José Alberto Grossi Júnior

Análise Comparativa de Técnicas de Extração de Metadados em Artigos Científicos

Belo Horizonte/MG, Brasil

Mar 2015, v-0.5.4

José Alberto Grossi Júnior

Análise Comparativa de Técnicas de Extração de Metadados em Artigos Científicos

Dissertação de Mestrado apresentada à coordenação do PPGCI/UFMG com o objetivo de obtenção de título de Mestre em Ciência da Informação.

Universidade Federal de Minas Gerais – UFMG

Escola de Ciência da Informação

Programa de Pós-Graduação em Ciência da Informação

Orientador: Marcello Peixoto Bax

Coorientador: Renato Rocha Souza

Belo Horizonte/MG, Brasil

Mar 2015, v-0.5.4

José Alberto Grossi Júnior
Análise Comparativa de Técnicas de Extração de Metadados em Artigos Científicos/ José Alberto Grossi Júnior. – Belo Horizonte/MG, Brasil, Mar 2015, v-0.5.4-
61 p. : il. (algumas color.) ; 30 cm.

Orientador: Marcello Peixoto Bax
Coorientador: Renato Rocha Souza

Dissertação (Mestrado) – Universidade Federal de Minas Gerais – UFMG
Escola de Ciência da Informação
Programa de Pós-Graduação em Ciência da Informação, Mar 2015, v-0.5.4.

1. Extração de informação 2. Metadados. I. Artigos científicos. II. Universidade Federal de Minas Gerais. III. Escola de Ciência da Informação. IV. Análise Comparativa de Técnicas de Extração de Metadados em Artigos Científicos

CDU 02:141:005.7

José Alberto Grossi Júnior

Análise Comparativa de Técnicas de Extração de Metadados em Artigos Científicos

Dissertação de Mestrado apresentada à coordenação do PPGCI/UFMG com o objetivo de obtenção de título de Mestre em Ciência da Informação.

Trabalho aprovado. Belo Horizonte/MG, Brasil, 25 de março de 2015:

Marcello Peixoto Bax
Orientador

Renato Rocha Souza
Coorientador

Professor
Professor Convidado 1

Professor
Professor Convidado 2

Belo Horizonte/MG, Brasil
Mar 2015, v-0.5.4

Este trabalho é dedicado a todas as pessoas que desejam, de uma forma ou outra, superar seus objetivos pessoais.

Resumo

A necessidade de contribuição entre a comunidade acadêmica é evidente quando da necessidade de leituras específicas de artigos científicos de autores espalhados pelo mundo. Porém, esta contribuição se dá de maneira muito pessoal, com envios manuais de artigos quando da necessidade de certos nichos acadêmicos. A dificuldade apresentada geralmente é a centralização de artigos de maneira livre e compensatória, por meio de extração automática de metadados relevantes para o catálogo destes documentos, de maneira a permitir que qualquer pesquisador, devidamente reconhecido, possa compartilhar e obter estes documentos de maneira eficaz e anônima.

Este trabalho demonstra que as técnicas livres existentes para extração de metadados em artigos científicos não são suficientes para abranger os diversos formatos existentes de apresentação dos conteúdos, uma vez que são baseados em layout pré-definidos, sem possibilidade de expansão ou adaptação de acordo com a necessidade de certos grupos de pesquisa, cujo formato de apresentação deste tipo de documento se dá de maneira diferenciada, ou até mesmo, adaptada para seu universo de pesquisadores.

Palavras-chaves: artigos científicos. extração de metadados. extração de dados em artigos.

Abstract

The need of contribution existent in the academic community is focused based on the sharing of papers from authors around the world, when specific studies are needed. However, this contribution is made in a very basic and personal way, with papers sent by manual interactions from some specific research groups. The main goal is focused on the papers centralization in a free and compensatory format, by automatic relevant metadata extraction to the indexation of these documents, allowing any researcher to share and get these documents in a very effective manner.

This work shows how the existent metadata extraction techniques in scientific papers are not totally perfect to perform the different papers formats to present research works, once they are based on pre-defined layouts, without any change of customization according with some groups needs, because of a different presentation format, or even, adapted to your researchers' worlds.

Palavras-chaves: scientific papers. metadata extraction. data extraction on scientific papers.

