

# Deep Learning para PLN



**Prof. Dr. Diego Renan Bruno**

Education Tech Lead na DIO

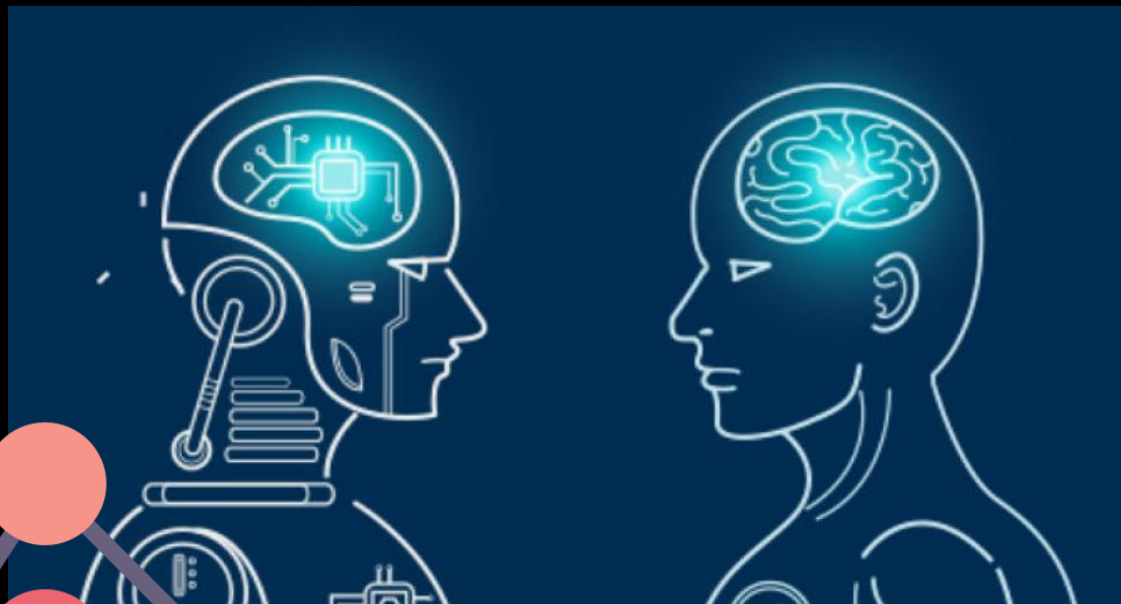
Doutor em Robótica e *Machine Learning* pelo ICMC-USP



