Análise de Abordagens para Recuperação de Informação em Tabelas na Web

Filipe Roberto Silva[1], **Ronaldo dos Santos Mello**[1]

[1]Departamento de Informática e Estatística– Universidade Federal de Santa Catarina
Caixa Postal 476 – 88.040-900 – Florianópolis – SC – Brasil

{filipesilva.sc,ronaldo}@inf.ufsc.br

Abstract. *The web has been turning into a rich data source. Recent researches try to create even more ways to use these data and/or to facilitate their access. There are a lot of tables in the Web that hold useful data for human consumption. They are characterized by the <table>tag. This paper details information retrieval approaches related to Web tables, presenting a brief description of them, as well as a comparison of their main features. Besides, some open issues in this research area are highlighted.*

Resumo. *A web tem se tornado uma rica fonte de dados. Pesquisas recentes tentam criar cada vez mais meios de utilizar e/ou facilitar o acesso a esses dados. Existem muitas tabelas na Web que possuem dados úteis para o consumo humano. Elas são caracterizadas pela tag <table>. Este trabalho detalha abordagens de recuperação de informação relacionadas a tabelas na web, apresentando uma breve descrição delas e mostrando um comparativo entre suas principais características. Além disso, são destacadas algumas questões em aberto nessa área de pesquisa.*

1. Introdução

Segundo Lai (2013) existe uma grande quantidade de tabelas na web e essas tabelas possuem dados que poderiam ser de grande proveito para a obtenção de informação útil para consumo humano. Essa informação poderia ser melhor aproveitada se processada por um computador e indexada, visto que este é um trabalho bastante árduo para ser feito de forma manual. Porém, essas fontes de dados não possuem um padrão bem definido. As tabelas na web são criadas para serem lidas por pessoas. Por isso podem estar em várias disposições diferentes e tratar sobre diversos assuntos. Enfim, pode ser complexo para um computador entendê-las e coletar seu conteúdo.

As tabelas na web, assim como as tabelas relacionais, basicamente são compostas por rótulos que caracterizam os atributos e valores que caracterizam as tuplas. Porém a forma como essas estruturas aparecem nas tabelas pode variar bastante. Por isso, muitas abordagens tratam de identificar essas estruturas nas tabelas. Outras abordagens observadas tratam de recuperar informação semântica presente nas tabelas. Isso poderia ser útil para uma indexação mais significativa.

Este artigo tem como objetivo descrever e comparar abordagens de recuperação de informação em tabelas na web, mostrando suas características, pontos fortes e fracos e por fim, apresentando sugestões de tópicos a serem abordados nessa área de pesquisa.

A Seção 2 apresenta uma breve descrição de cada abordagem pesquisada. Ela está dividida em três subseções: Identificação da estrutura, Extração de dados e Recuperação com semântica, de acordo com a intenção dos trabalhos analisados. A Seção 3 mostra um comparativo das abordagens apresentadas e a Seção 4 apresenta a conclusão obtida da pesquisa e algumas propostas de pesquisa na área de recuperação de informação em tabelas na web.

2. Abordagens

Nesta subseção são apresentadas as abordagens estudadas. Os trabalhos analisados foram subdivididos em três classificações de acordo com o conteúdo: identificação de estrutura, extração de dados e recuperação com semântica. Os trabalhos sobre identificação de estrutura visam descobrir como as tabelas estão estruturadas. Trabalhos sobre extração de dados possuem menor foco na estrutura das tabelas e mais na extração dos dados em si. Por fim, os trabalhos sobre recuperação com semântica identificam e anotam informação semântica nas tabelas na web.

2.1. Identificação da estrutura

Vários métodos de classificação de tabelas foram propostos no passado [Wang and Hu 2002, Cafarella and Wu 2008], porém grande parte das pesquisas considera esse um problema binário, ou seja, com duas possibilidades de classificação (genuína x não-genuína, relacional x não-relacional, *layout* x dados). Diferente destes trabalhos anteriores, Crestan e Pantel (2010) propõe uma nova taxonomia feita empiricamente para as tabelas na web.

