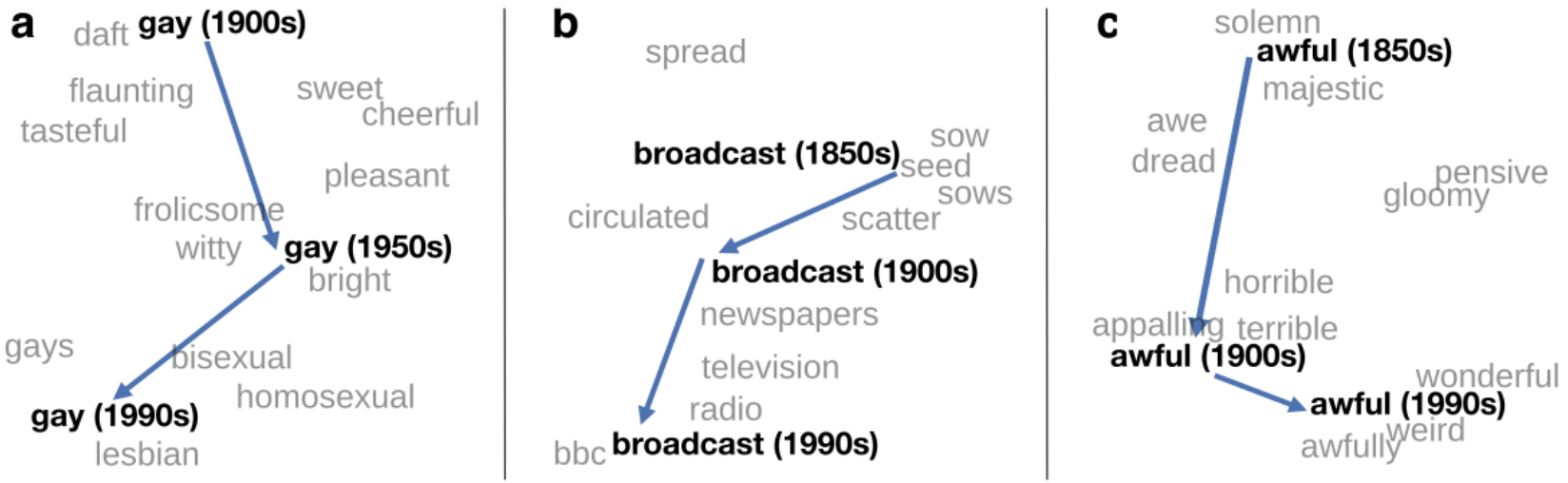
Exemplos

- Li et al., 2015
  - Espaço vetorial em uma tarefa de análise de sentimentos



Exemplos

- Mudanças históricas (Hamilton et al., 2016)



Exemplos

- Estereótipos e comportamentos sociais codificados nos vetores
  - Bolukbasi et al., (2016)
    - ‘computer programmer’ – ‘man’ + ‘woman’ = ?
    - ‘father’ → ‘doctor’ & ‘mother’ → ?

Exemplos

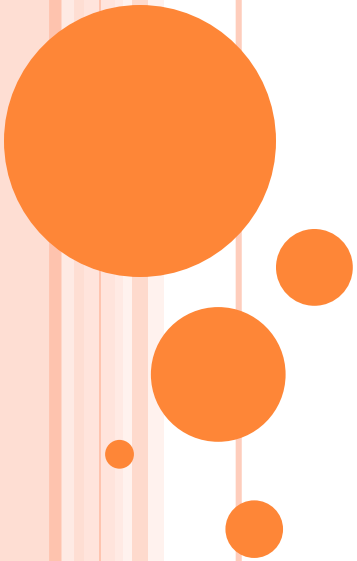
- Estereótipos e comportamentos sociais codificados nos vetores
  - Bolukbasi et al., (2016)
    - ‘computer programmer’ – ‘man’ + ‘woman’ = ‘homemaker’
    - ‘father’ → ‘doctor’ & ‘mother’ → ‘nurse’

Exemplos

- Estereótipos e comportamentos sociais codificados nos vetores
  - Caliskan et al. (2017)
    - Nomes americanos e europeus (‘Brad’, ‘Greg’, ‘Courtney’) relacionados a palavras boas
    - Nomes africanos (‘Leroy’ and ‘Shaniqua’) relacionados a palavras ruins

Limitações de modelos à la word2vec

- Não tem distinção para diferentes significados
  - "Manga" (fruta) e "manga" (de camisa) terão o mesmo vetor
- A semântica é implícita: não sabemos verdadeiramente qual o significado do termo
  - Temos apenas uma listagem de números
- Não há discriminação das relações diferentes que podem ocorrer entre os termos
- Só se aprende o que está nos dados
  - Para o bem e para o mal