Lista de ilustrações

Figura 1 – Processo de Extração de Metadados 13
Figura 2 – Distância representando a separação entre classes na técnica de SVM. . 25
Figura 3 – Exemplo de modelo HMM, onde X são os estados, Y as observações possíveis, A as probabilidades de mudança de estado e B as saídas das probabilidades. 28
Figura 4 – Estados utilizados por (ZHANG, 2001) em seu modelo HMM. 30
Figura 5 – Workflow da extração de metadados usando *cluster* de palavras 32
Figura 6 – CERMINE Extraction Workflow 37
Figura 7 – Processo de Metodologia 44

Lista de tabelas

Tabela 1 – Relação de classes utilizadas e comparação com o padrão Dublin Core. 24
Tabela 2 – Ferramentas selecionadas para testes 44
Tabela 3 – Seleção de artigos científicos para testes (ainda em desenvolvimento) . 45
Tabela 4 – Os metadados e seus pesos atribuídos 47
Tabela 5 – Resultados obtidos em cada metadado e sua precisão 48
Tabela 6 – Descrição de cada variável no Índice de Confiabilidade 48
Tabela 7 – Ferramentas e Ambientes de Testes 50

Lista de abreviaturas e siglas

PDF	Portable Document Format
IEEE	Institute of Electrical and Electronics Engineers
RSL	Revisão Sistemática de Literatura
ACM	Association for Computing Machinery
CAPES	Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
XML	eXtensible Markup Language
SVM	Support Vector Machines
HMM	Hidden Markov Models
CRF	Conditional Random Fields
URL	Uniform Resource Locators
HTML	HyperText Markup Language
DCMI	Dublin Core Metadata Initiative
SVM	Support Vector Machines

Sumário

1 INTRODUÇÃO . . . 12
1.1 Contexto . . . 13
1.2 Problema . . . 15
1.3 Justificativa . . . 15
1.4 Histórico . . . 16
1.5 Objetivos . . . 17
1.5.1 Objetivo Geral . . . 17
1.5.2 Objetivos Específicos . . . 17
1.6 Resultados Esperados . . . 18
1.7 Estrutura . . . 18

2 REFERENCIAL TEÓRICO . . . 20
2.1 Metadados . . . 21
2.1.1 Conceito de Metadado . . . 21
2.1.2 Padrões de Metadados . . . 22
2.1.3 Técnicas de Extração de Metadados . . . 23
2.1.3.1 Support Vector Machines (SVM) . . . 23
2.1.3.2 Hidden Markov Models (HMM) . . . 26
2.1.3.3 Word Clustering . . . 31
2.1.3.4 Conditional Random Fields (CRFs) . . . 33
2.2 Revisão de Estado da Arte na Extração de Metadados . . . 35
2.3 Técnicas de Algoritmos . . . 35
2.4 Ambientes de Extração de Metadados . . . 35
2.5 Trabalhos Comparativos . . . 35
2.6 Ferramentas e Projetos de Destaque . . . 35
2.6.1 Cermine . . . 36
2.6.2 TeamBeam . . . 37
2.6.2.1 Resultados Comparativos . . . 38
2.6.3 Layout-CRFs . . . 39
2.6.4 Mendeley Desktop . . . 39
2.6.5 CiteULike . . . 39
2.6.6 CiteSeer . . . 39
2.6.7 ParsCit . . . 41
2.6.7.1 Resultados Comparativos . . . 42

3 METODOLOGIA . . . 43

3.1 Escolha do Corpus . . . 43
3.2 Desenho do Experimento . . . 43
3.3 Ambiente Tecnológico . . . 43
3.4 Seleção das Técnicas/Ferramentas . . . 43
3.5 Seleção de Artigos . . . 44
3.6 Infraestrutura Computacional . . . 45
3.6.1 Testes In Logo . . . 46
3.7 Quadro Comparativo . . . 46
3.8 Metadados, Pesos e Resultados . . . 46
3.8.1 Índice de Confiabilidade . . . 48

4 ANÁLISE E APRESENTAÇÃO DE RESULTADOS . . . 49
4.1 Ambiente de Testes . . . 49
4.1.1 Servidores de Teste . . . 49

5 DISCUSSÃO / TRABALHOS FUTUROS . . . 51
5.1 Contribuições . . . 51
5.2 Trabalhos Futuros . . . 51
5.3 Considerações Finais . . . 51

Referências . . . 52

APÊNDICES 54

APÊNDICE A – LISTAGEM DE ARTIGOS ANALISADOS PARA REALIZAÇÃO DO EXPERIMENTO . . . 55

ANEXOS 58

ANEXO A – ELEMENTOS DO PADRÃO DUBLIN CORE, VERSÃO 1.1 . . . 59

1 Introdução

Comentário: Este texto inicial foi reformulado, corrigindo conjunções e aumentando a ligação entre as informações, já dando uma deixa para a apresentação do Contexto, que vem a seguir. Pequenas correções ortográficas foram feitas ao longo deste texto introdutório também.

A necessidade de contribuição acontece de forma natural no ser humano. Os desejos em ajudar ao próximo e inclusive contribuir com alguma parte de sua formação é algo que desperta um desejo cada vez mais amplo do ponto de vista social.

Somos seres realizados pela satisfação do outro, e seu sucesso de uma forma ou outra acarreta em nosso sucesso, nossa satisfação pessoal e de certa forma também profissional. Sentimos atraídos por contribuir e por compartilhar conhecimento, sendo ele umas das principais formas de realização como pessoa.