Crestan e Pantel (2010) explica que a maioria das tabelas na web são utilizadas para definir a estrutura de apresentação de dados na Web, estas são chamadas de tabelas *layout*. Também existem as tabelas que contêm dados relevantes para serem extraídos. Essas tabelas de dados são classificadas pelo trabalho como tabelas relacionais. Assim, são formados dois grandes grupos descritos pelo trabalho. Dentro dessas tabelas relacionais e *layout*, foram feitas classificações mais específicas. As tabelas relacionais são subdivididas entre os seguintes tipos:

- **Vertical**: as tuplas estão dispostas na direção vertical;
- **Horizontal**: as tuplas estão dispostas na direção horizontal;
- **Atributo/Valor**: caso específico de tabela Vertical/Horizontal que não apresenta o assunto na própria tabela, visto que este pode ser obtido do contexto da tabela. Muito utilizadas em especificações técnicas de produtos;
- **Matriz**: este tipo de tabela possui cabeçalhos na vertical e horizontal e no cruzamento entre os dois encontra-se o valor. São utilizados para cruzar dois atributos, por exemplo, número de acidentes por mês para cada estado;
- **Calendário**: tipo especial de tabela Matriz, sendo que um dos atributos é uma data;
- **Enumeração**: este tipo de tabela lista uma série de objetos relacionados;
- **Formulário**: este tipo de tabela é composto por campos de formulário;
- **Outros**: tabelas que não se enquadram nos tipos anteriores.

As tabelas *layout*, por sua vez, podem ser classificadas em dois tipos:

- **Navegação**: tabelas utilizadas para navegação pelo site, por exemplo, categorias de produtos disponíveis;
- **Formatação**: tabelas utilizadas para organizar visualmente os elementos da página.

Crestan e Pantel (2010) propõe, além dessa taxonomia para as tabelas na web, um sistema de classificação supervisionado para tabelas na web. Porém, esse sistema não utiliza toda a taxonomia proposta. Ele classifica as tabelas somente em atributo/valor, layout ou outras.

Lautert et al. (2013) também apresenta uma taxonomia semelhante a Crestan e Pantel (2010). Ele leva em conta alguns tipos de tabelas que não foram citados neste trabalho anterior. Assim, além das já citadas tabelas Verticais, Horizontais e Matriciais, são apresentadas as seguintes classificações:

- **Concisa**: tabela que possui células mescladas;
- **Aninhada**: tabelas dentro de tabelas;
- **Dividida**: tabelas que, por questão de espaço, são divididas horizontal ou verticalmente e suas partes são posicionados lado a lado ou uma sobre a outra;
- **Multivalorada Simples**: tabelas com múltiplos valores de um mesmo domínio em uma célula;
- **Multivalorada Composta**: tabelas com múltiplos valores de diversos domínios em uma célula.

Lautert et al. (2013) além de ser um pouco mais abrangente em sua classificação de tabelas, também apresenta um sistema de classificação supervisionado para as tabelas que, diferente de Crestan e Pantel (2010), utiliza toda a taxonomia proposta.

Son e Park (2013) propõe um método de classificação de tabelas entre relacionais e *layout*. O trabalho explica que as tabelas possuem informações de conteúdo e estruturais. Porém, nem sempre é fácil definir as características estruturais das tabelas. Por isso ele utiliza um algoritmo de análise de padrões denominado *Convolution Kernel*.

Segundo Son e Park (2013), existem dois tipos de informação estrutural. Uma delas consiste nas *tags* que constituem as tabelas e suas relações. O outro tipo consiste no contexto onde a tabela está inserida, ou seja, relações entre as *tags* internas e externas à tabela. Essas características, juntamente com as informações de conteúdo, são processadas separadamente em algoritmos de análise de padrões. Por fim, os padrões são utilizados para treinar máquinas de vetores de suporte (SVM) que verificam quais características melhor definem uma tabela de dados ou de layout. A partir de um modelo treinado é possível utilizar uma SVM para classificar novas tabelas.

Lai (2013) propõe um método de extrair a estrutura das tabelas na web e reorganizá-las para melhorar a acessibilidade aos usuários com deficiência visual. O modo mais comum para essas pessoas acessarem a web é através de softwares que traduzem texto em fala. Mas esses sistemas simplesmente falam o que está na tela de forma linear, o que dificulta o entendimento de tabelas na web. Por isso, o trabalho foca na extração da estrutura dessas tabelas para recuperar suas informações e melhor apresentá-las aos deficientes visuais.

Primeiramente é necessário classificar as tabelas em tabelas de layout e de dados. Para isso são verificadas similaridades das células horizontais e verticais das tabelas, o que

é chamado no trabalho de *Hparallel* e *Vparallel*. Essa similaridade é verificada através de características visuais (como dados CSS) e de texto utilizando funções de similaridade.

