Proceedings of the IV Brazilian Conference on Neural Networks - IV Congresso Brasileiro de Redes Neurais
pp. 192-197, July 20-22, 1999 - ITA, São José dos Campos - SP - Brazil

Utilização de Redes Neurais Artificiais na Recuperação de Informação em Bases de Dados Textuais

Ramon da Cunha Lopes[1]
Eduardo Fernandes Barbosa[2]
Antônio de Pádua Braga[3]
[1]Departamento de Engenharia Elétrica - ICMG
[2,3] Departamento de Engenharia Eletrônica - UFMG
E-mails: ramonc@gold.horizontes.com.br, ebarbosa@dcc.ufmg.br, apbraga@cisne.cpdee.ufmg.br

Abstract

The Information Retrieval in the scope of this work, was considered as the retrieval of a document from a set of documents gathered at random and organized by key-words and by the context of each document defined by a librarian. The documents used as base of information were initially classified by placing each key found in alphabetic order, in an inverted list. The keys were counted and gathered according to the number of times that they occurred in each document. Starting with a search algorithm using an Artificial Neural Network, a result of this search of key-words was obtained, associated with a specific context in textual database, with a smaller number of access in relation to the traditional search, using the binary tree algorithm. The algorithm was tested with a set of documents obtained in the Internet, in the areas of Electronics, Artificial Intelligence and Artificial Neural Networks. The weights and biases were stored in a hard disk for future simulations and queries. A study is made to characterize the behaviour at random of the distribution of the keys along the text. From this point on, a probabilistic approach is developed to measure the context of the key-words in the set of the documents.

1. Introdução

A quantidade de informação disponível em forma digital vem aumentando a uma taxa que dobra a cada vinte meses [1], levando os especialistas a se dedicarem no sentido da obtenção de resultados significativos na recuperação de informação a custos exequíveis.

Uma quantidade muito grande de informação se encontra em documentos textuais, como transações financeiras, procedimentos legais e atividades governamentais [1], o que torna a Recuperação de Dados (RD) através de bancos de dados uma tarefa complexa principalmente devido aos vários formatos em que os documentos e as informações estão armazenados.

A Recuperação de Informação (RI) se difere da RD devido a algumas propriedades [2] que a tornam mais próxima da forma como os dados estão disponibilizados nas diversas fontes e dão um significado relevante para o contexto da busca empreendida por um usuário não especialista que espera respostas exatas. A RD é mais precisa e mais rápida quanto à localização da posição de uma palavra chave mas nem sempre permite fornecer como resposta a uma consulta uma informação relevante ao contexto requerido.

Um mecanismo de RI possui aspectos dinâmicos, abertos e negociáveis em relação à consulta do usuário [3]. Esta negociação se refere a questões mais complexas da comunicação humana e possui um compromisso com a fidelidade da informação. A consulta feita por um usuário não especialista geralmente é imprecisa, logo exige do sistema de RI a incorporação do conhecimento do bibliotecário ou do especialista [3].

Alguns problemas vêm sido levantados sobre a Recuperação de Informação em muitos esforços de pesquisa para resolver o problema da evolução de Sistemas de Recuperação de Informação. Uma lista contendo as seis principais medidas a serem obtidas são indicadas por Cleverdon [4]:
- A cobertura da coleção, ou seja, a extensão para a qual o sistema inclui material relevante;
- O tempo de atraso, ou intervalo médio entre a requisição da busca e a sua resposta requerida;
- A forma de apresentação da saída;
- O esforço envolvido no uso para a obtenção das respostas para as consultas requeridas;
- O retorno do sistema, isto é, a proporção relevante de material atualmente recuperado na resposta para a consulta requerida;
- A precisão do sistema, isto é, a proporção de material recuperado que é atualmente relevante.

Com o advento de ferramentas modernas de Inteligência Computacional observa-se uma melhoria constante no desempenho de ferramentas para manipulação de documentos e obtenção de informação objetiva a partir de uma massa de dados. A proposta de modelamento de documentos utilizando uma RNA [5], para o escopo deste trabalho, sugere que cada conexão seja ajustada como uma representação local [6] onde cada nó da rede seja usado para representar um documento ou uma palavra-chave [7].

Uma representação através de um modelo conexionista pode representar um sistema de RI a partir de uma camada de consulta, uma camada de palavras-chave e uma camada de documentos [8].

