Implementação de um esquema de extração de dados tabulares da web

Stéphanie S. Leal, Marcelo M. Scheidt,
Carina F. Dorneles

Relatório Técnico INE 003/2015

Universidade Federal de Santa Catarina
Departamento de Informática e Estatística

Implementação de um esquema de extração de dados tabulares da web

Stéphanie S. Leal
Departamento de Informática e Estatística
Universidade Federal de Santa Catarina - UFSC
stephaniesleal@gmail.com

Marcelo M. Scheidt
Departamento de Informática e Estatística
Universidade Federal de Santa Catarina - UFSC
leloscheidt@gmail.com

Carina F. Dorneles
Departamento de Informática e Estatística
Universidade Federal de Santa Catarina - UFSC
dorneles@inf.ufsc.br

ABSTRACT

A large portion of the current information are distributed on the web in a non-structured way without being stored in any database, e.g. WebTables. Considering that the computational systems operate efficiently over structured data, many studies are performed to perform the extraction of this non-structured data to structured data models. This present article has the goal to demonstrate the implementation of a propose [1] of extraction of WebTables which includes an algorithm capable of to partition the rows of a table in a compartment per similar characteristics. The goal of the developed algorithm, called by the authors as logarithmic binning, is to find similarity among the rows to classify and extract them precisely. The result generated by the algorithm consists in a group of values that when united, will identify the role of each row has in the table, making the extraction process reachable in an automated way.

RESUMO

Grande parcela das informações atuais se encontram distribuídas na web de forma não estruturada sem estarem armazenadas em qualquer base de dados, como por exemplo em *WebTables*. Considerando que os sistemas computacionais operam eficientemente sobre dados estruturados, muitos estudos são realizados para realizar a extração destes dados não estruturados para modelos estruturados de dados. O presente artigo possui a finalidade de demonstrar a implementação de uma proposta [1] de extração de *WebTables* que inclui um algoritmo capaz de particionar as linhas de uma tabela em compartimentos por características semelhantes. O objetivo do algoritmo desenvolvido, denominado pelos autores de *logarithmic binning*, é encontrar a similaridade entre as linhas para poder classificá-las e extrai-las de forma precisa. O resultado gerado pelo algoritmo consiste em um conjunto de valores que reunidos identificará o papel que cada linha tem na tabela, tornando assim o processo de extração alcançável de forma automatizada.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**] : Search process.

General Terms

Algorithms, Measurement, Performance, Reliability, Experimentation, Verification.

Keywords

WebTables, Attributes, Logarithm Binning, Similarity.

1. INTRODUÇÃO

Na concepção do ser humano tudo é informação, relevante ou não, no entanto no ambiente das máquinas, para que seja possível formular uma definição completa do significado de uma informação é necessário que os dados estejam organizados de forma estruturada para serem recuperados. Atualmente, grande parte dos dados não se encontra armazenada em bases de dados, ou seja, a grande maioria dos dados não é estruturada e está distribuída na web, tornando-a um enorme repositório de dados. Este repositório, em função de sua estrutura despadronizada e sem semântica definida, torna o reconhecimento dos dados por máquinas uma tarefa complexa, impedindo a manipulação e a exploração dos dados eficientemente.

Pelo fato da complexidade gerada em função da constituição não estrutura das tabelas, pesquisas foram feitas e trabalhos realizados, (p. ex., [5, 6, 7, 8]) em prol de encontrar uma maneira para extrair dados não estruturados e torná-los facilmente reconhecidos por máquinas e assim manipulados eficientemente. A grande maioria das pesquisas relacionadas com a extração dos dados tabulares levaram em consideração dados presentes em tabelas simples como em [6], das quais possuem uma linha de cabeçalho e uma ou mais linhas que correspondem aos valores dos dados. De fato muitas tabelas da web possuem estrutura simples, entretanto existem tabelas com estruturas complexas que possuem alto grau de informações úteis a serem extraídas. Este aspecto foi o embasamento da pesquisa do artigo *Schema Extraction for Tabular Data on the Web* [1], no qual foi desenvolvido um método de extração considerando também as tabelas mais complexas.