Assim são verificadas as células similares horizontalmente e verticalmente, e comparadas com o total de colunas ou linhas. Dependendo de um valor de corte as células são tidas como *Vparallel* ou *Hparallel*. A seguir, a partir da quantidade dessas células similares, compara-se com a quantidade de células totais e dependendo de outro valor de corte a tabela é classificada como *layout* ou de dados. Os valores de corte são obtidos a partir de tabelas de treinamento. O sistema também busca por linhas ou colunas que possuam menor similaridade com o restante da tabela para identificar cabeçalhos ou rodapés.

Assim, identificadas as estruturas das tabelas, é possível transformá-las em estruturas mais fáceis de serem interpretadas por sistemas de leitura para deficientes visuais. Lai (2013) explica que os sistemas de leitura interpretam essas tabelas lendo linha por linha ou coluna por coluna dependendo da disposição da tabela, e associa o cabeçalho à célula que está sendo lida.

2.2. Extração de dados

Embley et al. (2011) mostra uma forma de manipular tabelas na web visando indexar seus valores relacionando-os com os cabeçalhos. Este trabalho trata em específico tabelas que segundo Crestan e Pantel (2010) possuem a classificação de Matriz. O trabalho introduz o conceito de *Header Path*, que organiza de forma hierárquica os cabeçalhos de uma tabela, dos níveis superiores até os inferiores. A Tabela 1, por exemplo, apresenta os cabeçalhos 'Temperature' e 'Day' e suas especializações que são 'Min', 'Max', 'Monday', 'Tuesday' e 'Wednesday'. Portanto o *Header Path* para esse caso seria:

- Temperature
  - Min
  - Max
- Day
  - Monday
  - Tuesday
  - Wednesday

Tabela 1. Tabela com cabeçalhos aninhados

		Temperature	
		Min	Max
Day	Monday	11C	22C
	Tuesday	9C	19C
	Wednesday	10C	21C

Para criar esses *Header Paths*, são processados arquivos CSV com as tabelas já extraídas da web. Os arquivos CSV são processados por rotinas escritas em Python de forma a encontrar os cabeçalhos e criar a estrutura dos *Header Paths*. Esses *Header Paths* podem ser criados para cabeçalhos na vertical, horizontal ou ambos. A partir disso, o trabalho apresenta uma linguagem de consulta baseada nos operadores lógicos de união e intersecção.

Por exemplo, novamente na Tabela 1, pode-se aplicar uma consulta na forma `(Temperature * Min) + (Temperature * Max)`, que seria uma união das colunas Min e Max, resultando no conjunto de valores completo de valores exibido na tabela. Porém, se aplicarmos a consulta da forma `(Temperature * Min) * (Day * Monday)` seleciona-se somente a célula da intersecção entre `Day = Monday` e `Temperature = Min`, no caso: 11C.

Por fim, a partir desses *Header Paths* e dessa linguagem de consulta, o trabalho demonstra como é possível gerar uma nova estrutura relacional, de forma a permitir consultas SQL sobre os dados extraídos da tabela web.

Nagy et al. (2011) é uma continuação do trabalho apresentado por Embley et al. (2011). Ele mostra um tratamento dado a tabelas mais complexas. Um exemplo dado pelo trabalho é a Tabela 2 que possui a célula no canto superior esquerdo com valor 'A', onde não se sabe se é um cabeçalho de linha ou de coluna. O trabalho explica que na grande maioria dos casos observados, essa célula de canto é um cabeçalho de linha. Por isso, nesses casos, esta célula é aceita como cabeçalho de linha.

Tabela 2. Tabela com canto indefinido [Nagy et al. 2011]

A	**B1**	**B2**
C1	D11	D12
C2	D21	D22

Também em casos onde a tabela não está na forma de matriz, o trabalho propõe utilizar os valores como índices e construir os *Header Paths* sobre eles. Porém, em casos como da Tabela 3 onde, por exemplo, os valores de 'State' se repetem, seria necessário utilizar mais de uma coluna de valores como índice. No caso da Tabela 3 seriam utilizadas as três primeiras colunas para construir o *Header Path* vertical.

