

Modelos neuronais para armazenamento e recuperação de informação II

J. Sacramento

INESC-ID Lisboa,
Instituto Superior Técnico,
Campus Taguspark, Porto Salvo, Portugal

Recuperação de Informação, DEAEIC, 2011

Outline

Implementação associativa de índices invertidos

Resultados gerais

Discussão

Outline

Implementação associativa de índices invertidos

Resultados gerais

Discussão

Em análise

V. J. Hodge and J. Austin. An evaluation of standard retrieval algorithms and a weightless neural approach.
(IJCNN 2000)

V. J. Hodge and J. Austin. An evaluation of standard retrieval algorithms and a binary neural approach.
(Neural Networks 2001)

Tarefa a resolver

Tarefa de indexação de documentos típica de RI

- ▶ É nos dada a memorizar uma colecção de documentos da forma $\{\mathbf{w}^1, \dots, \mathbf{w}^\mu, \dots, \mathbf{w}^M\}$
 - ▶ Índice μ identifica univocamente um determinado documento
 - ▶ $\mathbf{w}^\mu \in \{0, 1\}^m$ codifica a presença de termos no documento μ (representação booleana)

Procura no sentido “inverso”

- ▶ Dado um conjunto de termos $\tilde{\mathbf{w}} \in \{0, 1\}^m$, determinar a lista de documentos $\mathcal{D} = \{\mu : \forall \tilde{w}_i=1, w_i^\mu = 1\}$

Proposta “neuronal” apresentada

Representação explícita de cada documento por um vector $\mathbf{d} \in \{0, 1\}^M$ com exactamente 1 bit activo

$$d_j = \begin{cases} 1, & j = \mu \\ 0, & \text{caso contrário} \end{cases}$$

Guardar numa matriz \mathbf{A} usando a regra de Willshaw associações da forma $\mathbf{w}^\mu \mapsto \mathbf{d}^\mu$

- ▶ Após M associações, $a_{ij} = \min \left(1, \sum_{\mu=1}^M w_i^\mu d_j^\mu \right)$

Determinação do conjunto \mathcal{D}

Para uma dada consulta $\tilde{\mathbf{w}}$, um vector binário de resposta \mathbf{d} é determinado

- ▶ Cada componente d_μ indica a presença do documento μ no conjunto \mathcal{D} ,

$$d_\mu = \begin{cases} 1, & \sum_{i=1}^m a_{i\mu} \tilde{w}_i \geq \Theta \equiv \sum_{i=1}^m \tilde{w}_i \\ 0, & \text{caso contrário} \end{cases}$$

- ▶ Vector de resposta \mathbf{d} tem $|\mathcal{D}|$ bits activos (codificação posicional)

Codificação da matriz

Os autores verificam que ao utilizarem um corpo de texto natural (*Reuters 21578 Newswire*), a matriz resultante \mathbf{A} é maioritariamente esparsa

- ▶ Linhas “suficientemente” esparsas (nível de corte empírico) são codificadas como listas de índices inteiros

Reflecção

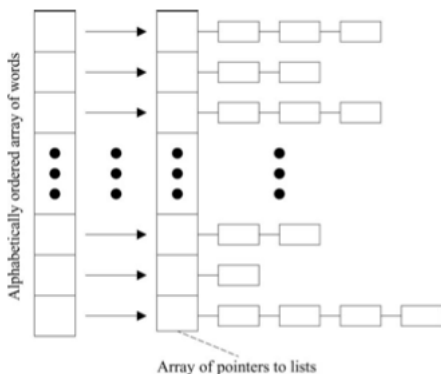
Algumas observações em ordem

- ▶ Repare-se que a codificação posicional onde cada \mathbf{d}^μ tem apenas um bit activo não induz armazenamento distribuído que caracteriza a memória neuronal de Willshaw
 - ▶ O valor de cada entrada a_{ij} depende apenas de um par, pois $j = \mu$
- ▶ Matriz \mathbf{A} resultante é na verdade equivalente à estrutura clássica de RI, já apresentada na disciplina
 - ▶ Mais próxima ainda pois os autores consideram o armazenamento eficiente das linhas, e discutem até uma forma (ingénua) de representação esparsa da matriz

Comparação com IFL

Autores comparam a performance da estrutura obtida com um tradicional índice invertido (designado no artigo por IFL)