BERT

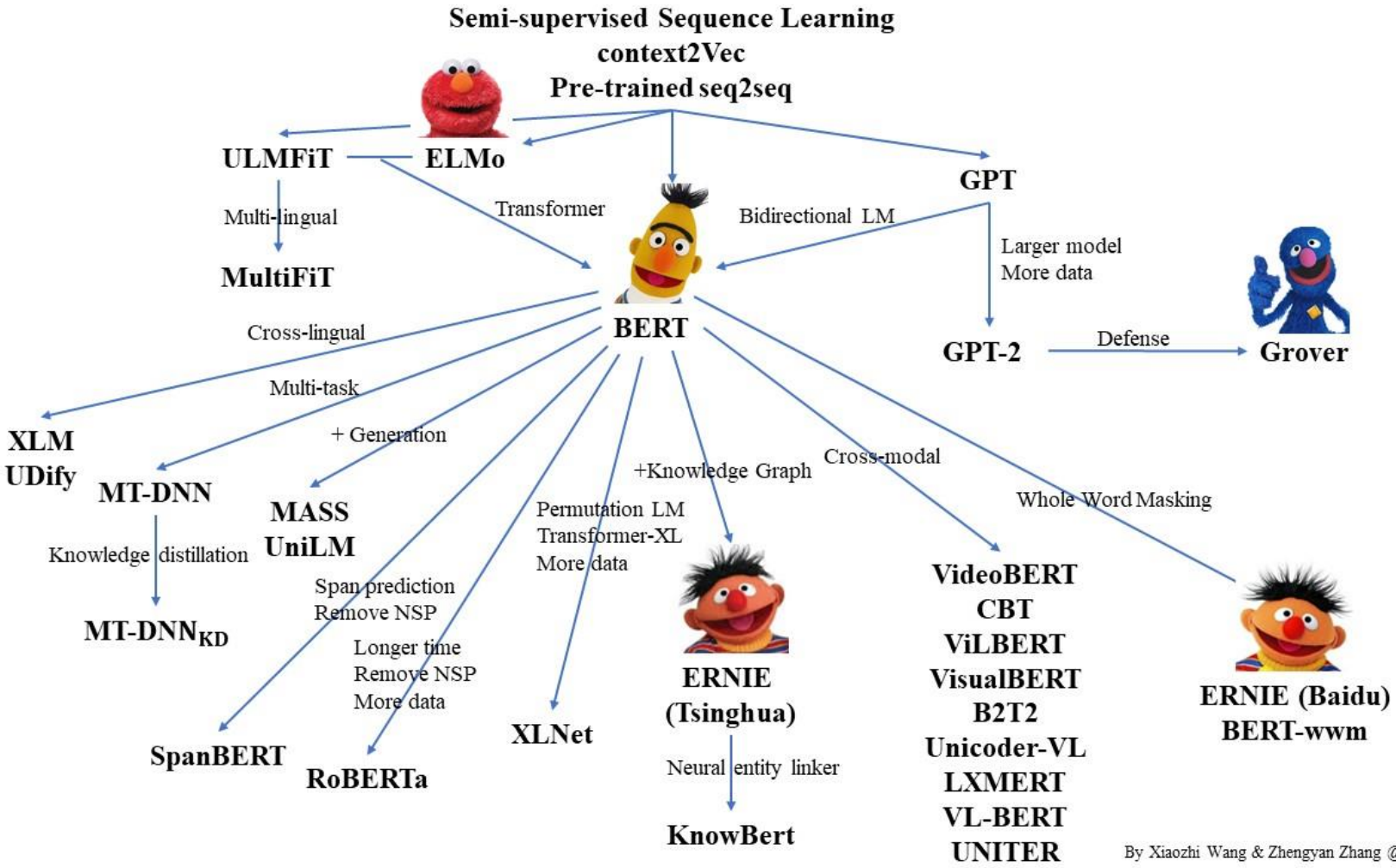
BERT

(Devlin et al., 2019)

- *Bidirectional Encoder Representations from Transformers*

- Diferencial
  - Solução "mais elegante" do que convoluções e recorrências
    - Atenção!
  - Computação paralelizável
    - Muito importante em função da quantidade de processamento e parâmetros envolvidos
  - *Embeddings* dinâmicos
  - Contexto levado em conta
    - O BERT é dito pertencer à família dos modelos "contextuais"

2020 NLP and NeurIPS Highlights



2020 NLP and NeurIPS Highlights

- *There was an explosion of new language models in 2020. The Huggingface transformers library now directly supports over 40 language model architectures, compared to 13 language models at the beginning of 2020. Besides minor improvements to benchmark scores, these language models demonstrate a creative variety of applications including tabular data understanding, cross-lingual understanding, style-controlled text generation, etc.*

- *The variety and volume of language models came with a much improved understanding of their workings, strengths, and shortcomings. However, the remarkable increase in traditional benchmark task performance seen in 2018 and 2019 was not repeated.*

  …

BERT & TRANSFORMER

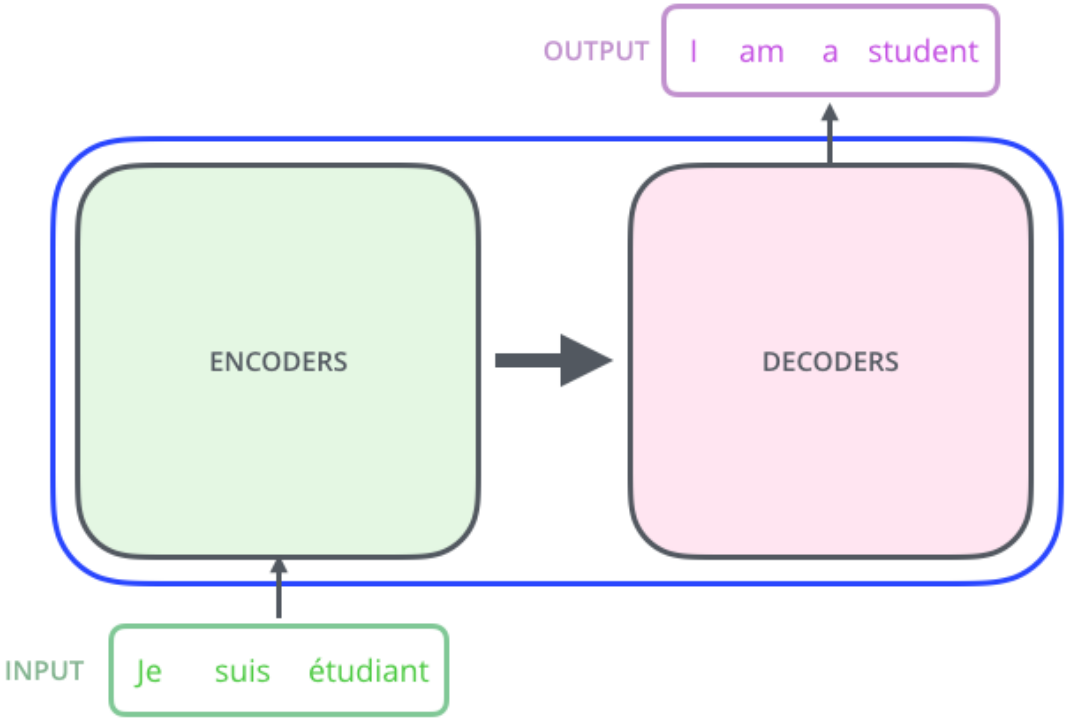
- Treinado com parte (modificada) da arquitetura do Transformer (Vaswani et al., 2017)
  - *Attention Is All You Need*

- Diversas tarefas de PLN produziram melhores resultados

- Indiretamente, criou uma área: "bertologia"
  - O que faz? Como faz? Por que faz?

Transformer

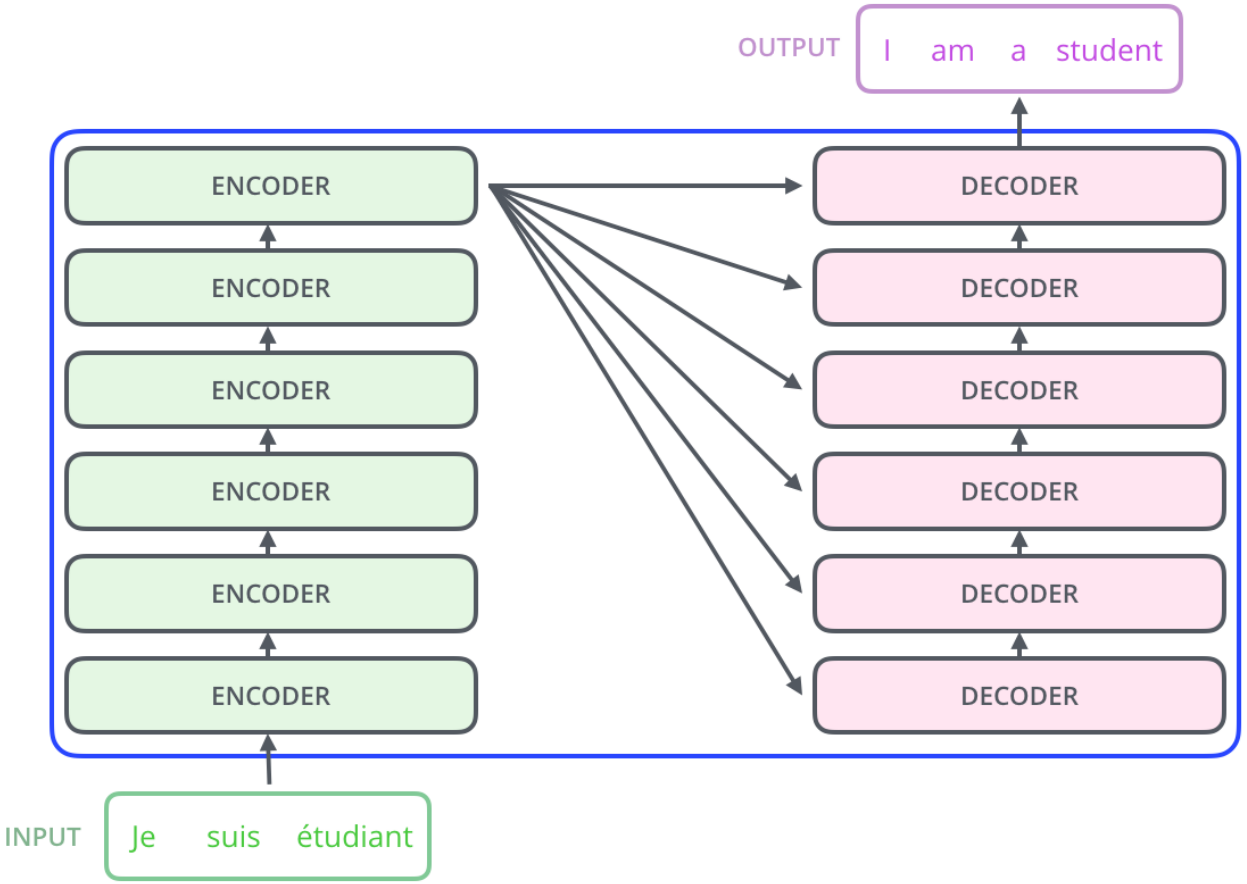
- Visão abrangente
  - Fez sua estreia na Tradução Automática