OpenCV

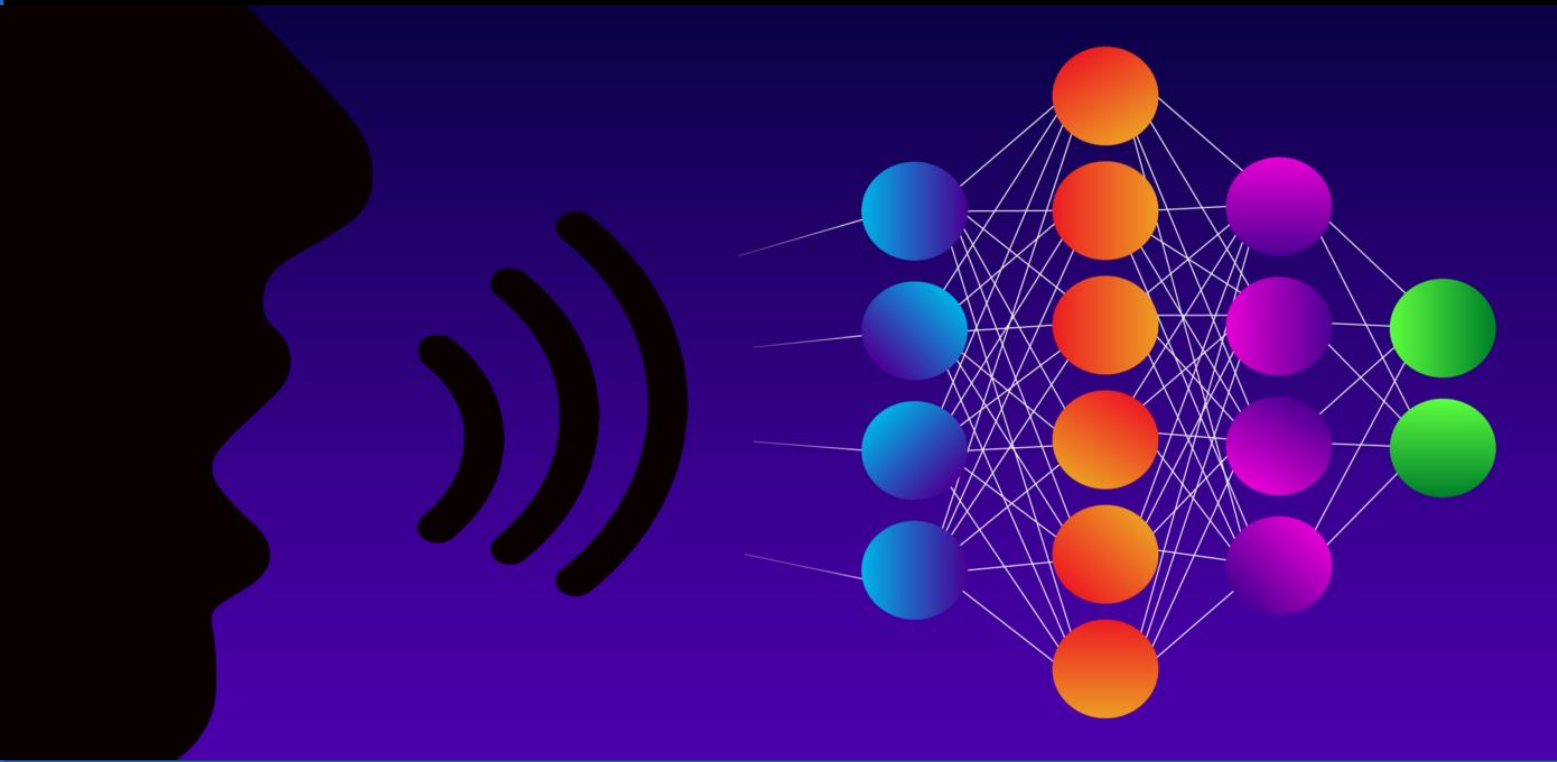
# Deep Learning para PLN

*Machine Learning*



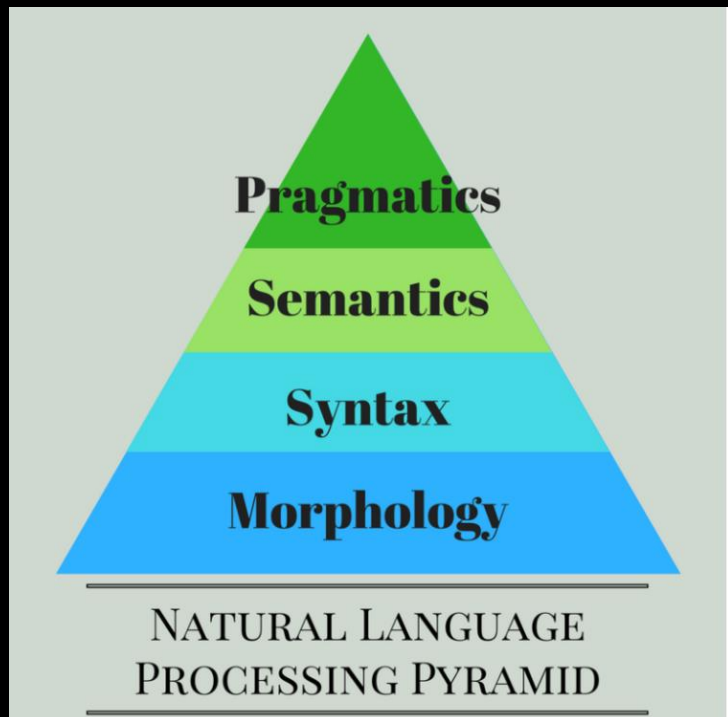
# Deep Learning para PLN

→ Sistema de interpretação de linguagem natural.



# Níveis do processamento

→ Sistemas: recomendação, comando por voz, chatbots...