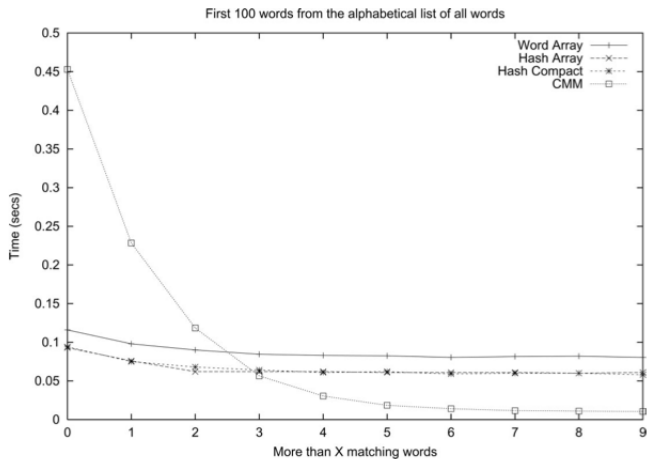
- ▶ Array ordenado de termos (representação explícita), com ligações a listas de IDs de documentos



Procura IFL vs. modelo “neuronal”

- ▶ IFL utiliza procura binária sob colecção ordenada de termos
- ▶ IFL é mais rápido para as procuras sequenciais, com \mathcal{D} da forma descrita até agora
- ▶ Autores consideram também procuras parciais da forma “pelo menos X termos presentes dos Y primeiros termos”
 - ▶ Caso neuronal: basta definir $\Theta = X$, activar Y primeiros bits do vector
 - ▶ Caso tradicional: contador por documento, Y procuras binárias para localizar os termos

Comparação tempos de procura



Adaptado de Hodge and Austin (2001)

Outline

Implementação associativa de índices invertidos

Resultados gerais

Discussão

Implementações em computadores sequenciais

Analisemos agora um caso geral. Seguimos o tratamento de Knoblauch et al. (2010)

- ▶ Associação de M pares arbitrários de padrões $\mathbf{u}^\mu \mapsto \mathbf{v}^\mu$,
 $\mathbf{u}^\mu \in \{0, 1\}^m$, $\mathbf{v}^\mu \in \{0, 1\}^n$
- ▶ Pergunta chave: existe alguma vantagem em utilizar um esquema de armazenamento distribuído do tipo Willshaw numa implementação em computador sequencial?

Condições favoráveis ao modelo de Willshaw

Tomamos como verdade uma série de pressupostos que optimizam o modelo de Willshaw

- ▶ Cada \mathbf{u}^μ tem exactamente k bits activos, $k \ll m$
- ▶ Cada \mathbf{v}^μ tem exactamente l bits activos, $l \ll n$
- ▶ Em ambos os casos os bits são uniformemente distribuídos pelas posições possíveis

Consideramos a implementação sequencial mais eficiente do modelo de Willshaw

- ▶ Cada linha de \mathbf{A} está comprimida (ponteiros, códigos de Huffman ou Golomb)

Solução tradicional

Definimos agora uma solução convencional para a tarefa

- ▶ Matriz $\mathbf{U} \in \{0, 1\}^{M \times m}$ guarda os padrões de endereçamento
- ▶ Matriz $\mathbf{V} \in \{0, 1\}^{M \times n}$ guarda os padrões de conteúdo
- ▶ Comprime-se matriz de procura \mathbf{U} por coluna

Tempo de procura (em média, assintoticamente)

$$t^C \sim \# \text{ bits na entrada} \times \# \text{ médio de bits por coluna} = k^2 M / m$$

Comparando tempos de procura

Tempo de procura de Willshaw (em média, assintoticamente)

$$t^W \sim \# \text{ bits na entrada} \times \# \text{ médio de bits por linha} = knp$$

- ▶ p é a fracção de bits activos na matriz \mathbf{A}
- ▶ Expressão não depende directamente do número de padrões M , mas este influencia p

Knoblauch et al. (2010) optimiza k e M para o modelo de Willshaw (obtendo um certo p óptimo) e inspecciona o rácio t^C/t^W

Resultados

Em resumo, e omitindo os cálculos

- ▶ Descobre-se que para a maioria dos parâmetros que optimizam o modelo de Willshaw, este fica sempre aquém da solução convencional

Em ligação com o artigo abordado anteriormente

- ▶ A codificação posicional unária evita os problemas referidos, pois o modelo degenera numa solução convencional (i.e., cada coluna, cada documento)

Outline

Implementação associativa de índices invertidos

Resultados gerais

Discussão

Discussão

Aliviando a carga negativa dos resultados anteriores

- ▶ Modelos neuronais de memória exigem em geral hardware específico, paralelo, caso queiram apelar pelo lado técnico
- ▶ Fica por investigar se a tolerância a ruído é suficiente a uma aplicação técnica do tipo “vizinho mais próximo”
- ▶ Possível problema com correlações nas entradas, como discutido na apresentação anterior

References I

- V. J. Hodge and J. Austin. An evaluation of standard retrieval algorithms and a binary neural approach. *Neural Networks*, 14(3):287–303, 2001.
- A. Knoblauch, G. Palm, and F. T. Sommer. Memory capacities for synaptic and structural plasticity. *Neural Computation*, 22(2):289–341, 2010.