Tabela 3. Índice com múltiplas colunas [Nagy et al. 2011]

State	Company Name	Plant I.D.	Plant Name	County	Biomass / Coal Cofiring Capacity	Total Plant Capacity
AL	DTE Energy Services	50407	Mobile Energy Services LLC	Mobile	91	91
AL	Georgia-Pacific Corp	10699	Georgia Pacific Naheola Mill	Choctaw	31	78
AL	International Paper Co	52140	International Paper Prattville Mill	Autauga	49	90
AZ	Tucson Electric Power Co	126	H Wilson Sundt Generating Station	Pima	173	559
ROWS OMITTED						
MI	S D Warren Co	50438	S D Warren Muskegon	Muskegon	51	51
MI	TES Filer City Station LP	50835	TES Filer City Station	Manistee	70	70
MN	Minnesota Power Inc	10686	Rapids Energy Center	Itasca	27	28
MN	Minnesota Power Inc	1897	M L Hibbard St	Louis	73	123
ROWS OMITTED						

Para solucionar esses problemas, o trabalho mostra uma solução supervisionada para correção dos *Header Paths*. A geração dos *Header Paths* em si, é feita do mesmo modo como é mostrado em Embley et al. (2011). Porém, é acrescentada a verificação pelo usuário que precisa informar ao software se a detecção de cabeçalhos e dados está correta.

Ainda na linha de extração de dados, Cafarella et al. (2009) descreve o sistema *WebTables*, desenvolvido para extrair dados estruturados apresentados na forma de tabelas na web. O sistema *WebTables* utiliza uma combinação de classificadores para recuperar as tabelas relacionais de um conjunto de tabelas na web. Após essa classificação é obtido um grande conjunto de dados relacionais.

São apresentados dois passos para a obtenção das bases de dados relacionais a partir de tabelas HTML cruas. Primeiramente, uma amostra de tabelas é classificada em relacional e *layout*. Para isso são utilizadas heurísticas escritas manualmente para filtrar tabelas com características mais específicas. Por exemplo, as que possuem somente uma linha ou somente uma coluna, tabelas utilizadas para mostrar calendários e tabelas com formulários.

O segundo passo é rotular as tabelas restantes como relacionais e *layout* usando classificadores treinados. Esses classificadores se baseiam em aspectos pré definidos que caracterizam cada tipo de tabela, como número de linhas, colunas, células vazias e etc.

Depois desse filtro o sistema tenta recuperar os metadados de cada relação. A principal fonte de metadados apresentada são os cabeçalhos das tabelas. Para recuperá-los foi utilizado outro classificador treinado que compara a primeira linha de cada coluna com o corpo da tabela para detectar se existe cabeçalho ou não.

Sardi Mergen et al. (2010) apresenta um sistema de busca por dados em tabelas na web que utiliza uma linguagem de consulta simples e que possibilita uma seleção mais precisa de dados obtidos de tabelas na web.

Primeiramente o sistema indexa tabelas na web a partir de seus atributos. Não está claro no trabalho como são obtidos esses atributos, mas eles são referentes aos dados contidos nas tabelas, como por exemplo, título do filme, ano de lançamento e etc. Esses índices são criados na forma $atributo \rightarrow valores \rightarrow tabelas$ e $atributo \rightarrow tabelas \rightarrow valores$. Também são identificados os tipos de valores presentes nas tabelas. A partir disso, o usuário pode criar consultas selecionando atributos e especificando condições de consulta.

2.3. Recuperação com Semântica

Venetis et al. (2011) descreve um sistema que busca recuperar a semântica das tabelas na web acrescentando nelas anotações. O objetivo principal desse sistema é contribuir com as buscas na web.

O trabalho explica que os motores de busca da web tratam as tabelas como documentos de texto comuns. Porém, as tabelas poderiam ser melhor aproveitadas se fossem tratadas de forma diferente, observando a semântica contida nelas. Por exemplo, muitas vezes as tabelas não possuem cabeçalhos explícitos demostrando o assunto tratado na mesma. Com a recuperação da semântica dessas tabelas seria possível utilizá-las como resultado de buscas mesmo elas não contendo dados explicitamente relacionados à palavra chave. O conhecimento da semântica das tabelas também permitiria a aplicação de operações de combinação de tabelas, como *join* e *union*.