A proposta do trabalho resume-se em avaliar todas as linhas da tabela e classificá-las em categorias que representam seu papel na tabela para depois extrair os dados a fim de alcançar um alto nível de acurácia tanto na extração como na interpretação das tabelas. No artigo [1], os autores buscam organizar as linhas heterogêneas em compartimentos separados de acordo com a análise do conjunto das características de células constituintes da linha, para que posteriormente cada linha possua um conjunto de valores decorrentes do cálculo do *logarithmic binning* desenvolvido pelos autores. A partir destes valores é possível classificar cada linha

com um rótulo, sendo este o papel que descreve a função de cada linha na tabela. Para esta etapa de classificação os autores indicam um classificador capaz de tornar o processo automatizado.

O trabalho de Iniciação Científica desenvolvido teve como objetivo a implementação da proposta dos autores Adelfio e Samet (2013) para posterior comparação com proposta a ser desenvolvida pelo grupo de pesquisa de banco de dados da UFSC. Desta forma, neste artigo é descrito o processo de implementação de extração dos dados tabulares dos autores supracitados.

O artigo está organizado como segue. Na Seção 2 é descrito o método de extração proposto, a Seção 3 descreve todo o processo de implementação do *logarithmic binning* bem como o processo para realizar a classificação, na Seção 4 está relatado os experimentos realizados e os resultados encontrados e na Seção 5 está descrita a conclusão do trabalho.

2. MÉTODO DE EXTRAÇÃO

O método de extração criado pelos autores Marco D. Adelfio e Hanan Samet no trabalho [1] consiste na classificação de cada célula de uma tabela quanto a uma série de características. Estas características combinadas constituem os atributos das células e a partir destes é realizado o cálculo do *logarithmic binning*, desenvolvido pelos autores, para posteriormente processar estes dados em um classificador treinado baseado em campos condicionais aleatórios [2]. A saída do classificador permite discernir tabelas relacionais e não relacionais, bem como a classificação de cada linha da tabela. A fim de demonstrar a veracidade da acurácia de tal método, foi implementado em três módulos o processo de extração dos dados de tabelas, consistindo em: classificação dos atributos de células, aplicação do *logarithmic binning* e classificação das linhas através do classificador treinado.

3. IMPLEMENTAÇÃO

Esta Seção descreve a implementação realizada neste trabalho utilizando como base a teoria proposta pelos autores no artigo base [1]. Foi implementado todo o algoritmo para o cálculo do *logarithmic binning* para a realização da classificação. O processo está representado no diagrama da figura 1 abaixo.

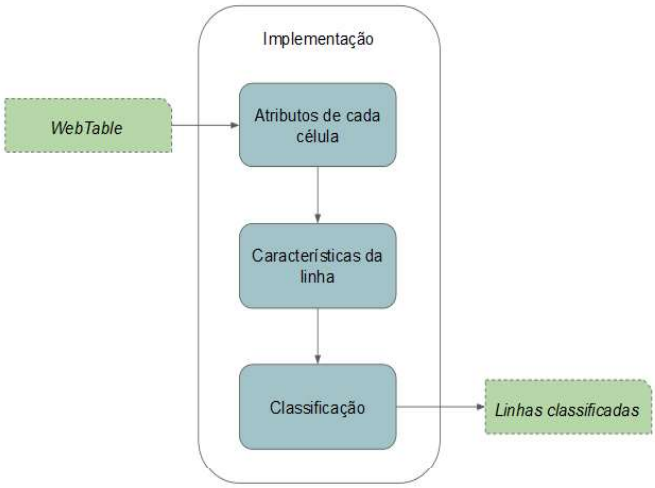

Figura 1. Processo de Implementação.

No diagrama da figura 1 está descrito a ordem em que os módulos são executados. Em síntese, a partir de uma *WebTable* extraída da web aplica-se o módulo Atributos de célula que trará como resultado os atributos de cada célula da tabela, esta saída será a entrada para o módulo seguinte, Aplicação do *logarithmic binning,* este por sua vez, retorna as características das linhas e por último estes valores são passados para a classificação da linha.

3.1 Atributos de células

Este módulo da implementação consiste em extrair as características de cada célula, para este processo contou-se com o auxílio da ferramenta HtmlUnit [3], considerando que grande quantidade de páginas da web são estruturadas em código HTML.