Neste trabalho são propostas duas abordagens para a busca de informação armazenada em bases de dados textuais. A primeira trata de desenvolver um modelo para representar a lista invertida que aponta para a base de dados textual, permitindo que o usuário entre com uma palavra-chave e encontre a posição dessa palavra-chave no arquivo de índices para posteriormente localizar o endereço no documento original.

A segunda abordagem avança em direção ao significado da consulta proposta pelo usuário, delimitando um contexto para a palavra-chave. Nesta implementação o usuário fornece como entradas para a Rede Neural Artificial (RNA) o contexto e a palavra-chave e são retornado um ou mais documentos relevantes para a consulta empreendida.

A relevância de palavras-chave em documentos tem sido definida como uma variável aleatória [9] . Para a segunda implementação proposta neste trabalho, surge a necessidade de considerar este aspecto para definir o tipo de RNA a ser utilizado para a obtenção de uma recuperação que atenda ao contexto especificado na consulta.

A RNA utilizando a arquitetura Perceptron multicamadas não ofereceu resultados satisfatórios durante o desenvolvimento deste trabalho quando implementou o conhecimento a partir de uma abordagem determinística utilizando por exemplo o algoritmo Backpropagation [10]. Ao mudar o enfoque, utilizando a RNA para classificar palavras-chave em contextos fornecidos pelo usuário durante o treinamento a Rede Neural de Base Radial (RNBR) apresenta um desempenho melhor em relação a uma rede MLP [11]. A justificativa para essa diferença de comportamento é devida à forma como os dados estão distribuídos .

2. Ordenação de Índices em Bases de Dados Textuais Estáticas

Um banco de dados constituído de textos geralmente recebe adições periódicas, mas nenhuma atualização é realizada nas informações já existentes [12]. A informação armazenada em uma base de dados textual depende de um mecanismo para armazená-la e recuperá-la de maneira eficaz. Um método usado para esta recuperação é a criação de um arquivo de índices montado a partir de uma lista invertida das palavras-chave [13]. O objetivo principal da ordenação é facilitar a recuperação posterior de ítens do conjunto ordenado [14]. A classificação em grandes bases de dados textuais se divide em uma ordenação de blocos contendo parte do arquivo texto na memória principal numa primeira fase, e numa segunda fase ordena-se a partir da memória externa o conteúdo dos blocos ordenados, visto não ser possível executar toda a intercalação na memória principal devido ao tamanho do arquivo, conforme sugere o diagrama da Figura 1, onde o algoritmo indexador I classifica as palavras-chave retiradas dos documentos$Doc1$, $Doc2$ e $Doc3$, gerando inicialmente os arquivos de índice $IT1$, $IT2$ e $IT3$ que serão classificados e codificados numericamente pela rotina de intercalação C que fornecerá na saída um índice contendo todas as palavras-chave encontradas nos documentos de entrada.

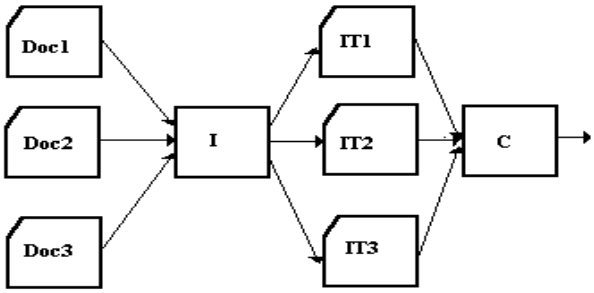

Figura 1: Indexação e Rotina de Intercalação

O sistema de RI deve estruturar os novos documentos em um índice contendo as chaves que vão sendo inseridas após cada atualização. Uma forma de organizar um documento utilizando suas palavras-chave é através da montagem de um PAT array. O PAT array [15] é uma estrutura de dados construída a partir de uma sequência de chaves semi-infinitas, ou seja, chaves que se iniciam na posição atual do texto e se estendem à direita tanto quanto for necessário ou até o fim do documento. O PAT array é uma representação compacta da árvore PATRICIA (*Pratical Algorithm To Retrieve Information Coded In Alphanumeric*) por armazenar apenas os nodos externos da árvore [16]. Criado originalmente por Morrisson [15] para ser utilizado na recuperação de informação em arquivos de grande porte o algoritmo sofreu modificações que permitiram uma otimização em relação às árvores binárias trie [14],[15] e [16].