Assim, para recuperar a semântica das tabelas, o trabalho propõe a criação de duas bases de dados obtidas automaticamente a partir de textos da web. A base *isA* com pares na forma (classe,instância) é obtida utilizando basicamente padrões linguísticos. A outra possui relações em triplas na forma (argumento1,predicado,argumento2). Ela é obtida utilizando o TextRunner [Banko and Etzioni 2008] como ferramenta de extração de informação a partir de textos.

Segundo o trabalho, uma coluna *A* é rotulada com uma classe *C* da base de dados *isA*, se uma fração substancial das células na coluna *A* são rotuladas com a classe *C* na

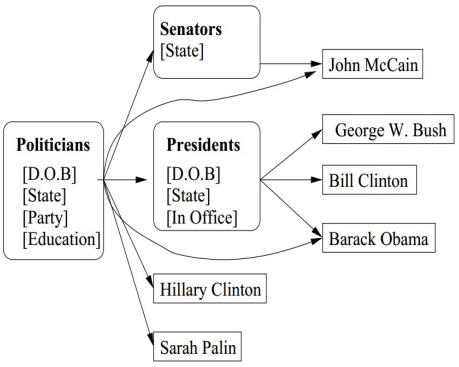

Figura 1. Um fragmento do Probase [Wang et al. 2012]

base de dados *isA*. De forma semelhante, a relação entre duas colunas A e B é rotulada com R se uma fração substancial de pares de valores de A e B ocorre nas extrações na forma (a,R,b) na base de dados de relações.

Wang et al. (2012) também propõe uma forma de recuperar a semântica das tabelas na web. Ele cita que a chave para entender as tabelas é saber qual conceito melhor descreve as entidades e atributos contidos nas tabelas. Para encontrar esses conceitos, o trabalho utiliza uma base de conhecimento chamada Probase [Wu et al. 2012]. Esta base contém conceitos, atributos e entidades. A Figura 1 mostra um exemplo disso. Ela possui conceitos, como 'Politicians', 'Presidents' e 'Senator', atributos como 'State', 'Party', 'D.O.B' e entidades como 'John McCain' e 'Bill Clinton'. As linhas e colunas das tabelas são comparadas a essa base e é acrescentada semântica às tabelas.

Feita essa detecção da semântica das tabelas, o trabalho apresenta um sistema de busca semântica em tabelas. Esse sistema, ao invés de recuperar páginas inteiras a partir de palavras chave, retorna tuplas e atributos específicos contidos em tabelas, semelhante a consultas SQL.

Fan et al. (2013) apresenta um sistema semi-supervisionado para encontrar conceitos em tabelas. O objetivo do trabalho é descobrir esses conceitos e utilizá-los em comparações de tabelas na web. Esse processo seria o mesmo que recuperar o esquema das tabelas e realizar um *schema matching*.

Para encontrar os conceitos de um grupo de tabelas, Fan et al. (2013) utiliza uma base de conhecimento, buscando adicionar conceitos a cada coluna dessas tabelas. O sistema verifica o grau de dificuldade em se descobrir os conceitos de cada coluna. Para isso, é utilizada uma função de similaridade. Essa função tem como entradas os valores de uma coluna A e um conceito C e retorna a probabilidade de A e C serem relacionados. Com isso, cada coluna recebe pesos associados a conceitos. Quanto mais idênticos são os pesos, maior a dificuldade em classificar a coluna. Com isso é determinado um grau de dificuldade para cada coluna. Para os casos considerados difíceis, o usuário deve realizar a classificação manualmente.

A seguir Fan et al. (2013) explica que saber os conceitos de algumas colunas pode ajudar a descobrir os conceitos de outras colunas. Por isso ele calcula o grau de influência de uma coluna sobre as outras. Descobertos os conceitos de cada tabela, é possível compará-las.

3. Comparativo das Abordagens

Esta seção apresenta um comparativo entre as abordagens, destacando suas similaridades e diferenças. A Tabela 4 resume este comparativo.