Fonte: *The Illustrated Transformer*

Transformer

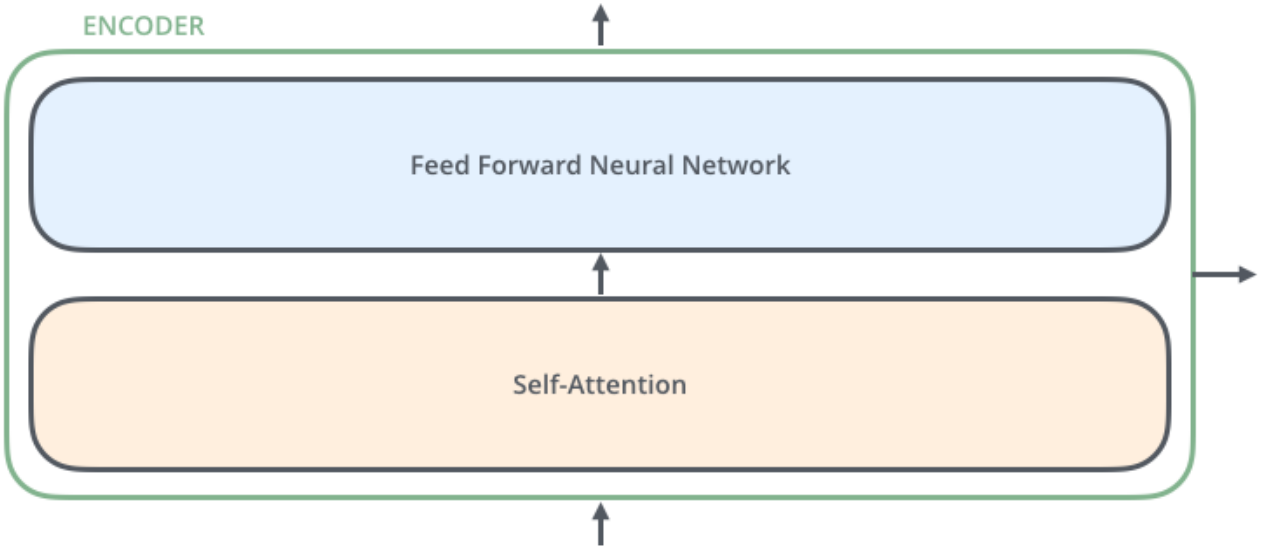
- Abrindo cada componente



Fonte: *The Illustrated Transformer*

Transformer

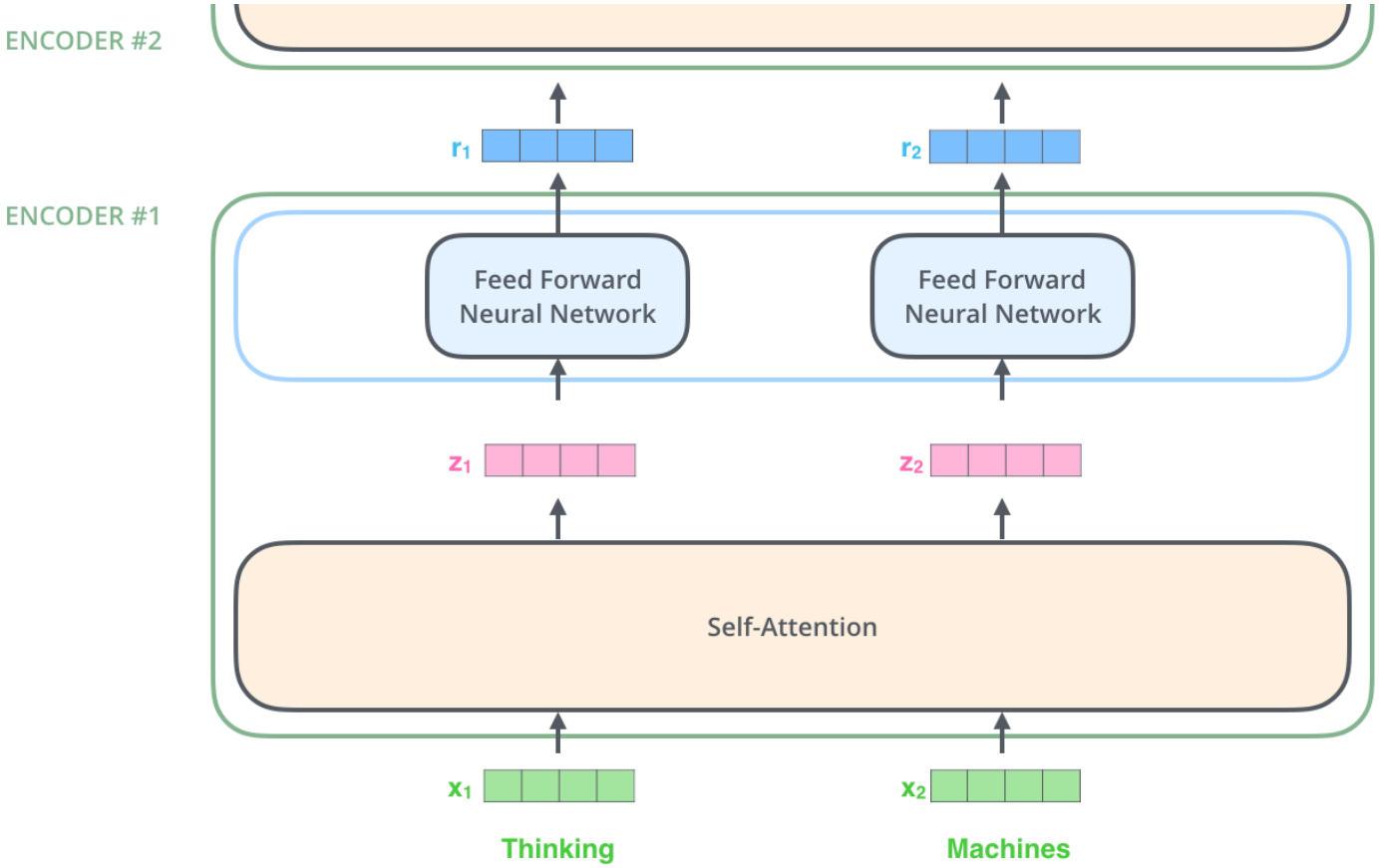
- Abrindo um “encoder”



