Influência da ordem das palavras e Categorização Multi-Label de Texto com VG-RAM

Relatório T1 - MAB786 Redes Neurais Sem Peso

Rodrigo Azevedo

Maio 2018

1 Introdução

Este relatório consiste em apresentar dois artigos que tratam do problema de categorização de texto utilizandos distintas abordagens e algoritmos de Redes Neurais Sem Peso. O primeiro, de título *The Influence of Order on a Large Bag of Words* introduz três técnicas de classificação propostas pelos autores, em que uma leva em consideração a ordem das palavras no conteúdo. O segundo, de título *Automated Multi-label Text Categorization with VG-RAM Weightless Neural Networks*, utiliza o VG-RAM para atribuir labels a textos que podem receber uma ou mais categorias.

2 Influência da ordem das palavras

As abordagens mais comuns têm como alvo problemas de classificação de texto em que os dados e o problema apresentam poucas classes e o conteúdo do texto apresentam muitas informações, tornando uma possível solução bem mais fácil de ser atingida. Muitas informações como palavras em poucas e distintas classes tornam o problema bem mais fácil de ser resolvido, seja distinguindo dados entre positivo ou negativo, ou textos sobre culinária, história e física, por exemplo. O problema torna-se muito maior quando os dados estão distribuídos em centenas ou milhares de classes, tendo uma dificuldade ainda maior ao possuir poucas palavras por exemplo. Este problema é refletido no CNAE (Classificação Nacional de Atividades Econômicas). O CNAE é uma forma de classificar negócios e empresas em diferentes categorias de atividades econômicas e no Brasil geralmente autoclassifica-se estas atividades. Essa autoclassificação pode não ser uma boa estratégia e classificação manual das atividades pode demandar mão-de-obra especializada e custos para o processo. A necessidade de projetar um mecanismo para classificar as atividades econômicas de forma automática surge com o grande aumento de dados e escassez de tempo.
A CNAE é organizado de forma hierárquica com 5 camadas, estas são: Seção, Divisão, Grupo, Classe e Subclasse. Como exemplo, a Subclasse 0500-3/02

Beneficiamento de Carvão Mineral, pertence a Seção B Indústrias Extrativas, Divisão 05 Extração de Carvão Mineral, Grupo 050 Extração de Carvão Mineral e Classe 0500-3 Extração de Carvão Mineral. As informações para cada negócio possuem poucas palavras, portanto o algoritmo deve ser capaz de classificar corretamente no enorme número de categorias (1301 subclasses) atraveś de poucas palavras.
O foco do estudo foi o uso de classificadores baseados na WiSARD, um modelo de rede neural sem peso, e o ganho obtido adicionando informações sintáticas além da semântica, neste caso adicionando a ordem das palavras como informação.

2.1 WiSARD Perceptron

O classificador WiSARD é uma rede neural sem peso em que os neurônios se baseiam em RAMs (Random Access Memory) e são capazes de armazenar 2^n bits, os neurônios então recebem e produzem valores binários no treinamento e predição. De forma mais específica, cada neurônio com suas posições iniciadas com 0 armazena o valor 1 na posição de endereço da RAM para cada valor recebido do input, esses neurônios são mapeados aleatoriamente com as posições do input da retina. Neurônios baseados em RAM são extremamente rápidos nas fases de treinamento e predição. A organização dos neurônios é feita através de discriminadores que são responsáveis por aprender ou reconhecer inputs.

2.2 Classificação das CNAEs

Os experimentos foram conduzidos com um dataset da CNAE contendo 3264 descrições de empresas e 764 registros do nível Subclasse. Para o treinamento da WiSARD foram utilizados os registros enquanto o teste foi realizado com as descrições. Utilizando técnicas de pré-processamento como stemming e remoção de stop-words, 1001 termos foram extraídos dos registros. A ocorrência dos termos foi considerado no preenchimento dos valores, recebendo 1 caso a palavra ocorra no texto e 0 caso contrário.

2.3 Single WiSARD (SiW)

Cada vetor de 1001 elementos foi representado por uma imagem de 32x32 com as últimas 23 posições preenchidas por 0. O algoritmo foi treinado com 764 registros de subclasses e testado com 3264 descrições de empresas. Cada discriminador tem como output o número de palavras reconhecidas menos o número de palavras ausentes. O resultado é representado pelo maior output de um discriminador. O algoritmo foi capaz de classificar corretamente 61% dos exemplos.

2.4 Hits-only WiSARD (HoW)

Neste caso o número de palavras ausentes não é subtraído do output, resultando apenas no número de palavras reconhecidas. Essa modificação trouxe um ganho de 3% no resultado, totalizando um percentual de 64% de classificações corretas.