Tabela 4. Comparativo das Abordagens

Trabalho	Abordagem	Tipo de Tabelas Processadas	Processamento	Objetivo	Estratégia
[Crestan and Pantel 2010]	Estrutura	Dados e Layout	Supervisionado	Classificação	Árvore de decisão
[Lautert et al. 2013]	Estrutura	Todos os tipos dentro da taxonomia	Supervisionado	Classificação	Rede neural
[Son and Park 2013]	Estrutura	Dados e Layout	Semi-supervisionado	Classificação	*Convolution kernel* (análise de padrões)
[Lai 2013]	Estrutura	Horizontais, Verticais, Matriciais	Automático	Identificação da Estrutura	Funções de similaridade
[Embley et al. 2011]	Extração de dados	Matriciais	Automático	Extração de dados	Header Paths, fatoração algébrica
[Nagy et al. 2011]	Extração de dados	Horizontais, Verticais, Matriciais	Supervisionado	Extração de dados	Header Paths, fatoração algébrica
[Cafarella et al. 2009]	Extração de dados	Horizontais	Semi-supervisionado	Extração de dados	Classificador treinado
[Sardi Mergen et al. 2010]	Extração de dados	Horizontais	Semi-supervisionado	Extração de dados	Índices
[Wang et al. 2012]	Recuperação com semântica	Horizontais	Automático	Busca semântica	Base de Conhecimento (Probase)
[Venetis et al. 2011]	Recuperação com semântica	Horizontais	Automático	Busca semântica	Base de Conhecimento (própria)
[Fan et al. 2013]	Recuperação com semântica	Horizontais	Semi-supervisionado	*Schema Matching*	Base de Conhecimento e funções de similaridade

As abordagens de identificação de estrutura são úteis no sentido de descobrir como a tabela está disposta ou se ela possui dados ou não. Isso pode facilitar a descoberta das informações contidas ali. Dentro dessas abordagens, Lautert et al. (2013) é a que mais se destaca por sua classificação mais completa. Vale lembrar que Crestan e Pantel (2010) também apresenta uma taxonomia bastante abrangente, porém seu classificador não utiliza a taxonomia completa em seu sistema de classificação supervisionado, ao contrário de Lautert et al. (2013). Son e Park (2013) se destaca por seu algoritmo de detecção de padrões, tornando menos necessária a interação humana com entradas de dados. Porém, esse trabalho classifica somente as tabelas em dados e layout e dentro das tabelas de dados existe uma grande variedade de disposições de tabelas. Lai (2013) possui uma abordagem relativamente simples de encontrar linhas, colunas e cabeçalhos nas tabelas, não utilizando inteligência artificial para isso.

Quanto às abordagens de extração de dados, todas exceto Sardi Mergen et al. (2010) buscam extrair os dados e inseri-los em bases de dados relacionais. Embley et al. (2011) e Nagy et al. (2011) com sua linguagem de consulta e seu *Header Path*, poderiam ser utilizados no acesso aos dados diretamente das tabelas na web, porém esses trabalhos utilizam diretamente dados já extraídos para arquivos CSV. Cafarella et al. (2009) por sua vez, extrai os dados diretamente das páginas web, utilizando um classificador treinado. Sardi Mergen et al. (2010) se destaca por obter informações da web em tempo real, sem a necessidade de guardar as tabelas em um repositório. Isso torna os dados sempre atualizados de acordo com as páginas web.

Os trabalhos de recuperação com semântica são bastante semelhantes entre si. Todos utilizam uma base de conhecimento e buscam acrescentar anotações ou conceitos às tabelas. Wang et al. (2012) e Venetis et al. (2011) utilizam as anotações semânticas para aprimorar as buscas por tabelas na web. Em termos de precisão, Wang et al. (2012) apresentou melhores resultados nos experimentos de busca, porém os dois trabalhos possuem abordagens de busca um pouco diferentes. Enquanto Venetis et al. (2011) retorna tabelas inteiras, Wang et al. (2012) retorna tuplas retiradas das tabelas. Fan et al. (2013) por sua vez, possui seu foco em descobrir os esquemas das tabelas e após isso, tenta descobrir outras tabelas com mesmo esquema. Vale lembrar que dos trabalhos de recuperação de semântica estudados, Fan et al. (2013) é o único que não possui um sistema automático.

Notou-se na grande maioria dos trabalhos, o uso de ferramentas de inteligência artificial. Dentre essas ferramentas, as mais utilizadas foram bases de conhecimento e sistemas treinados. Também notou-se o uso explícito de funções de similaridade em pelo menos dois trabalhos. Porém, em nenhum trabalho foram encontradas verificações de sinônimos.

Em praticamente todos os trabalhos estudados, apesar de alguns não darem tanto foco nisso, notou-se a necessidade de separar tabelas de dados e de layout, porém, cada trabalho apresenta uma forma um pouco diferente de realizar esse processo. Também em muitos trabalhos notou-se a necessidade de detecção dos cabeçalhos contidos nas tabelas, sendo que em alguns casos, eles não estão presentes diretamente nas tabelas, e sim, implícitos no contexto em que as tabelas estão inseridas.