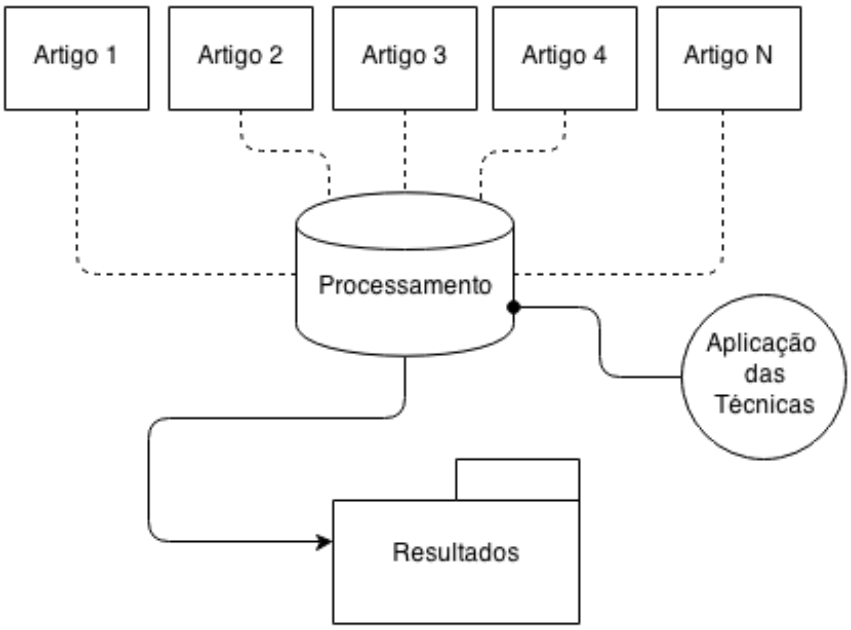
Com o crescimento da pesquisa em todo o mundo um grande número de publicações foram inseridas no meio, fazendo com que uma infinidade de material estivesse disponível em poucos segundos. Deste modo, a necessidade de centralização automatizada dos dados e a contribuição dentre os pesquisadores é inerente ao desenvolvimento desta área.

No âmbito acadêmico sempre contribuímos de uma forma ou outra com a formação de nossos colegas e parceiros de pesquisa. Esta contribuição pode ser feita com base em uma conversa informal ou até mesmo com uma ajuda em documentação ou sugestão de um texto para leitura. Esta sugestão de leitura geralmente possui um caráter muito técnico, e envolve na maioria dos casos a utilização de artigos acadêmicos, objeto central de análise e estudo deste trabalho.

Sabemos da existência de bases de conhecimento nacionais e internacionais, porém, quando estamos falando da contribuição social, em pequena escala, interpessoal, estamos falando de contribuições físicas, com envio de sugestões de artigos para nossos amigos pesquisadores. Este envio é feito de maneira informal, reduz tempo e aumenta consequentemente a praticidade do processo de pesquisa.

Sendo assim, esta experiência como objetivo global seria uma ferramenta poderosa de apoio à pesquisa, com pesquisadores compartilhando conhecimentos de maneira informal, anônima, e segura. Esta forma de disseminação de conhecimento traria um benefício muito grande socialmente falando, uma vez que pesquisadores iriam se unir, mesmo que virtualmente, na transmissão de conhecimento, fazendo do processo da pesquisa um

Figura 1 – Processo de Extração de Metadados



processo mais focado e desta maneira evitando o desperdício de tempo durante a fase de pesquisa em busca por conhecimento.

Para isso, a utilização de técnicas de extração de metadados deve ser utilizada de maneira eficaz, para que, de maneira automática, diversos artigos sejam analisados e catalogados em pequenos universos de pesquisa. Entende-se por metadados os campos básicos e necessários para que uma pesquisa por título de um artigo, por exemplo, seja feita com sucesso. Resume-se então que os metadados que esperam-se ser extraídos destes artigos são: o título do artigo, o nome e e-mail de seus autores, o resumo/*abstract* e as referências utilizadas no texto.

Basicamente estes campos já permitem que uma pesquisa mais detalhada fosse feita e então o artigo localizado. Já as referências são necessárias para se fazer referências inversas de autores que publicam e são citados posteriormente, facilitando ainda mais aos pesquisadores poder, por exemplo, encontrar artigos semelhantes de uma mesma área do conhecimento.

1.1 Contexto

Comentário: Seção nova. Fala um pouco sobre de que maneira a ideia do trabalho surgiu. Peço que avaliem se este tópico não ficou muito pessoal. Tentei escrevê-lo o menos possível em primeira pessoa.