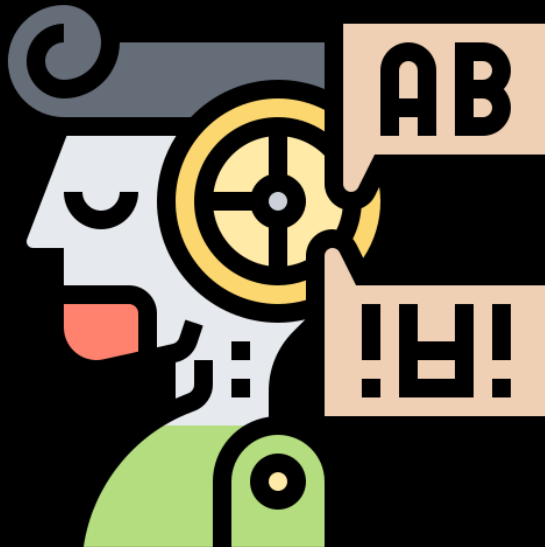
**Pragmática** é o ramo da linguística que estuda a linguagem no contexto de seu uso na comunicação.

Relacionado com à análise detalhada da composição, derivação, flexão das palavras e de seus processos de formação.

# Deep Learning para PLN

Os sistemas de NLP permitem que a **tecnologia usada não apenas entenda o significado** literal de cada palavra que está sendo dita, como também considere aspectos como:

- Contexto da conversa;
- Significados sintáticos e semânticos;
- Interprete os textos;
- Análise sentimentos e mais.



# Tipos de redes para PLN

## *Redes de Deep Learning:*

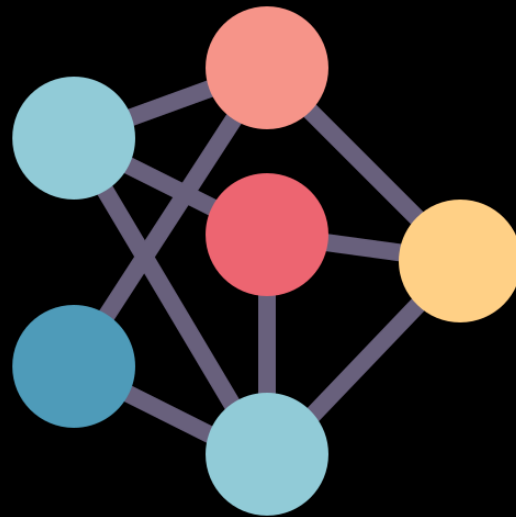
*Os primeiros modelos de linguagem usavam arquiteturas NN feedforward ou NN convolucional, mas elas não capturavam muito bem o contexto. Contexto é como uma palavra ocorre em relação às palavras circundantes na frase. Para capturar o contexto, foram aplicados NNs recorrentes.*