Existem vários algoritmos de ordenação na memória interna, sendo que o mais utilizado devido ao seu bom desempenho é o Quicksort, desenvolvido e publicado por Hoare [17].

O algoritmo Quicksort funciona a partir da escolha de um ítem x chamado pivô de modo a particionar o vetor A que conterá ao final chaves menores ou iguais a x na parte esquerda e chaves maiores ou iguais a x na parte direita. O custo do algoritmo, no melhor caso [12] ocorre quando o arquivo é dividido em duas partes iguais conforme a Equação (1).

$$C(n) = 2C(n/2) + n \tag{1}$$

Onde n é o número total de chaves.

Caso o arquivo a ser ordenado tenha um tamanho maior que a memória principal, é necessário particioná-lo e classificar cada partição internamente executando posteriormente uma intercalação entre os

vários módulos. A ordenação em memória secundária se dá de forma diferente em relação à memória principal. O processo inicia-se com uma fase onde o documento é dividido em vários conjuntos de chaves menores que podem ser classificadas na memória principal e armazenadas novamente em disco. Uma segunda fase de intercalação é executada então, no sentido de obter todas as chaves classificadas em um único arquivo em disco. O desempenho do algoritmo utilizado para a intercalação balanceada para vários caminhos é dado pela Equação (2).

$$P(n) = log_f \frac{n}{m} \quad (2)$$

Onde P é o número de passadas em cada conjunto de chaves, n é o número de registros, m é a memória disponível em palavras e f é o número de caminhos. A codificação das palavras-chave é feita substituindo os caracteres alfa-numéricos por números decimais de 0 a 36 e dividindo por 100 a cada novo caracter. A palavra "$monitor$", por exemplo, resulta 0.23252419302528.

3. Redes Neurais Artificiais

A Inteligência Computacional tem fornecido ferramentas que exploram o uso massivo do paralelismo para a solução de problemas que tinham, no mínimo, o desempenho comprometido quando eram abordados com a utilização de modelos desenvolvidos segundo a computação sequencial de von Neumann, conforme citado em Lippmann [18].

Alguns problemas práticos não podem ser modelados através de funções contínuas ou discretas lineares ou não lineares, principalmente se a componente aleatória do sinal da entrada considerado é muito alta. A quantidade de ocorrências das chaves de um documento em uma lista invertida se comporta como uma variável aleatória embora possua uma distribuição monotônica ao longo do texto. Algumas palavras-chave terão a probabilidade de ocorrer um maior número de vezes em um documento de um determinado contexto A em relação a um contexto B. A RNBR torna-se mais adequada para este trabalho porque permite a representação das ocorrências das palavras-chave através das gaussianas descritas por este tipo de RNA. Uma função de base radial típica, centrada em c_j e com raio em r_j é dada pela gaussiana da Equação (3).

$$h_j(x) = exp(-\frac{(x - c_j)^2}{r_j^2}) \quad (3)$$

O mecanismo de busca proposto para obtenção de informação através da RNBR recebe palavras-chave fornecidas por um conjunto de documentos. Nos experimentos realizados neste trabalho foram utilizados 9 documentos para treinar a RNA. A RNBR possui nove saídas, cada uma apontando para um dos documentos utilizados no treinamento.

Para a obtenção da classificação dos documentos, ordenando-os do mais significativo para o texto de menor relevância, foi inserida mais uma camada de neurônios implementados de modo a competirem entre si para indicar o neurônio vencedor, selecionando-se assim o documento de maior significado para a consulta realizada pelo usuário, como será detalhado na seção 5 e pode ser visto na Figura 8.

4. Modelagem de Documentos

Um conjunto de nove documentos obtidos a partir de resumos de artigos técnicos da IEEE e da ACM são utilizados para os testes apresentados neste capítulo. Estes documentos foram submetidos a uma indexação das chaves e posterior codificação numérica para o treinamento da RNBR.

A Rede Neural MLP (Multi-Layer Perceptron) possui entre suas características positivas, a facilidade em discriminar funções a partir de um conjunto de padrões de treinamento, obtendo uma função que se aproxima dos padrões com um erro mínimo. A partir do índice obtido, são retiradas algumas chaves e suas respectivas posições no disco para servirem de padrão de treinamento para a RNA. Após o treinamento, a RNA é simulada a partir de um conjunto de chaves que permite avaliar o erro em torno dos valores esperados. Uma função descrevendo a distribuição das palavras-chave pode ser obtida como resposta da RNA após o treinamento como sugere a Figura 2. Como o erro apresentado pela RNA depende muito da presença de palavras-chave que estão aleatóriamente distribuídas, faz-se necessário um estudo sobre a característica do sinal, ou seja, se pode ou não ser representado adequadamente por uma função contínua.