Fonte: *The Illustrated Transformer*

Transformer

- Abrindo ainda mais os “encoders”

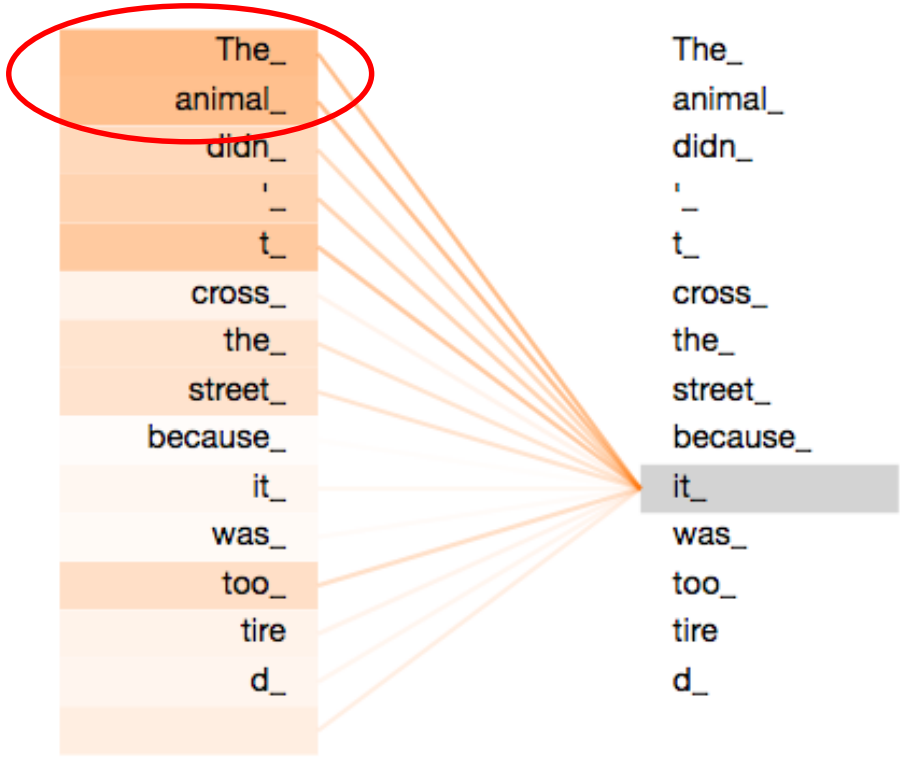


Fonte: *The Illustrated Transformer*

Transformer

- Computando “atenção” e determinando o que é mais relevante para o processamento

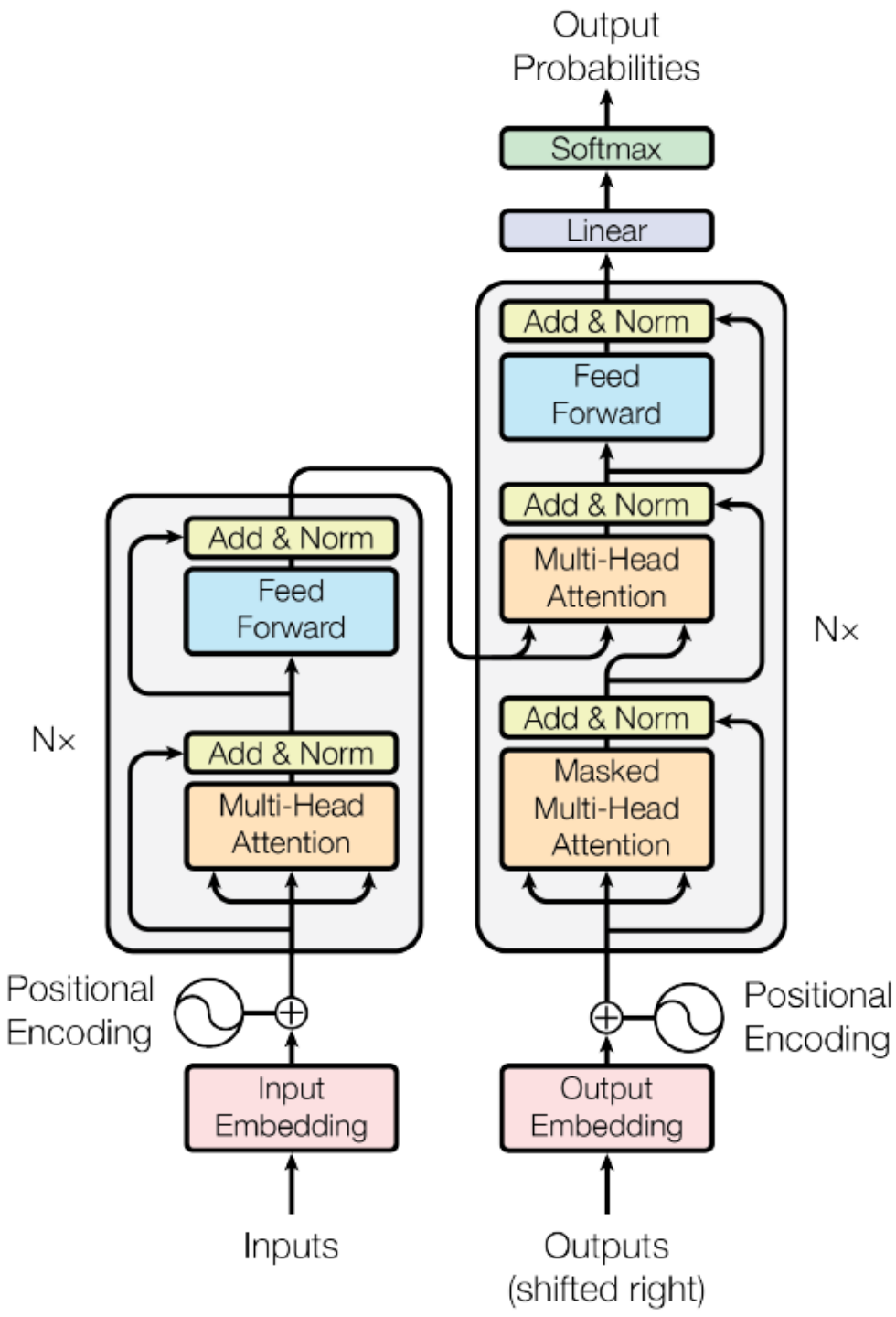
Procurando as palavras mais relevantes para o processamento de “it”