2.5 Ordered WiSARD (OrW)

Neste caso, a ordem das palavras foi levada em consideração, adicionando ao classificador uma informação sintática. Portanto para cada palavra apresentada ao HoW no processo de predição o discriminador vencedor só foi escolhido antes de iniciar a divergência do número de discriminadores vencedores. Foi obtido uma melhora de 7% na classificação com esta abordagem.

3 Categorização Multi-Label de Texto com VG-RAM

A categorização de texto é um problema computacional bastante presente nos setores empresariais e acadêmicos. Seu uso para análise de sentimento, categorização de conteúdos, detecção de spam, entre outros, são problemas presentes no mundo real. A maioria dos trabalhos têm tratado do problema em sua versão single-label, em que apenas um único rótulo ou categoria é associado ao texto. Geralmente muito dos problemas são inclusive de classificação binário, em que uma classe é positiva ou negativa. Um problema de classificação Multi-label já permite a associação de um ou mais rótulos a um texto dentre inúmeras possibilidades de rótulos. Num ponto de vista geral, um problema multi-label pode ser representado por n problemas single-label independentes. Isso acontece apenas se for considerado que a associação de um determinado rótulo é independente da associação de um outro rótulo. É proposto então o uso do VG-RAM WNN (Virtual Generalizing Random Access Memory Weightless Neural Networks) para resolução do problema de classificação multi-label.

3.1 Categorização Multi-label

No problema de categorização multi-label um ou mais categorias podem ser associadas a um documento ou texto, por exemplo para um determinado texto de notícia jornalística sobre alimentos transgênicos podemos atribuir categorias como ciência, saúde e agro ou para uma determinada sinopse de filme podemos atribuir mais de uma categoria como terror e suspense.
Seja $\mathcal{D}$ o domínio de documentos, $\mathcal{C} = \{c_1, c_2, ..., c_{|\mathcal{C}|}\}$ um conjunto de possíveis categorias e $\Omega = \{d_1, ..., d_{|\Omega|}\}$ o corpus de documentos já categorizados (exemplos) em subconjuntos de $\mathcal{C}$, o problema multi-label busca uma função $f : \mathcal{D} \times \mathcal{C} \rightarrow \mathbb{R}$ que retorne um valor real para cada par de documento-categoria (d_j, c_i) que represente uma probabilidade ou alguma métrica para definir que o documento d_j deva ser classificado como c_i. Os valores da função podem ser ordenada e transformada numa função *rank*.
Para validação em um cenário multi-label, 4 diferentes métricas foram utilizadas: *hamming loss*, *one-error*, *coverage*, *average precision*.

3.1.1 Hamming loss ($hloss_j$)

O Hamming loss avalia exatamente o conjunto de categorias atribuído aos documentos de exemplo. É calculado para um determinado documento d_j quantas classificações incorretas foram feitas, ou seja, uma categoria que não pertence ao documento foi prevista pelo algoritmo ou uma categoria que pertence ao documento não foi prevista.

$$hloss_j = \frac{1}{|\mathcal{C}|}|P_j \Delta C_j|$$

onde $|\mathcal{C}|$ é o número de possíveis categorias, P_j é conjunto de categorias prevista pelo algoritmo, C_j conjunto de categorias verdadeiras do exemplo de teste e Δ é a diferença simétrica entre os conjuntos.

$$hloss_j = \frac{1}{|\mathcal{C}|}|P_j \Delta C_j|$$

O melhor desempenho ocorre quando $hloss = \frac{1}{p}\sum_{j=1}^{p} hloss_j = 0$.

3.1.2 One-error (one-error$_j$)

Verifica se categoria no topo do ranking está presente no conjunto de categorias atribuídas ao exemplo de teste.

$$\text{one-error}_j = \begin{cases} 0 & \text{se } [argmax_{c \in \mathcal{C}} f(d_j, c)] \in C_j \\ 1 & \text{caso contrário.} \end{cases}$$

O melhor desempenho ocorre quando one-error $= \frac{1}{p}\sum_{j=1}^{p} \text{one-error}_j = 0$.

3.1.3 Coverage (coverage$_j$)

Mede o quão longe precisa descer na lista rankeada de categorias atribuídas até cobrir todas as categorias verdadeiras do documento.

$$coverage_j = max_{c \in \mathcal{C}_|} r(d_j, c) - 1$$

O melhor desempenho ocorre quando coverage $= \frac{1}{p}\sum_{j=1}^{p} \text{coverage}_j = \frac{1}{p}\sum_{j=1}^{p}(|C_j| - 1)$.