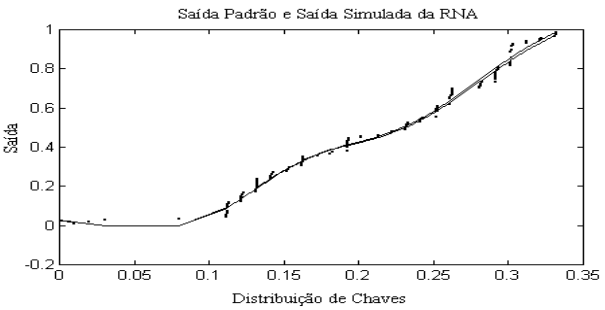

Figura 2: Treinamento utilizando MLP

Para avaliar apropriadamente como se comporta a posição das chaves em função do valor numérico das mesmas em ordem crescente foi utilizado o conceito de auto-correlação [19] que permite quantizar a presença de componentes aleatórias em uma determinada massa de dados.

O Teorema da correlação [20] fornece um mecanismo que pode ser aplicado a um sinal para verificar se o mesmo apresenta um alto grau de componentes aleatórias, e é calculado a partir da Equação (4).

$$z(t) = \int_{-\infty}^{\infty} x(\tau)h(t + \tau)d\tau \quad (4)$$

Aplicando-se o processo de auto-correlação sobre os pontos utilizados para o treinamento da RNA da seção anterior, conforme pode ser visto na Figura 2, observa-se que há uma alta correlação entre chaves diferentes como é apresentado na Figura 3.

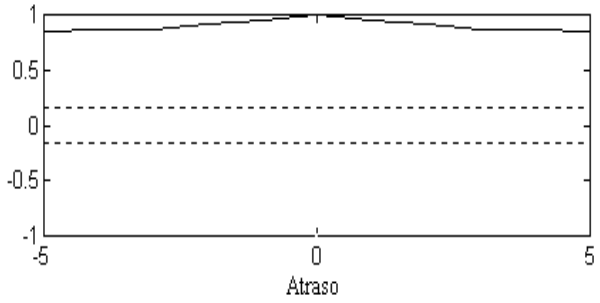

Figura 3: Auto-correlação da Função de Distribuição

Aplicando-se o cálculo da auto-correlação sobre a frequência de ocorrência de cada chave, obtém-se uma resposta que possui uma presença maior de sinais aleatórios em relação à representação anterior, como mostrado na Figura 4.

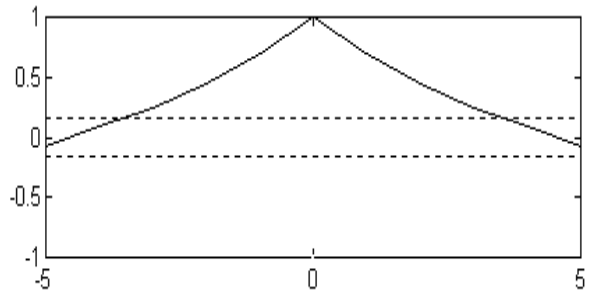

Figura 4: Auto-correlação da Distribuição de chaves

Este estudo conduz a um modelamento utilizando a RNBR que representa melhor o conjunto de padrões apresentados para o treinamento. As entradas da RNBR utilizadas para treinamento são as chaves amostradas do conjunto de documentos, o contexto de cada documento e o nível de cada documento. Este nível permite diferenciar dois documentos que possuam as mesmas chaves de entrada com o mesmo contexto. As saídas da RNBR apontam cada uma para um documento e fornecem a quantidade de ocorrência de uma determinada chave de entrada neste documento, como apresentado na Figura 5.

A lei de Zipf [21] afirma que a frequência de um termo em um documento é inversamente proporcional ao peso dado a este termo no documento. Com isto é possível nivelar palavras comuns que ocorrem várias vezes a palavras que ocorrem um menor número de vezes, atribuindo-lhes pesos apropriados de maneira que a simples repetição de um termo não o elege a uma chave de alta relevância no documento. A distribuição das palavras-chave em ordem decrescente de ocorrência nos documentos é apresentada na Figura 6.