Fonte: *The Illustrated Transformer*

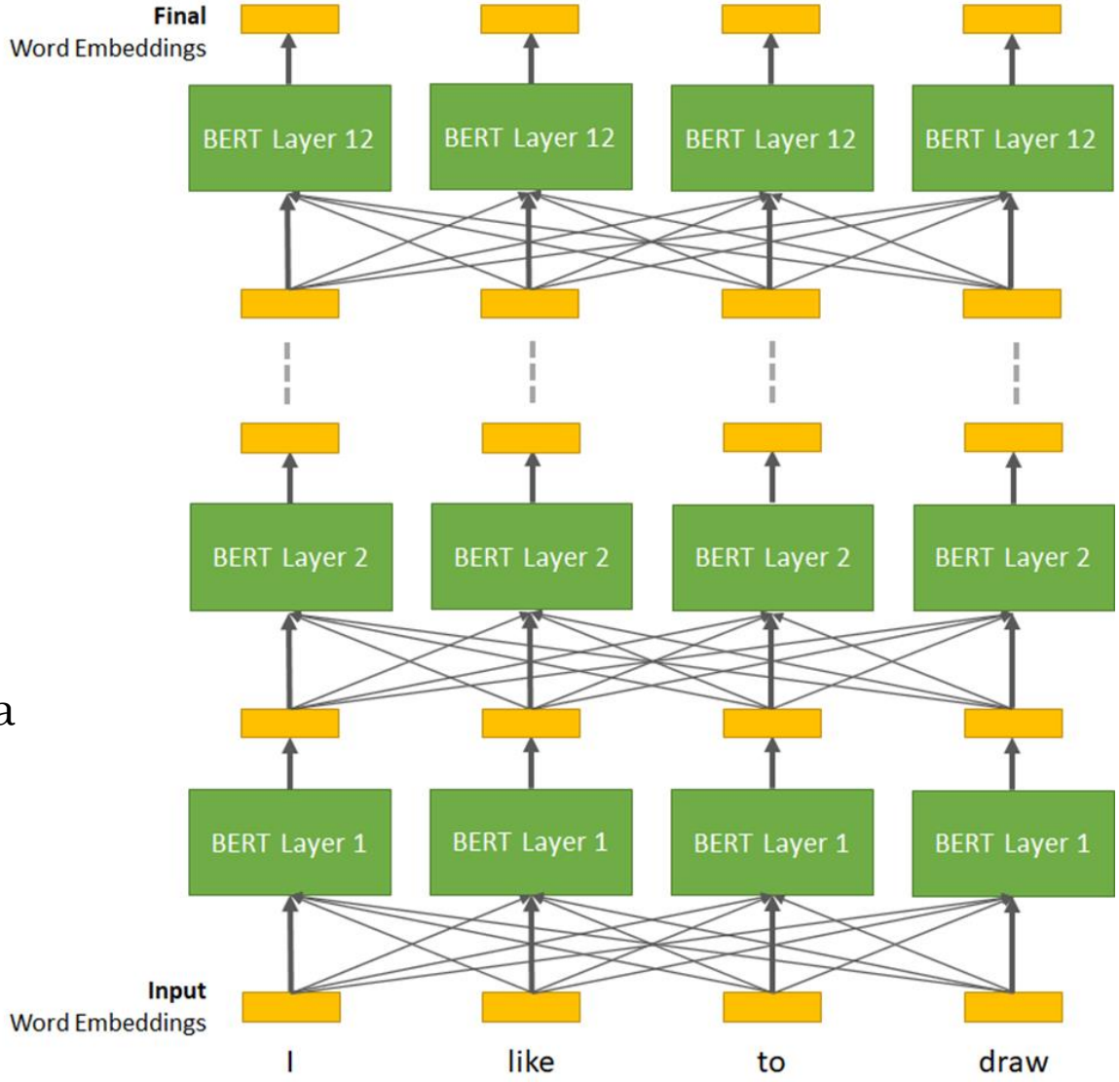
Transformer

- Visão detalhada
  - Decisões muitas vezes determinadas empiricamente
    - É um desafio justificar algumas delas ☺



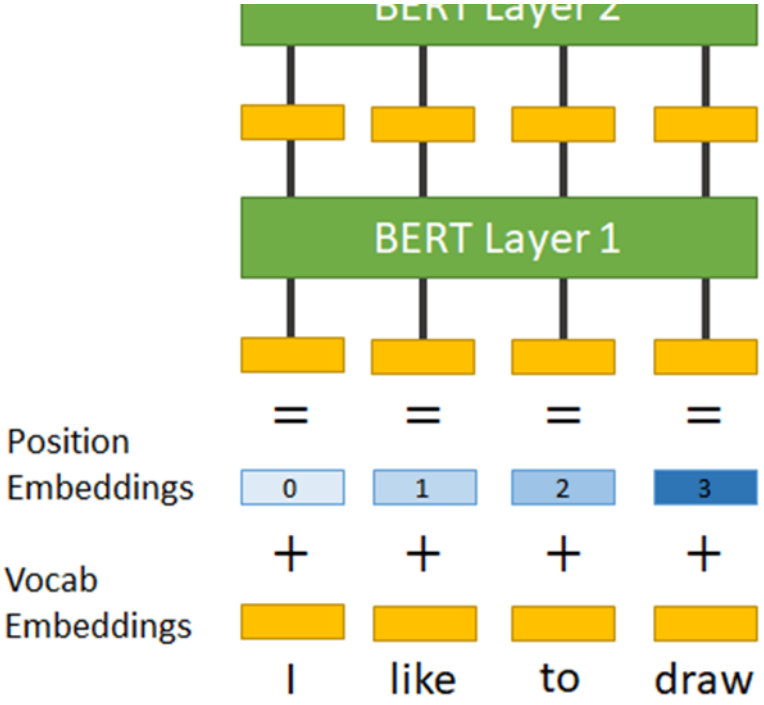
BERT

- 12 camadas de encoders na configuração mais comum
  - Cada elemento de cada camada pode ser executado em paralelo aos demais
  - Cada elemento recebe como entrada todas as palavras
    - E surge o contexto!