As informações de cores, alinhamentos, aninhamentos, formatação de texto, todas estas características e muitas outras são definidas através do recurso de *tags* que o HTML provém. O HtmlUnit auxiliou no processo de percorrer as células de cada linha e verificar os seus atributos, para isto foram criados 15 métodos, divididos em 3 categorias: layout, estilo e valor.

Na categoria layout foi verificado se a célula possui as seguintes características: de ser aninhada, de estar alinhada e qual o seu alinhamento. Na categoria estilo foram verificadas as seguintes características: negrito, itálico, sublinhado, colorido, a fonte e o formato. Por fim na categoria valor avaliou-se se a célula é vazia, se é texto, número, data, se é um texto longo ou curto baseado na média realizada avaliando o tamanho dos textos em todas as células, e por fim se a célula possui a palavra "total". Este processo pode ser visto no pseudo-código 1 abaixo.

Algoritmo 1 Extração das Características das Linhas.

```
begin
  for each cell in table do:
    bold <- check if cell has bold attribute
    italic <- check if cell has italic attribute
    align <- get cell aligment
    …
    save all on cell object
  end for
  end
```

Após classificar os atributos de cada célula, foi avaliado a similaridade, o qual consiste em para uma determinada célula, as células acima e abaixo são verificadas para cada uma característica, basicamente verifica-se se os atributos de uma célula são iguais a célula superior e diferentes a célula inferior, para o caso da primeira e última célula foram comparadas com a célula inferior e superior respectivamente. Isto é feito para verificar o padrão das células umas as outras, por exemplo, para separar células do cabeçalho da tabela das células que possuem dados. Após esta etapa cada célula possui suas características computadas.

3.2 Aplicação do *logarithmic binning*

Neste módulo da implementação foi realizado o agrupamento de características por linha, para isto, nesta etapa foi aplicado o *logarithmic binning* para agrupar as células de uma mesma linha para cada característica.

Isto é feito utilizando a formula abaixo para cada característica da célula, sendo *c* o número de células que possuem um atributo e *r* o número de células total da linha. Este cálculo resultará em dois valores, (*a*, *b)* como um par ordenado para cada atributo da célula.

$$a = \begin{cases} 0, & \text{if } c = 0 \\ \lfloor log_2(c) + 1 \rfloor, & \text{if } 0 < c \leq r/2 \\ \lfloor log_2(r - c) + 1 \rfloor^-, & \text{if } r/2 < c < r \\ 0^-, & \text{if } c = r \end{cases}$$

$$b = \lfloor log_2(r) \rfloor$$

Figura 2. Fórmula para o cálculo do *logarithmic binning* [1].

O algoritmo informa a semelhança entre duas linhas utilizando os valores *a* e *b,* ou seja, duas linhas são consideradas semelhantes quando podem ser alocadas no mesmo compartimento de acordo com o diagrama de compartimentos abaixo.

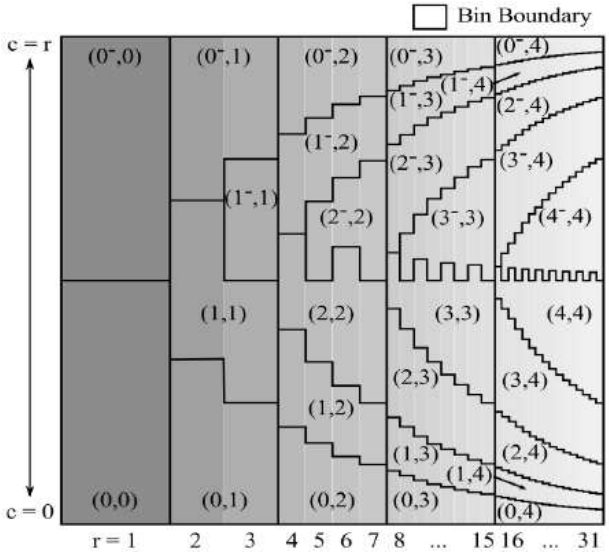

Figura 3. Diagrama de compartimentos [1].

Este processo pode ser visto no pseudo-código 2 abaixo.