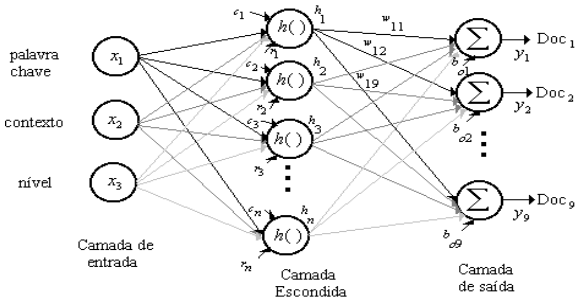

Figura 5: Estrutura da RNBR

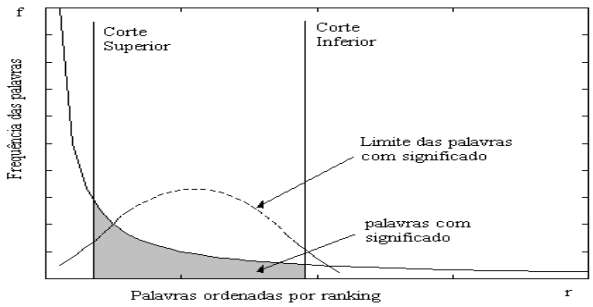

Figura 6: Curva hiperbólica da lei de Zipf

O algoritmo desenvolvido para o escopo deste trabalho não processa a análise automática de documentos. Embora a definição do contexto não seja feita automaticamente, o especialista pode ajustar o raio da curva de Gauss, delimitando até qual caracter um conjunto de palavras possui o mesmo significado. Outra intervenção manual feita pelo bibliotecário é a definição dos cortes inferior e superior onde o especialista passa a considerar um filtro sobre os dados definindo qual o número mínimo e máximo de ocorrências definirão as chaves que estão contidas no assunto em questão, conforme mostrado na Figura 7.

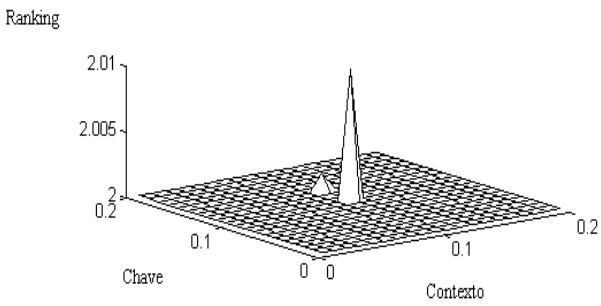

Figura 7: Resposta da RNBR

5. Simulação e Análise de Desempenho

A simulação foi obtida a partir da leitura dos pesos e termos de polarização que se encontravam em disco, salvos na fase de treinamento e lidos por uma rotina escrita em C que responde com precisão a partir das chaves de entrada e do contexto desejado.

Para se obter a classificação, foi feito um arranjo baseado no aprendizado por competição utilizado nos sistemas auto-organizativos.

O neurônio Instar proposto por Grossberg [22] considera a interferência da inibição dos neurônios laterais e leva em conta também o valor anterior de saída do mesmo neurônio como ativação. Esta característica conduz a um sistema de primeira ordem dado pela Equação (6).

$$\dot{x_i} = -Ax_i + (B - x_i)(\sum_i (x_i w_i + b)) - x_i(\sum (x_k w_k + b)) \quad (5)$$

A partir da implementação de neurônios com inibição lateral foi observado que após a primeira iteração a única saída que permaneceu positiva era referente à maior entrada, desde que não houvessem entradas iguais. Esta observação permitiu utilizar uma função de ativação do tipo:

$$f(x) = \begin{cases} 0 & x \leq 0 \\ x & x > 0 \end{cases} \quad (6)$$

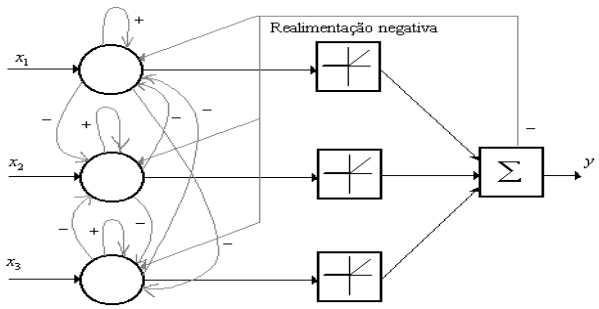

Figura 8: Recuperação de Informação utilizando uma Rede Competitiva

O sistema proposto pode ser avaliado em termos de desempenho a partir da carga dos pesos e termos de polarização com uma complexidade *O(n)*. Mesmo que haja um baixo desempenho devido ao processamento da RNA em memória principal, acessando um elevado número de pesos e termos de polarização, o número de operações só aumentará proporcionalmente ao aumento do número de palavras-chave, ou seja, cada palavra-chave acessada deverá ler todos os pesos de entrada e todos os pesos de saída da RNBR. Como o número de pesos aumenta com o número de palavras-chave o custo cresce proporcionalmente ao número total de palavras-chave, enquanto uma busca binária, por exemplo, possui uma complexidade da ordem de O(log(n)) para o caso médio [12], onde n é o número total de palavras-chave.

Nesta seção será comparada a efetividade da recuperação de documentos do sistema proposto usando a RNBR em relação à efetividade utilizando a regra do cosseno [23][24] e que busca a similaridade de uma consulta com um documento, dado pela Equação (7).

$$cos(Q, D_d) = \frac{1}{W_d W_q} \sum_{t \in Q} f_{d,t} (log \frac{N}{f_t})^2 \quad (7)$$

Onde W_d é o peso do documento, W_q é o peso da consulta, $f_{d,t}$ é a frequência de cada termo para cada documento, N é o número total de documentos e f_t é a frequência do termo nos documentos.

A curva apresentada na Figura 9 permite avaliar e comparar métodos de recuperação de informação, comparando o desempenho de um mecanismo em relação ao outro.

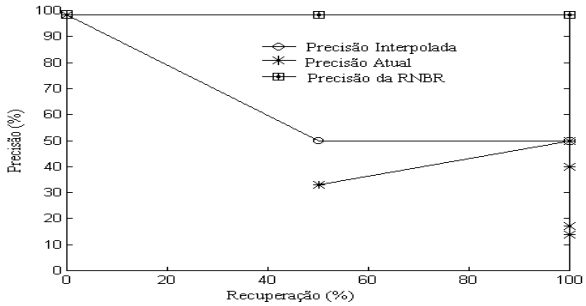

Figura 9: Curva de Precisão x Recuperação para avaliar o grau de similaridade no mecanismo de busca

6. Conclusão

Alguns aspectos negativos podem ser analisados a partir do trabalho aqui proposto. O primeiro ponto negativo é a exigência da presença de todos os especialistas relacionados aos contextos envolvidos nos documentos durante a fase de treinamento da RNBR. O método do cosseno classifica automaticamente o contexto através da lei de Zipf dando maior peso a palavras que concentram-se em alguns documentos e possuem pequena frequência em outros. Um outro ponto negativo é a impossibilidade de atualizar o mecanismo dinâmicamente à medida que são inseridos novos documentos devido à necessidade de treinamento da RNBR para atualização dos pesos e termos de polarização.

Existem, no entanto, características positivas que motivam futuras pesquisas nesta área, citadas a seguir. O tempo gasto no acesso e o consumo de memória principal são pontos positivos da RNBR em relação ao método do cosseno porque são carregados apenas os pesos e termos de polarização da RNA referentes ao contexto fornecido na consulta, reduzindo significativamente a quantidade de palavras-chave a serem manipuladas. Outro ponto que pode ser considerado positivo é a incorporação do conhecimento do especialista dentro do mecanismo de busca. No método do cosseno, a validação de um documento para um determinado contexto só é permitida após a

confirmação do especialista, definindo se um documento está ou não naquele contexto de consulta. A precisão da consulta torna-se exata com a RNBR porque a resposta a uma consulta representa um retorno exatamente do contexto inserido pelo especialista na fase de treinamento, enquanto no mecanismo do cosseno a avaliação do especialista só é feita depois da consulta, correndo-se o risco de retornar documentos fora do contexto.