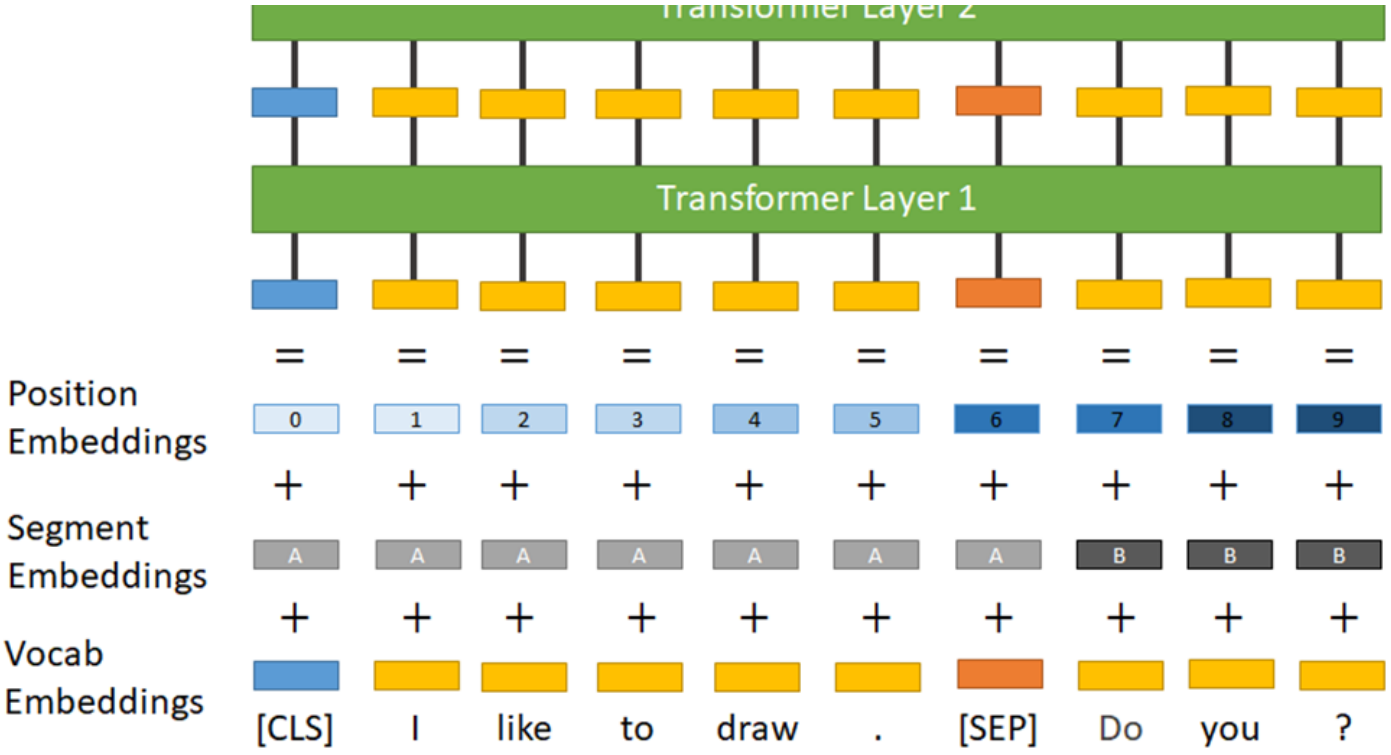
BERT

- Em mais detalhes
  - Considerando as posições dos tokens: informação importante!



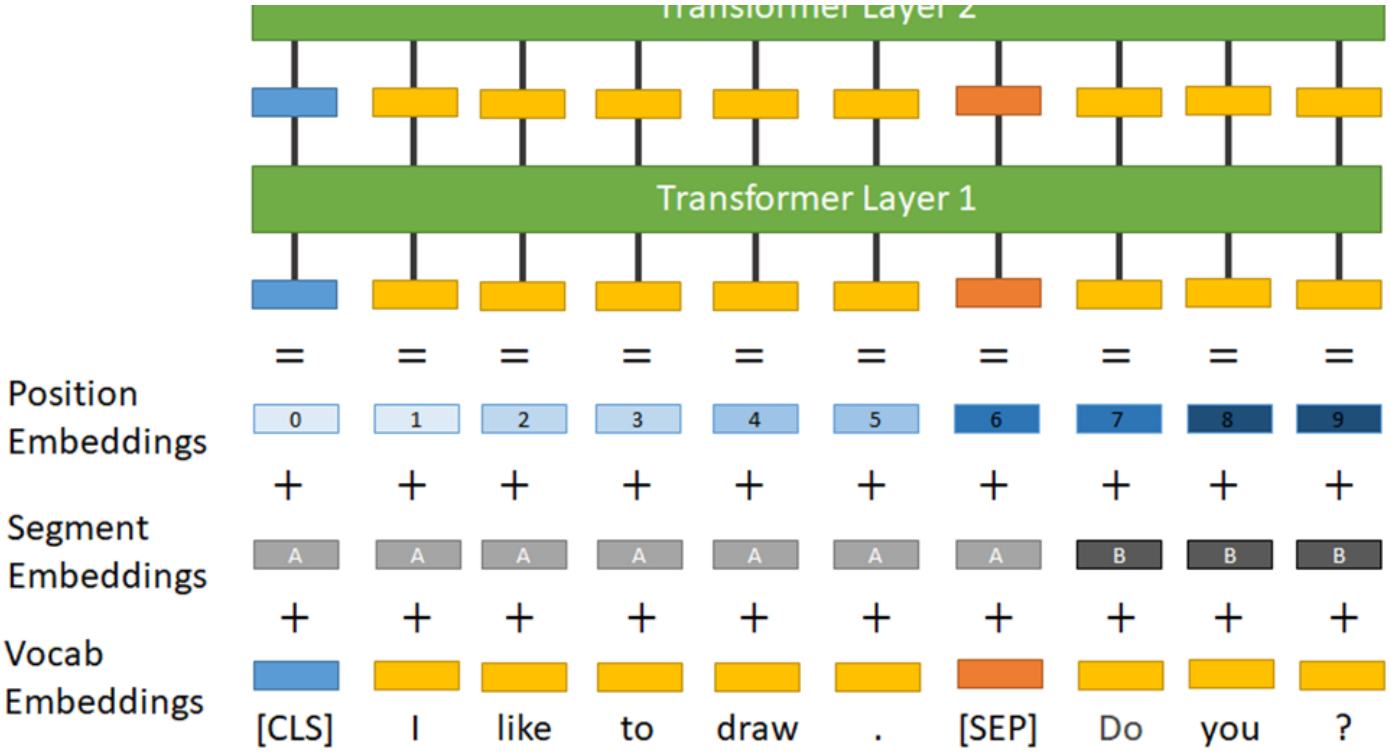
BERT

- Considerando pares de segmentos
  - Indicação dos segmentos das palavras
  - Tokens especiais: [CLS] e [SEP]



BERT

- Em mais detalhes
  - Tarefas "fake" utilizadas para treinamento
    - *Masked Language Model* (MLM)
    - *Next Sentence Prediction* (NSP)

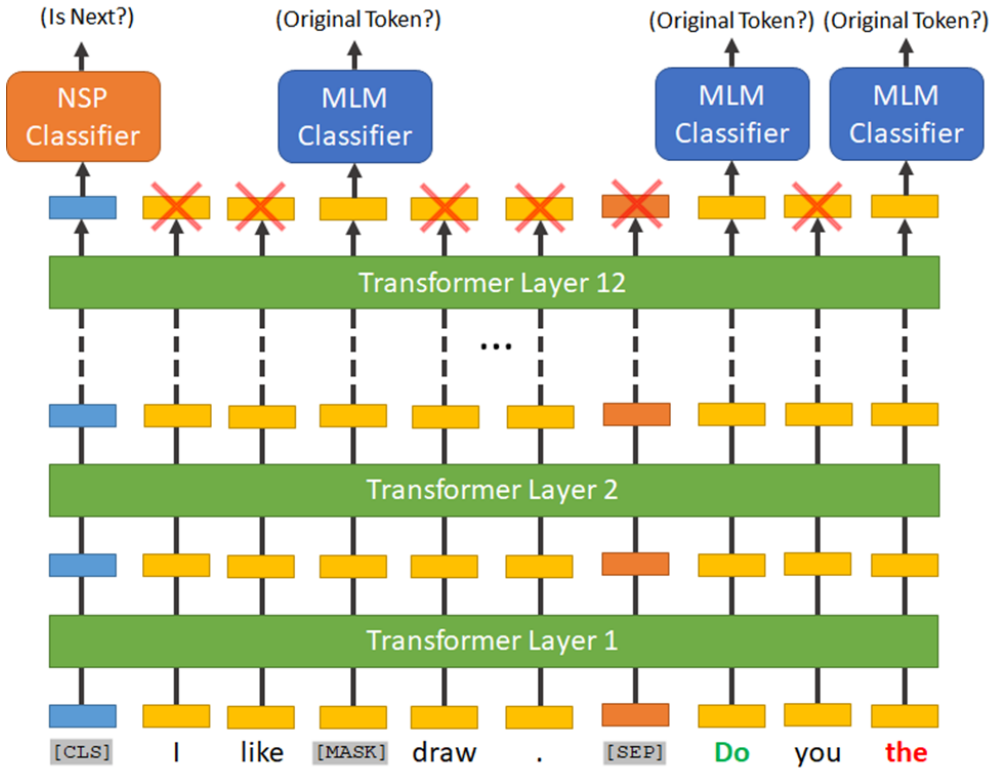


BERT

- Em mais detalhes
  - Tarefas "fake" utilizadas para treinamento
    - *Masked Language Model* (MLM)
    - *Next Sentence Prediction* (NSP)