Algoritmo 2 Cálculo do *Logarithmic Binning*.

```
begin
for each cell in table do:
  c <- count cells with the same atributte in same row
  r <- count total cells in row
  a <- calculate a (c,r)  // using formula rules.
  b <- calculate b (r)
end for
end
```

O objetivo deste esquema *binning* consiste em diferenciar larguras entre tabelas, considerando que tabelas com larguras diferentes geram diferentes distribuições de linhas, e seus recursos devem ser divididos em compartimentos separados. Além disso, é possível destacar linhas uniformes; as linhas uniformes devem estar em compartimentos separados das linhas mais heterogêneas. Ao fim desta etapa, cada linha possui um valor de *a* e *b* relacionado a cada característica.

3.3 Classificação das linhas

Para a última etapa da implementação foi necessário categorizar cada linha com base nas características agrupadas. O algoritmo define valores para os rótulos de linha de acordo com o significado daquela linha na tabela. Abaixo a tabela 1 com os rótulos propostos.

Tabela 1. Classificação dos rótulos de linhas

Rótulo	Definição
H	Header (Cabeçalho), as linhas rotuladas cabeçalho contêm valores de células que descrevem os valores contidos nas linhas subsequentes da coluna.
D	Data (Dado), linhas rotuladas como dado são as que contêm registros de dados.
T	Title (Título), linhas rotuladas como título descrevem o conjunto de dados encontrados na tabela.
G	Group (Grupo), as linhas de grupo provêm categorias para as linhas subsequentes.
A	Aggregate (Agregação), estas linhas são as que agregam o resultado das linhas anteriores.
N	Non-relational (Não relacional) são as linhas que não contribuem com dados ou com valores para a estrutura da tabela, como as notas.
B	Blank (Em branco), linhas que possuem células vazias.

Para classificar as linhas de acordo com os rótulos é necessário treinar uma classificador *Condicional Random Fields* (CRF) [1] especificadas pelos autores. Este é um modelo de classificador para rotular entidades de acordo com múltiplas características, é principalmente utilizado no tratamento de linguagens naturais. O modelo descrito pelos autores revelou possuir um alto nível de acurácia, sendo indicado para a classificação das linhas das tabelas.

Visto isto, como medida de treinamento foram reunidas 10 tabelas de domínios diferentes, classificadas manualmente e enviadas as informações ao classificador para treiná-lo e depois classificar de forma automatizada as linhas, entretanto este processo não obteve resultados satisfatórios quanto a compreensão da classificação do CRF em virtude de sua complexidade e do entendimento para manuseá-lo, bem como a falta de exemplos práticos e bibliotecas mais robustas sendo assim, a avaliação baseou-se nas classificações realizadas manualmente.

Após este procedimento foram coletadas um total de 2155 tabelas de três domínios distintos: filmes, carros e enciclopédias digitais e aplicado todos os módulos da implementação. Na Seção 4 são relatados os experimentos bem como os resultados alcançados com testes preliminares.

4. EXPERIMENTOS

Os experimentos foram realizados com o objetivo de testar a veracidade do cálculo do *logarithmic binning* para uma quantidade grande de valores. Para isso, os seguintes passos foram

executados: extração das tabelas da web, união do código HTML das tabelas em um único arquivo texto, execução do módulo para coletar os atributos das células, aplicação do cálculo do *logarithmic binning* e por fim a realização da classificação manual com base nos valores encontrados.
Além disso, os experimentos foram executados em um conjunto de dados de 10 páginas Web distintas, contendo ao todo 2155 *WebTables.*
O processo de extração de tabelas da web contou com o auxílio do crawler desenvolvido no trabalho [9], otimizando desta forma a busca por tabelas da web. Em seguida, foram aplicados ao código HTML extraído destas tabelas os módulos implementados neste trabalho, e resultando em um arquivo texto constituído com os valores gerados no cálculo do *logarithmic binning*. A partir deste resultado, foi selecionada uma amostra das tabelas e classificadas manualmente. Finalmente foram avaliados os resultados e constatado a eficiência do algoritmo, tais resultados encontram-se detalhados na Seção a seguir.