A lista contendo as seis principais medidas a serem obtidas para um Sistema de Recuperação de Informação, indicadas por Cleverdon [4]: a cobertura da coleção, o tempo de atraso, a forma de apresentação da saída, o esforço envolvido no uso para a obtenção das respostas para as consultas requeridas, o retorno do sistema e a precisão do sistema são atendidas em parte pelo sistema proposto utilizando Redes Neurais Artificiais. Estes requisitos são em parte atendidos porque como todas as chaves fornecidas para treinamento estão representadas através de pesos e termos de polarização, a recuperação da informação é rápida e fornece a precisão especificada pelo profissional que participa na fase de treinamento, porque, a priori, é ele quem define a tolerância em torno de cada palavra-chave e cada contexto.

O trabalho proposto abre uma possibilidade de pesquisa futura na área de Recuperação de Informação utilizando Redes Neurais Artificiais associando o contexto do usuário com mecanismos automáticos no sentido de incorporar o conteúdo ao mecanismo de busca e generalizar o conhecimento do especialista para o reconhecimento de novos documentos.

7. Agradecimentos

Agradecemos a oportunidade viabilizada através de esforços do CNPQ e da Associação Brasileira de Redes Neurais para a realização do IV Congresso Brasileiro de Redes Neurais.

Referências

[1] I. H. Witten, A. Moffat, and T. C. Bell. *Managing gigabytes: compressing and indexing documents and images*. Van Nostrand Reinhold, New York, 1994.

[2] C. J. Rijisbergen. *Information Retrieval*. Whitefriars Press Ltd, London, 1979.

[3] R. S. Taylor. *Question Negotiation and Information Seeking in Libraries*. American Librarian Association, 1968.

[4] J. M. e. M. K. C. Cleverdon. *Factors Determining the Performance of Indexing Systems: ASLIB Cranfield Research Project. Volume 1: Design.* ASLIB Cranfield Research Project, Cranfield, 1966.

[5] J. L. M. e D. E. Rumelhart. *Parallel Distributed Processing, Psychological and Biological Models*. MIT Press, Cambridge, Mass, 1986.

[6] R. J. B. e D.L. McGuinness. *Knowledge representation, connectionism, and conceptual retrieval*. ACM SIGIR, 1988.

[7] P. Biron. *Connectionist Information Retrieval A Survey of Recent Work.* Computer Science, New York, 1990.

[8] K. L. Kwok. *A neural network for probabilist information retrieval*. Twelfth Annual International ACMSIGIR Conference, Cambridge, MA, 1989.

[9] F. Gebhardt. *A Simple Probabilist Model for the Relevance Assessment of Documents.* Information Processing and Management, USA, 1975.

[10] P. D. Wasserman. *Advanced Methods in Neural Computing*. Van Nstrand Reinhold, New York, 1993.

[11] F. C. e C. J. Van Rijisbergen. *Modelling Adaptive Information Retrieval*. Departament of Computing Science. University of Glasgow, Scotland - U.K., 1993.

[12] N. Ziviani. *Projeto de algoritmos: com implementações em Pascal e C*. Pioneira, São Paulo, 1996.

[13] G. H. G. e R. Baeza-Yates. *Handbook of Algorithms and Data Structures*. Addison-Wesley, Mass, 1991.

[14] E. F. Barbosa. *Efficient Text Searching Methods for Secondary Memory*. UFMG, MG, Brasil, 1995.

[15] D. R. Morrisson. *PATRICIA - Pratical Algorithm To Retrieve Information Coded In Alphanumeric*. Journal of the ACM, 1968.

[16] D. E. Knuth. *The Art of Computer Programming, v. 3: Sorting and Searching*. Addison-Wesley, Mass, 1973.

[17] C. A. R. Hoare. *Quicksort*. The Computer Journal, 1962.

[18] R. P. Lippmann. *An Introduction to Computing With Neural Nets*. IEEE ASSP Magazine, 1997.

[19] K. R. Godfrey. *Signal Processing for Control Lecture Notes*. Springer, Yerlag, 1986.

[20] E. O. Brigham. *The Fast Fourier Transform and its applications*. Prentice Hall, New Jersey, 1988.

[21] G. Zipf. *Human Behavior and the Principle of Least Effort*. 1949.

[22] S. Grossberg. *Studies of the Mind and Brain*. Reidel Press, Drodrecht, Holland, 1982.

[23] G. Salton. *Automatic Information Organization and Retrieval*. McGraw-Hill, New York, 1968.

[24] G. Salton. *The Smart Retrieval System - Experiments in Automatic Document Processing*. Prentice-Hall, Englewood Cliffs, New Jersey, 1947.