# Tipos de redes para PLN

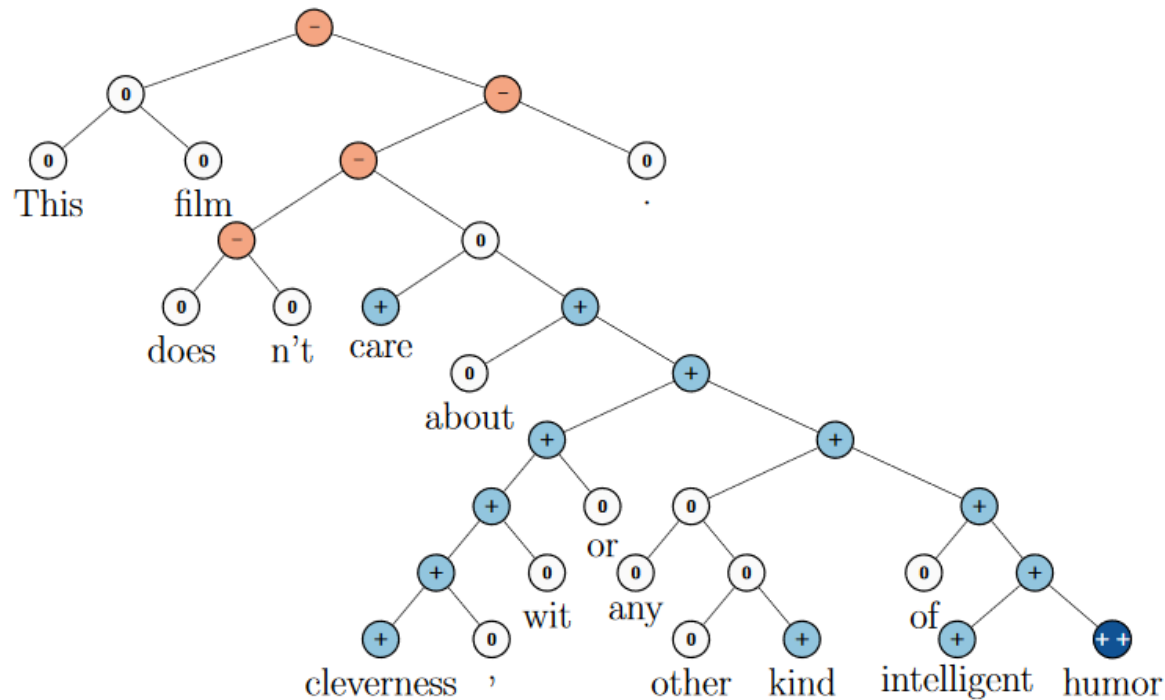
## *Redes de Deep Learning:*

*O LSTM, uma variante do RNN, foi então usado para capturar o contexto de longa distância. O LSTM bidirecional (BiLSTM) melhora o LSTM ao observar as sequências de palavras nas direções para frente e para trás*



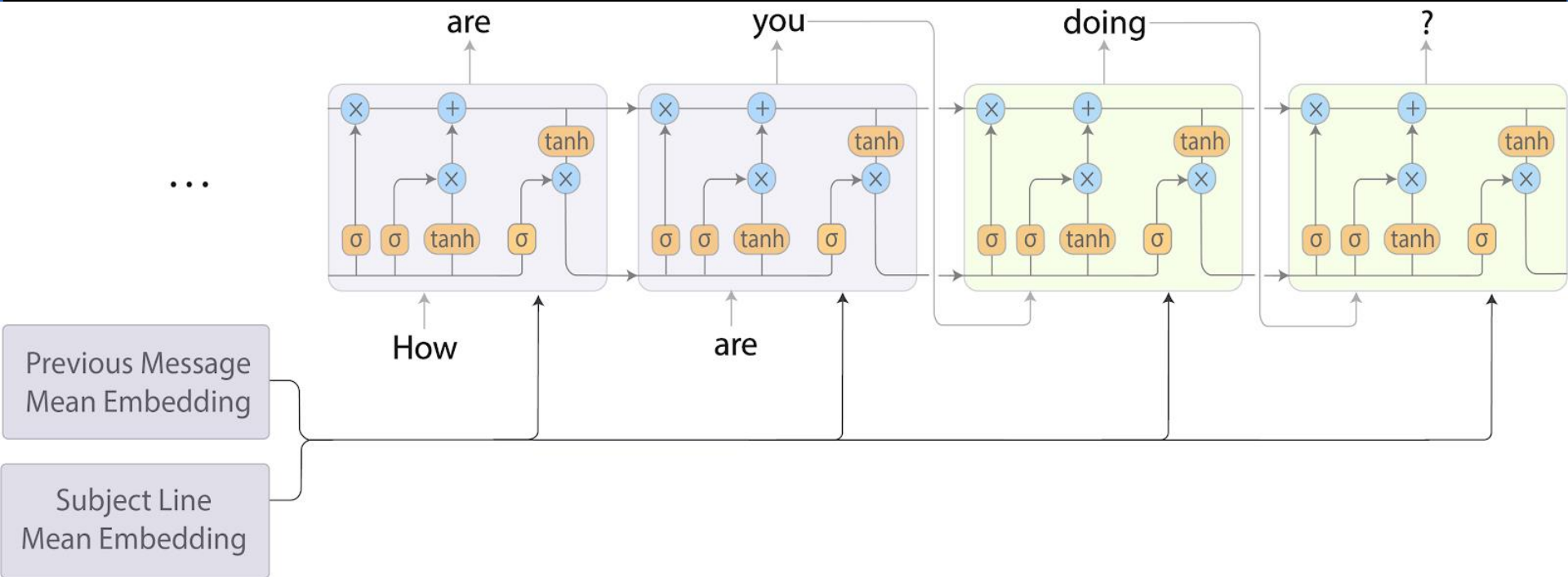
# Exemplos do mundo real em PLN?