4.1 RESULTADOS

Como resultado dos experimentos realizados foram levadas em consideração duas métricas: precisão e overhead. Cada qual detalhada a seguir:

4.1.1 Precisão

Para definir a precisão do algoritmo foram selecionadas 20 tabelas extraídas de domínios diferentes e separadas 10 tabelas para treino e 10 tabelas para teste. Para as tabelas de treino, as linhas foram classificadas manualmente observando a tabela em si e posteriormente foram verificados os valores do par ordenado para cada atributo de célula e assim relacionado com o rótulo já definido, utilizando como auxílio o diagrama de compartimentos, os valores e os rótulos associados foram tomados como referência para realizar a classificação das tabelas de teste. Para o segundo grupo de tabelas, as de teste, a classificação ocorreu relacionando somente os valores do par (*a, b*) com os rótulos. Ao fim deste teste foram avaliadas as linhas classificadas corretamente do total de linhas resultando em uma precisão de 68% do algoritmo implementado. Entretanto faz-se necessário uma classificação em larga escala a partir do uso de um classificador para observar se esta precisão se mantém com uma quantidade significativa de tabelas.

4.1.2 Overhead

Esta medida relaciona o tempo necessário para calcular o *logarithmic binning* de cada tabela. O *logarithmic binning* realizou o cálculo em aproximadamente 108,76 segundos para as 2155 tabelas coletadas, ou seja, o cálculo do esquema *binning* leva em torno de 0,05 segundos para cada tabela. Entretanto como foram classificadas manualmente as tabelas a partir dos valores extraídos, não é possivel informar com precisão o tempo relacionado com a classificação dos rótulos para cada tabela.

5. CONCLUSÃO

O desenvolvimento do *logarithmic binning* permitiu visualizar na prática a teoria proposta pelos autores no artigo utilizado como base para este trabalho. A partir das tabelas extraídas, avaliando uma amostra e classificando-as com os dados obtidos do cálculo do algoritmo implementado, pode-se perceber a eficiência do comportamento desta proposta de classificação, partindo do princípio da separação das características diferentes de cada linha em compartimentos distintos. Como trabalho futuro, cita-se a proposta da realização de experimentos com um classificador, como o CRF citado no artigo, para comprovar em termos numéricos o nível de eficiência da classificação em larga escala.

6. AGRADECIMENTOS

Meu agradecimento ao CNPq pela concessão da bolsa de estudos de Iniciação Cientifica na qual foi possível adquirir novos aprendizados e aprimorar técnicas, bem como a realização de experimentos até então somente citados em teoria. Ao Laboratório para Integração de Sistemas de Informação e Aplicações Avançadas (LISA), ao Departamento de Informática e Estatística (INE) e a Universidade Federal de Santa Catarina (UFSC) por toda a infraestrutura necessária.

7. REFERÊNCIAS BIBLIOGRÁFICAS

[1] Adelfio D. Marco, Samet Hanan. 2013. Schema extraction for tabular data on the web. In ACM, University of Maryland, USA.

[2] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In ICML,Williamstown, Mas-sachussetts, USA.

[3] HtmlUnit. The HtmlUnit Project. Disponível em <http://htmlunit.sourceforge.net/>.

[4] Cafarella, M. J., Halevy A. Y. Wang Z. D., Wu E. Zhang Y. 2008. WebTables: exploring the power of tables on the web. In ACM, University of Washington, USA.

[5] M. J. Cafarella, A. Halevy, D. Z. Wang, E. Wu, and Y. Zhang. 2008. Uncovering the relational web. In WebDB, Vancouver, Canada.

[6] G. Limaye, S. Sarawagi, and S. Chakrabarti. 2010. Annotating and searching web tables using entities, types and relationships.In ACM, IIT Bombay, India.

[7] H. H. Chen, S.-C. Tsai, and J.-H. Tsai. 2000. Mining tables from large scale HTML texts. In COLING. Saarbr¨ucken, Germany.

[8] W. Gatterbauer, P. Bohunsky, M. Herzog, B. Kr¨upl, and B. Pollak. 2007. Towards domain-independent information extraction from web tables. In WWW. Banff, Canada.

[9] Scheidt, M. Marcelo. 2013. Ferramenta para extração de WebTables e criação de scripts SQL. Universidade Federal de Santa Catarina, Brasil.