3.1.4 Average Precision (avgprec$_j$)

Calcula a precisão média do rank cortando a lista após cada categoria $c_i \in C_j$.

$$avgprec_j = \frac{1}{|C_j|}\sum_{k=1}^{|C_j|} precision_j(R_{jk})$$

Onde R_{jk} é o conjunto de categorias rankeadas que iniciam no topo até uma posição k de uma categoria $c_i \in C_j$ para o documento de teste d_j, e $precision_j(R_{jk})$ é o número de categorias corretas em R_{jk} dividido por $|R_{jk}|$.

O melhor desempenho ocorre quando avgprec $= \frac{1}{p}\sum_{j=1}^{p}$ avgprec$_j = 1$.

3.2 VG-RAM WNN

Redes neurais baseados em RAM não armazenam conhecimento através de pesos em conexões dos neurônios mas sim em Random Access Memories (RAM) dentro dos neurônios. O treinamento pode ser feito com um único envio de input ao armazenar o output esperado com o endereço associado ao input. A saída de cada neurônio da VG-RAM é obtida através do par na lookup table mais próxima do input apresentado, a função utilizada para medir essa distância é a distância de hamming. Se houver mais de um output com a mesma distância mínima, um dos pares é escolhido aleatoriamente.

lookup table	X_1	X_2	X_3	Y
entrada 1	1	1	0	categoria 1
entrada 2	0	0	1	categoria 2
entrada 3	0	1	0	categoria 3
...				...
input	1	0	1	categoria 2

Table 1: Lookup table de um neurônio VG-RAM

Para categorizar documentos usando VG-RAM foi representado cada documento de texto como um vetor em que cada elemento possui um peso associado ao termo do documento, foi usado no caso o número de ocorrências do termo. As sinapses são conectadas aleatoriamente as entradas do sistema, que são constituidas pelos elementos do vetor correspondente ao documento. Cada sinapse x_i forma uma célula minchinton com a próxima sinapse x_{i+1} (a última $x_{|N|}$ forma uma célula com a primeira x_1). O tipo de célula minchinton usada retorna 1 caso a o valor associado ao input seja maior que o valor associado na próxima sinapse, retorna 0 caso contrário. A saída do neurônio é armazendo no lookup table com suas respectivas categorias associadas. Durante a fase de teste, para cada documento é contado o número de outpus que apontam para cada categoria. A saída final é calculada dividindo a contagem de cada categoria pelo número total de neurônios, esse valor varia entre 0 e 1 e representa a percentagem de neurônios que apontaram pra categoria específica. Um threshold pode ser usado para definir o conjunto de categorias associados ao exemplo de teste.

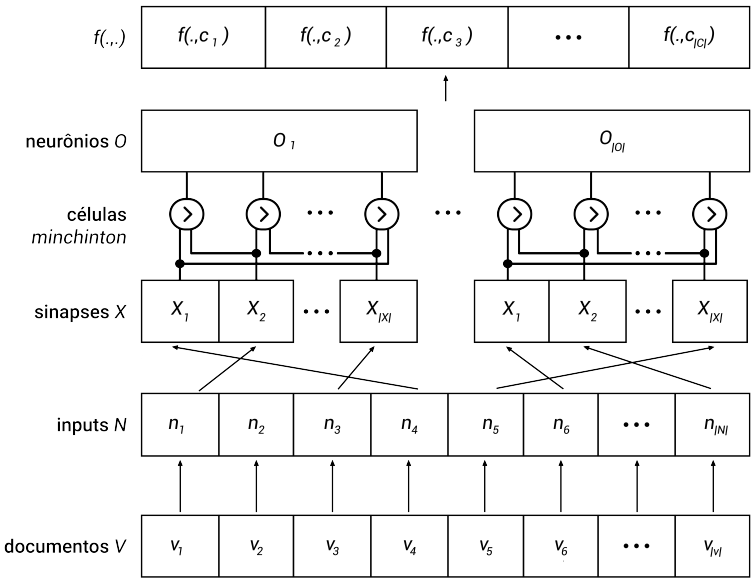

Figure 1: Arquitetura do VG-RAM WNN.

References

Charles B. Prado; Felipe M. G. França; Ramon Diacovo; Priscila M. V. Lima. The Influence of Order on a Large Bag of Words. *Intelligent Systems Design and Applications*, 2008.

Alberto F. De Souza, Felipe Pedroni, Elias Oliveira, Patrick M. Ciarelli, Wallace Favoreto Henrique, Lucas Veronese, and Claudine Badue. Automated multi-label text categorization with VG-RAM weightless neural networks. *Neurocomputing*, 2009