O Google substituiu seu sistema de tradução baseado em frases pela Neural Machine Translation (NMT). Isso reduz os erros de tradução em 60%. Ele usa uma rede LSTM profunda com 8 camadas de codificador e 8 de decodificador.



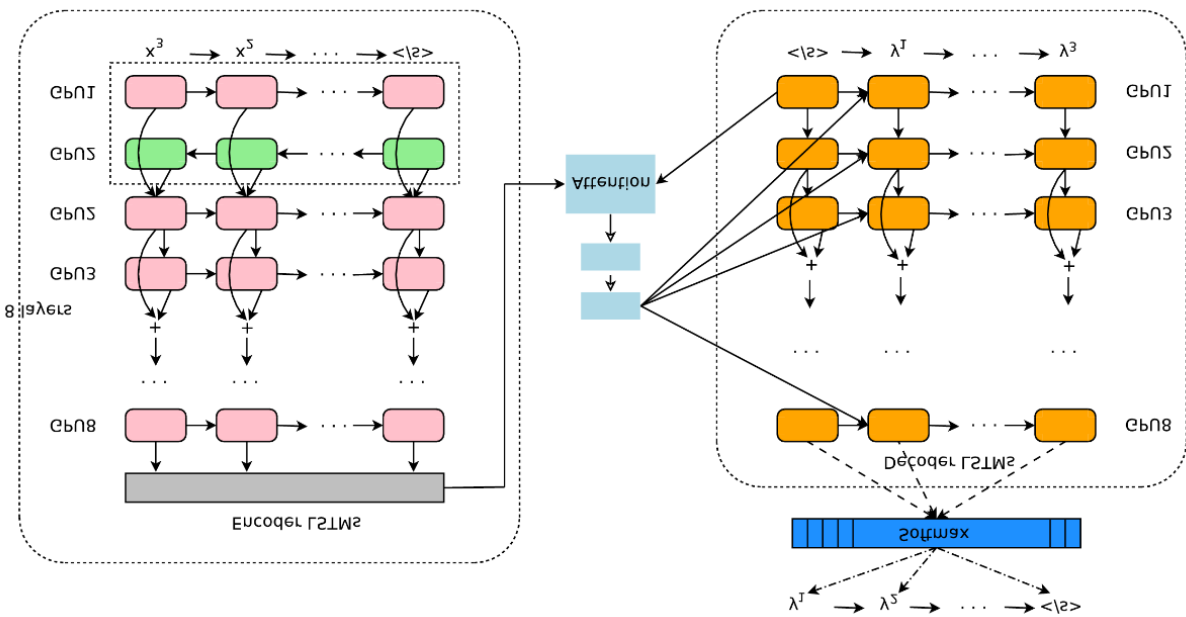


# Exemplos do mundo real em PLN?



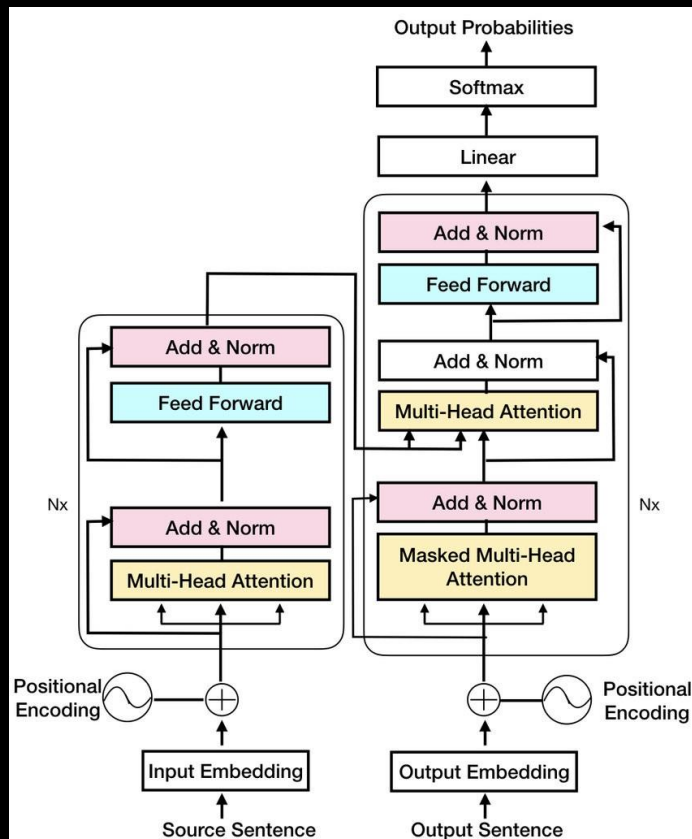
# Exemplos do mundo real em PLN?