4. Conclusão

Este trabalho apresenta abordagens na literatura relacionadas à recuperação de informação contida em tabelas na web. Uma breve descrição de cada trabalho é mostrada, vantagens e desvantagens são apontadas e um comparativo foi produzido. Os trabalhos possuem abordagens que diferem umas das outras, porém cada um contribui de certa forma para recuperar informações contidas nas tabelas.

Alguns temas, como a detecção de estrutura das tabelas e classificação entre tabelas de layout e de dados são de grande necessidade na manipulação de tabelas na web. Também existe uma grande tendência no uso de ferramentas de inteligência artificial e funções de similaridade e as abordagens de extração de dados e anotação semânticas, em sua essência, são bastante semelhantes entre si.

Porém, alguns temas poderiam ser melhor explorados, como a detecção de temas das tabelas a partir da exploração do contexto onde estão inseridas, extratores baseados nas classificações propostas, além do uso mais efetivo de dicionários nas anotações semânticas. Outras abordagens promissoras seriam a análise das tabelas na web sem a necessidade de extraí-las para bases de dados relacionais, execução de junções entre tabelas e a descoberta de tabelas similares para fins de consulta unificada.

Referências

Banko, M. and Etzioni, O. (2008). The tradeoffs between open and traditional relation extraction. In *ACL*, pages 28–36.

Cafarella, M. J., Madhavan, J., and Halevy, A. (2009). Web-scale extraction of structured data. *SIGMOD Rec.*, 37(4):55–61.

Cafarella, M. J. and Wu, E. (2008). Uncovering the relational web. In *In under review*.

Crestan, E. and Pantel, P. (2010). A fine-grained taxonomy of tables on the web. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM '10, pages 1405–1408, New York, NY, USA. ACM.

Embley, D. W., Krishnamoorthy, M., Nagy, G., and Seth, S. (2011). Factoring web tables. In *Proceedings of the 24th international conference on Industrial engineering and other applications of applied intelligent systems*, IEA/AIE'11, pages 253–263, Berlin, Heidelberg. Springer-Verlag.

Fan, J., Lu, M., Ooi, B. C., Tan, W.-C., and Zhang, M. (2013). A hybrid machine-crowdsourcing system for matching web tables. Technical report, Technical Report.

Lai, P. P. Y. (2013). Adapting data table to improve web accessibility. In *Proceedings of the 10th International Cross-Disciplinary Conference on Web Accessibility*, W4A '13, pages 33:1–33:4, New York, NY, USA. ACM.

Lautert, L. R., Scheidt, M., and Dorneles, C. F. (2013). Web table taxonomiy and formalization. *SIGMOD*.

Nagy, G., Seth, S. C., Jin, D., Embley, D. W., Machado, S., and Krishnamoorthy, M. S. (2011). Data extraction from web tables: The devil is in the details. In *ICDAR*, pages 242–246. IEEE.

Sardi Mergen, S. L., Freire, J., and Heuser, C. A. (2010). Indexing relations on the web. In *Proceedings of the 13th International Conference on Extending Database Technology*, EDBT '10, pages 430–440, New York, NY, USA. ACM.

Son, J.-W. and Park, S.-B. (2013). Web table discrimination with composition of rich structural and content information. *Appl. Soft Comput.*, 13(1):47–57.

Venetis, P., Halevy, A., Madhavan, J., Paşca, M., Shen, W., Wu, F., Miao, G., and Wu, C. (2011). Recovering semantics of tables on the web. *Proc. VLDB Endow.*, 4(9):528–538.

Wang, J., Wang, H., Wang, Z., and Zhu, K. Q. (2012). Understanding tables on the web. In *Proceedings of the 31st international conference on Conceptual Modeling*, ER'12, pages 141–155, Berlin, Heidelberg. Springer-Verlag.

Wang, Y. and Hu, J. (2002). A machine learning based approach for table detection on the web. In *Proceedings of the 11th International Conference on World Wide Web*, WWW '02, pages 242–250, New York, NY, USA. ACM.

Wu, W., Li, H., Wang, H., and Zhu, K. Q. (2012). Probase: A probabilistic taxonomy for text understanding. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, SIGMOD '12, pages 481–492, New York, NY, USA. ACM.