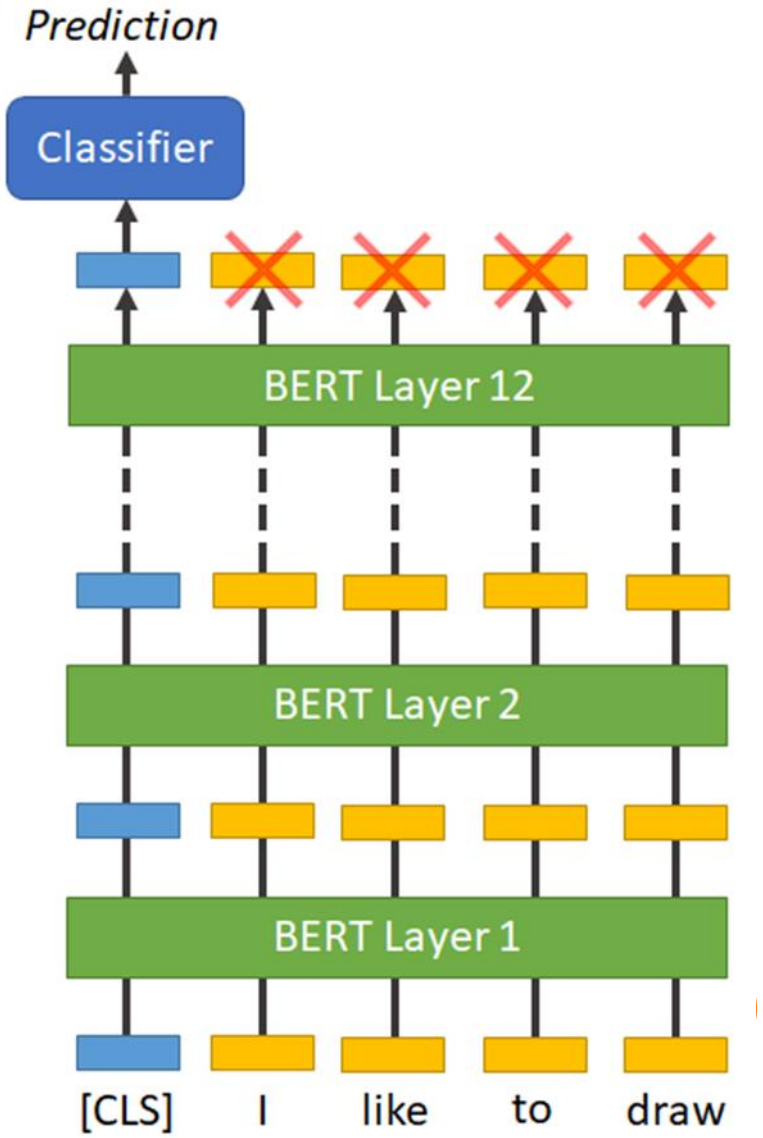
MLM: 15% dos tokens são "mascarados"

- 80% escondidos → [MASK]
- 10% trocados por token aleatório
- 10% mantidos



BERT

- *Embedding* do token [CLS] normalmente utilizado como *embedding* de todo o conteúdo apresentado, para diversas tarefas, e não apenas NSP



Detalhes

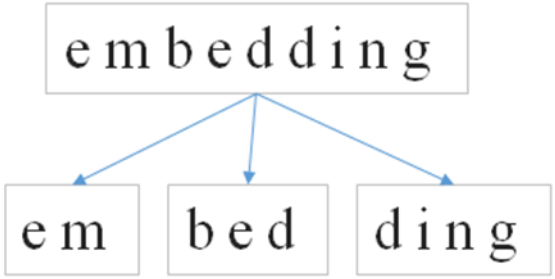
- Entrada padrão: sequências de 512 tokens
  - Se mais tokens, pode-se truncar
  - Se menos, *padding*

- $\text{BERT}_{\text{BASE}}$
  - 12 camadas
  - *Embeddings* de 768 dimensões
  - ~110 milhões de parâmetros

- $\text{BERT}_{\text{LARGE}}$
  - 24 camadas
  - *Embeddings* de 1024 dimensões
  - ~240 milhões de parâmetros

Detalhes

- Tokenização: segmentos menores do que palavras
  - WordPiece (Wu et al., 2016)
    - Interessante para lidar com palavras fora do vocabulário e auxiliar na generalização do modelo