O Google substituiu seu sistema de tradução baseado em frases pela Neural Machine Translation (NMT). Isso reduz os erros de tradução em 60%. Ele usa uma rede LSTM profunda com 8 camadas de codificador e 8 de decodificador.



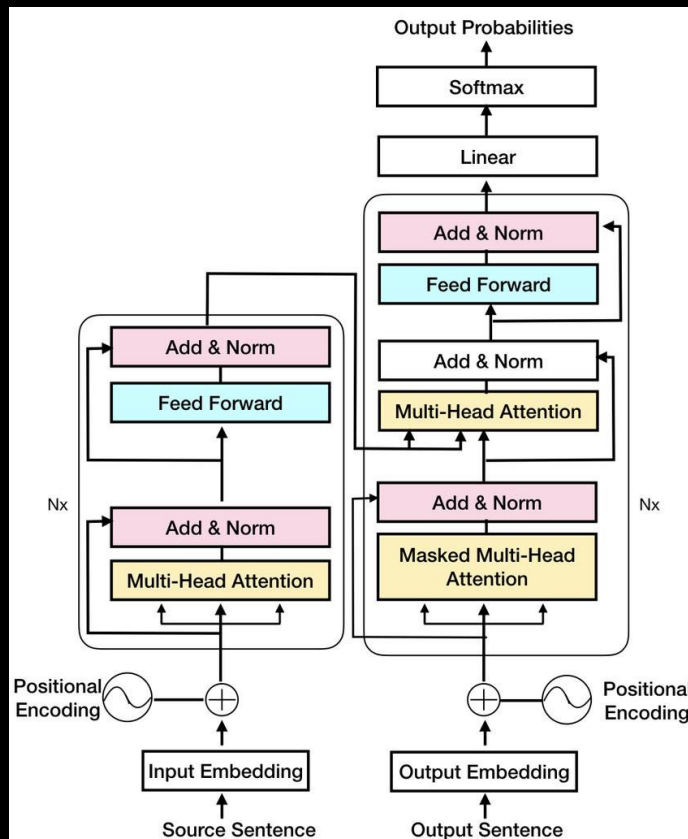
# Exemplos do mundo real em PLN?

A revolução na área de NLP com Deep Learning teve início em 2018 com o lançamento dos modelos de linguagem pré-treinados ELMo e ULMFiT. Mas, foi a proposta de uma nova arquitetura de redes neurais, denominada Transformer, baseada unicamente em mecanismos de atenção, que mudaria para sempre as pesquisas nessa área.



