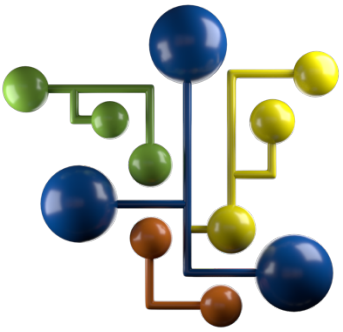

Data Science Academy

www.datascienceacademy.com.br

Processamento de Linguagem Natural

O Problema da Dissipação do Gradiente

O Problema de Degradação ou Dissipação do Gradiente (Vanishing Gradient) é uma dificuldade encontrada no treinamento de certas Redes Neurais Recorrentes com métodos baseados em gradientes (como por exemplo o Backpropagation). Em particular, esse problema torna muito difícil aprender e ajustar os parâmetros das camadas anteriores na rede. Esse problema fica pior à medida que o número de camadas na arquitetura aumenta.

Este não é um problema fundamental com as redes neurais - é um problema com os métodos de aprendizagem baseados em gradientes causados por certas funções de ativação. Procuremos compreender intuitivamente o problema e a causa por trás disso.

Problema

Os métodos baseados em gradientes aprendem o valor de um parâmetro ao entender como uma pequena alteração no valor do parâmetro afetará a saída da rede. Se uma alteração no valor do parâmetro provoca uma alteração muito pequena na saída da rede - a rede simplesmente não consegue aprender o parâmetro efetivamente, o que é um problema.

Isso é exatamente o que acontece no problema da degradação do gradiente - os gradientes da saída da rede em relação aos parâmetros nas camadas iniciais tornam-se extremamente pequenos. Essa é uma maneira elegante de dizer que mesmo uma grande mudança no valor dos parâmetros para as camadas iniciais não tem um grande efeito na saída. Vamos tentar entender quando e porque esse problema acontece.

Causa

O problema de degradação do gradiente depende da escolha da função de ativação. Muitas funções de ativação comuns (por exemplo, sigmoid ou tanh) "espalham" sua entrada em uma faixa de saída muito pequena de uma maneira muito não-linear. Por exemplo, sigmoid mapeia a linha do número real para um intervalo "pequeno" de [0, 1]. Como resultado, existem grandes regiões do espaço de entrada que são mapeadas para uma faixa extremamente pequena. Nessas regiões do espaço de entrada, mesmo uma grande mudança na entrada, produzirá uma pequena alteração na saída - daí o gradiente é pequeno.

Isso se torna muito pior quando empilhamos várias camadas de tais não-linearidades uma sobre a outra. Por exemplo, a primeira camada mapeará uma região de entrada grande para uma região de saída menor, que será mapeada para uma região ainda menor pela segunda camada, que será mapeada para uma região ainda menor pela terceira camada e assim por diante. Como resultado, mesmo uma grande alteração nos parâmetros da primeira camada não altera muito a saída.

Podemos evitar esse problema usando funções de ativação que não possuem essa propriedade de "esmagar" o espaço de entrada em uma região pequena. Uma escolha popular é a Unidade Linear Retificada (ReLu).