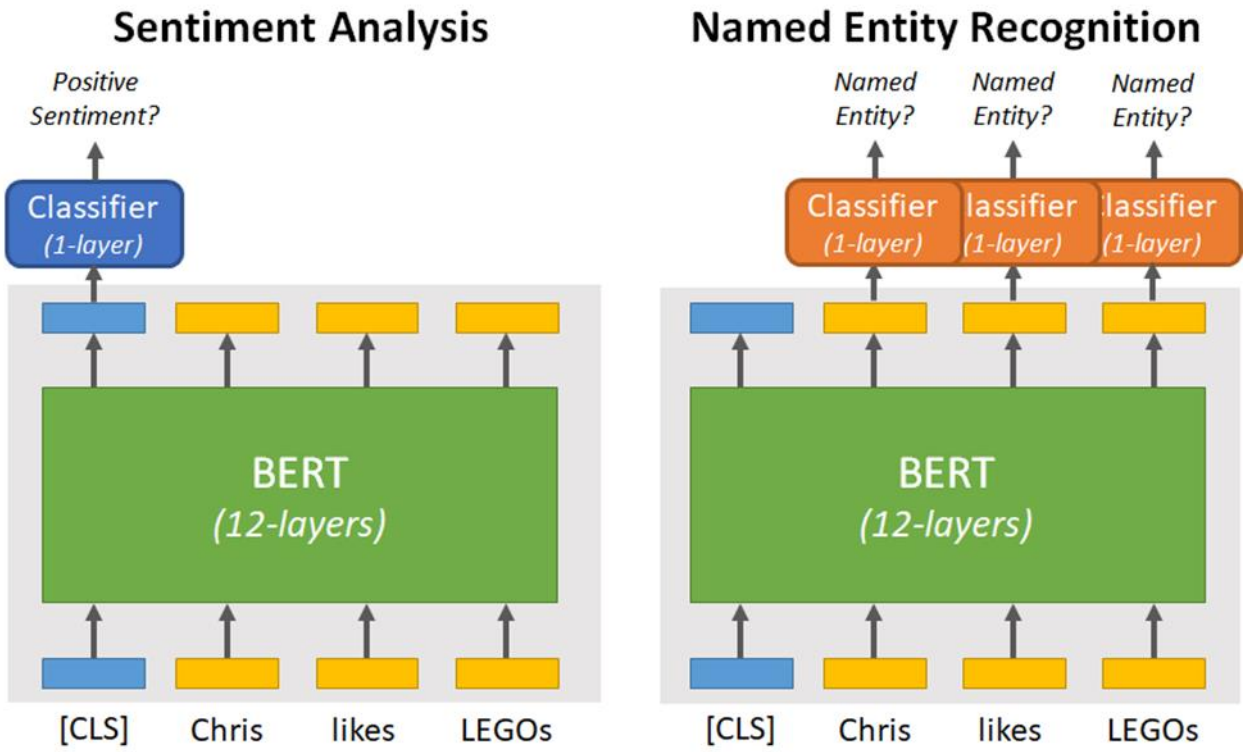


BERT e aprendizado

- As possibilidades de *transfer learning*
  - Etapas
    - *Pre-training* (normalmente pronto, utilizando-se as bases disponíveis)
    - *Fine-tuning*
      - Camadas adicionais para a tarefa de interesse
  - Vantagens
    - Desenvolvimento mais rápido e barato
    - Menos dados necessários
    - Resultados melhores, normalmente

BERT e aprendizado

- As possibilidades de *transfer learning*



Avanços significativos

- Diversas tarefas (Devlin et al., 2019)

System	Dev		Test	
	EM	F1	EM	F1
Top Leaderboard Systems (Dec 10th, 2018)				
Human	-	-	82.3	91.2
#1 Ensemble - nlnet	-	-	86.0	91.7
#2 Ensemble - QANet	-	-	84.5	90.5
Published				
BiDAF+ELMo (Single)	-	85.6	-	85.8
R.M. Reader (Ensemble)	81.2	87.9	82.3	88.5
Ours				
$\text{BERT}_{\text{BASE}}$ (Single)	80.8	88.5	-	-
$\text{BERT}_{\text{LARGE}}$ (Single)	84.1	90.9	-	-
$\text{BERT}_{\text{LARGE}}$ (Ensemble)	85.8	91.8	-	-
$\text{BERT}_{\text{LARGE}}$ (Sgl.+TriviaQA)	**84.2**	**91.1**	**85.1**	**91.8**
$\text{BERT}_{\text{LARGE}}$ (Ens.+TriviaQA)	**86.2**	**92.2**	**87.4**	**93.2**

Stanford Question Answering Dataset

Avanços significativos

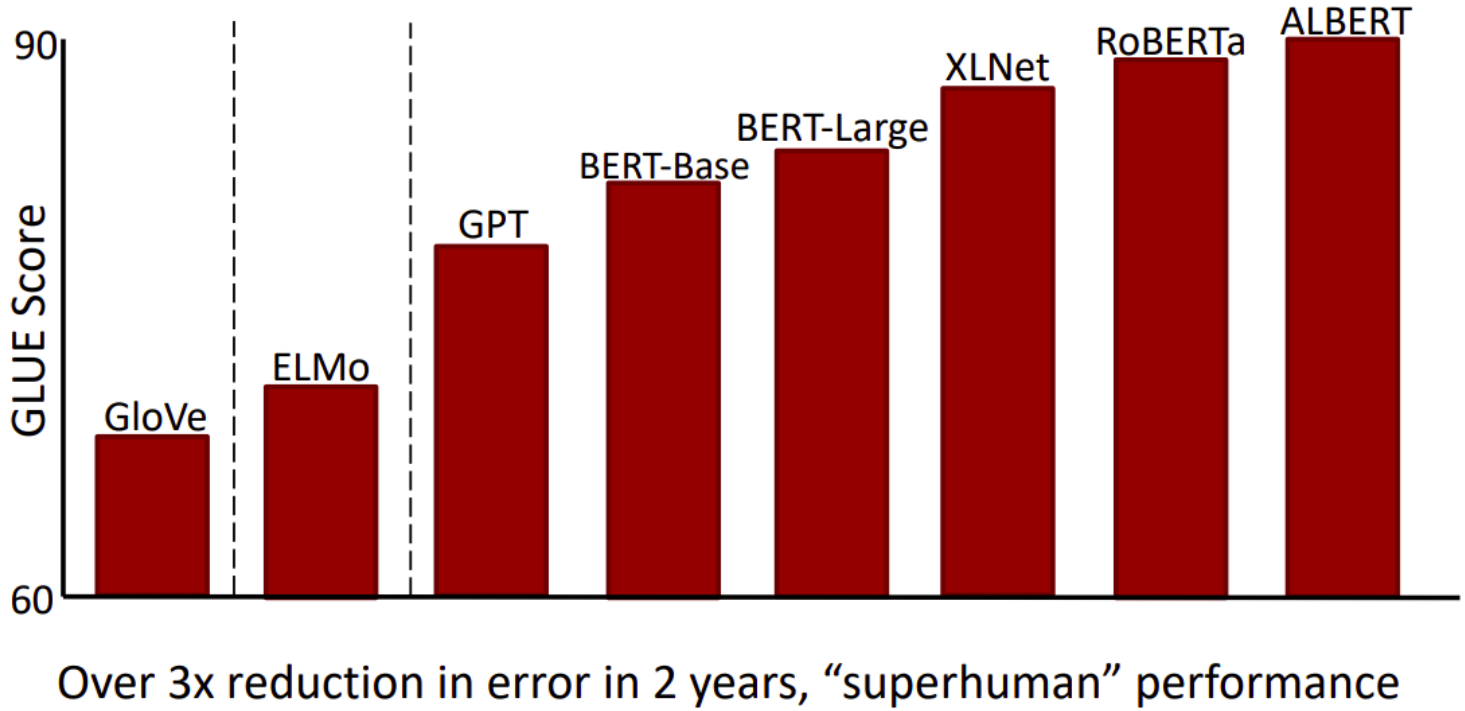
- Diversas tarefas (Devlin et al., 2019)

System	Dev	Test
ESIM+GloVe	51.9	52.7
ESIM+ELMo	59.1	59.2
OpenAI GPT	-	78.0
$\text{BERT}_{\text{BASE}}$	81.6	-
$\text{BERT}_{\text{LARGE}}$	**86.6**	**86.3**
Human (expert)†	-	85.0
Human (5 annotations)†	-	88.0

Situations With Adversarial Generations

Avanços significativos

- Ao longo do tempo: GLUE (*General Language Understanding Evaluation*) (Wang et al., 2019)



Limitações de modelos à la BERT

- A semântica ainda é implícita: não sabemos verdadeiramente qual o significado do termo
- Também não há discriminação das relações diferentes que podem ocorrer entre os termos
- Ainda só se aprende o que está nos dados
  - A semântica ainda é limitada!

(a leitura da semana é bombástica ☺)

O QUE O FUTURO NOS RESERVA?

- Nesta frente, é difícil prever
  - Há modelos especializados para tarefas particulares
  - Há tentativas de simplificação dos modelos (menos parâmetros!) sem comprometer a qualidade geral
  - A evolução tem sido rápida!
    - A área e o perfil dos trabalhos têm mudado radicalmente nos últimos anos

Tarefas da semana

Leitura

- Bender, E.M. and Koller, A. (2020). Climbing towards NLU: On Meaning, Form, and Understanding in the Age of Data. In the Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 5185-5198.
  - No e-Disciplinas

Provinha 6 disponível à tarde no e-Disciplinas