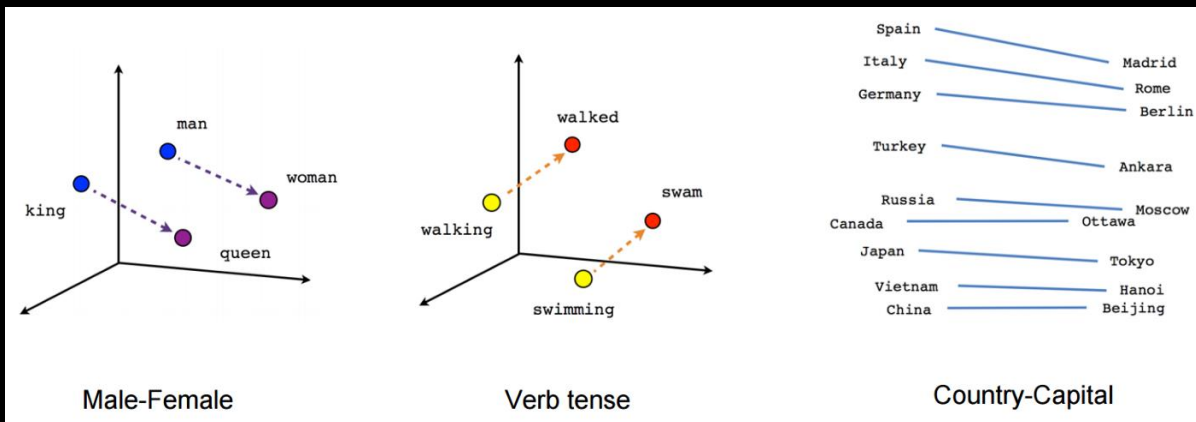
# Exemplos do mundo real em PLN?

A arquitetura Transformer permitiu que o treinamento fosse realizado com um volume muito maior de dados do que era possível antes. Isso levou ao desenvolvimento de modelos de linguagem pré-treinados, que são previamente treinados e, posteriormente, são submetidos a um treinamento com um ajuste fino (fine-tuning) nas tarefas específicas de linguagem.



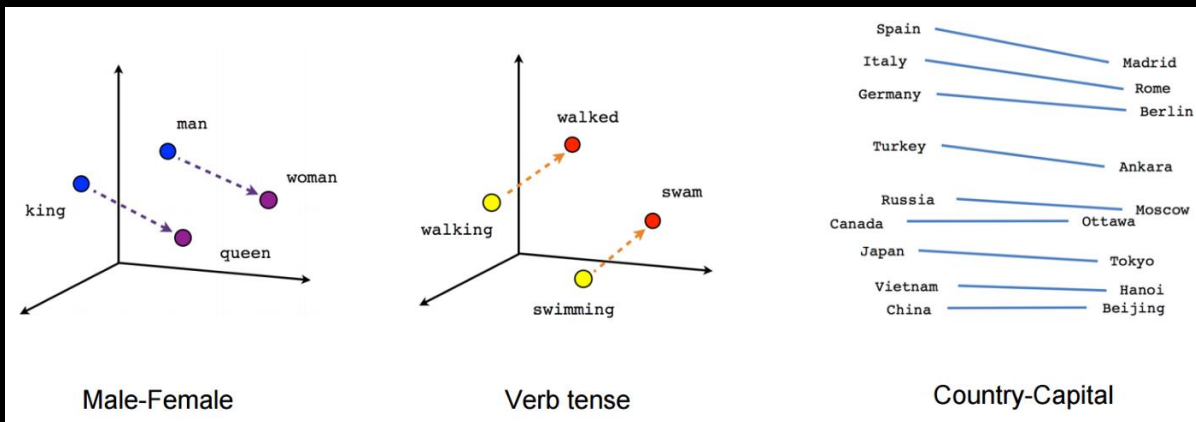
# Mas como isso é possível?

Os **word embeddings** são representações vetoriais das palavras, que permitem capturar o contexto e relacionamento das palavras nos documentos (Figura 1), sem a necessidade de realizar engenharia de features com anotações exaustivas nas sentenças.



# Mas como isso é possível?

Os **word embeddings** são representações vetoriais das palavras, que permitem capturar o contexto e relacionamento das palavras nos documentos (Figura 1), sem a necessidade de realizar engenharia de features com anotações exaustivas nas sentenças.



# Conclusões

O momento **ImageNet, em 2012**, marcou o início de um enorme interesse de pesquisadores e empresas no mundo todo por Deep Learning. Já o ano de 2018 determinou o início da revolução da área de NLP com os modelos de linguagem pré-treinados, como **ELMo, GPT e BERT**, que produziram avanços significativos em várias tarefas de linguagem natural, tais como inferência, análise de sentimento e tradução de linguagem, em um curto espaço de tempo.

# Obrigado!

*Machine Learning*

Prof. Dr. Diego Bruno



OpenCV