A ideia desta pesquisa surgiu de uma necessidade pessoal durante a disciplina de Fundamentos da Ciência da Informação. Era necessária a leitura de um artigo científico de um determinado autor, porém, este era muito difícil de se encontrar disponível na Internet, o que fez com que tivesse de recorrer ao professor da disciplina para consegui-lo.

Após o envio deste artigo pelo professor, conforme solicitado, surgiu a seguinte reflexão:

> *"Esta contribuição ocorreu de maneira pessoal, entre mim e o professor, porém, se existisse um local onde os pesquisadores pudessem contribuir com o envio destes artigos este processo seria muito mais rápido e eficaz. Seria como se pudéssemos encontrar o artigo instantaneamente, dentro de nossa necessidade temporal. Seria pesquisador ajudando pesquisador."*

Desta forma, inclusive por minha formação acadêmica na área de Ciência da Computação, e por minha experiência profissional de mais de uma década em desenvolvimento Web, surgiu a ideia de desenvolver uma ferramenta de contribuição entre pesquisadores, totalmente online e independente. Todavia, estes repositórios de artigos já existiam na Internet e eram bem populados por sinal, porém, todos com um certo caráter comercial, com uma empresa ou instituição por trás, objetivando venda e lucro de uma maneira ou outra.

A ideia do desenvolvimento desta ferramenta ecoou durante dias, visto sua ampla aplicabilidade, como também sua contribuição social para com o universo da pesquisa, facilitando a vida de pesquisadores ao redor do mundo, sem complicações. Porém, para sua eficácia no armazenamento dos artigos enviados era necessária uma análise automatizada dos conteúdos destes arquivos, de maneira a extrair as partes mais importantes, como título, autores e referências, para que pudessem ser catalogados de maneira organizada e estruturada, a fim de facilitar a busca e permitir uma performance satisfatória por parte do usuário. Estes dados retirados dos documentos analisados são referenciados neste projeto pelo nome de "metadados", e são detalhados mais a frente de maneira aprofundada.

Sendo assim iniciou-se minha pesquisa pelo campo da extração de informação e principalmente pelo *machine learning*, levando ao objeto central deste projeto, de maneira a identificar as características de cada ferramenta disponível atualmente e suas melhores aplicações dentro deste universo. Nada mais justo que contribuir com um projeto de pesquisa tendo como base fundamental o resultado de uma pesquisa acadêmica.

1.2 Problema

Comentário: O problema foi reformulado, visando a identificação com o objetivo e resultados esperados.

Diversas ferramentas e técnicas para extração de metadados em artigos podem ser facilmente encontradas com uma rápida pesquisa pela Internet. Porém, algumas são propriedade de universidades ou até mesmo de instituições privadas, o que dificulta sua análise e teste, visto que seu código fonte é fechado, ou seja, sem acesso.

As técnicas abordadas por este trabalho são aquelas cujo código fonte é aberto, ou seja, que pode ser analisado, testado e alterado, o que chamamos de projetos *open source* (código livre).

De modo geral, estas técnicas livres de extração são focadas em *layouts* (disposição visual) pré-definidos, geralmente seguindo modelos de conferências e/ou congressos internacionais, que possuem um padrão visual parecido, como é o caso do IEEE por exemplo, que serve de referência para diversos outros eventos da área da computação, tomando seu *layout*, a princípio, como base.

Porém, existem diversos outros eventos que possuem *layouts* de artigos considerados fora do padrão e, portanto, necessitam de adaptações destas técnicas para que seus trabalhos possam ser analisados e catalogados de maneira eficaz. Esta customização promoveria uma série de tentativas para verificar o melhor *layout* para ser utilizado em cada caso, automaticamente.

Algumas técnicas/ferramentas são aparentemente muito eficazes para um certo grupo de artigos, já seguindo um padrão visual pré-determinado. Porém, para alguns *layouts* pouco comuns, de áreas do conhecimento diversas, espera-se que estas ferramentas não sejam tão eficazes, o que varia de acordo com a tecnologia aplicada e principalmente de acordo com o princípio teórico utilizado pelos seus autores, que será mais aprofundado nos próximos capítulos.

1.3 Justificativa

Comentário: Justificativa foi reformulada.

Interessante notar a necessidade de centralização de artigos científicos para o meio acadêmico, além de permitir à contribuição coletiva, que seria uma forma de aumentar cada vez mais o acesso aos materiais de pesquisa.

Diante disso, esta extração de metadados automatizada e eficaz traria benefícios para que estes repositórios de artigos fossem criados e catalogados com precisão, tendo então milhões e milhões de documentos em suas bases de dados, prontos para serem pesquisados por pesquisadores de todas as nacionalidades e áreas do conhecimento.

Este trabalho busca, portanto, identificar a melhor maneira que isso pode ser feito, levando em consideração o melhor resultado matemático possível dentro das técnicas e ferramentes disponíveis atualmente. Este resultado traria, para tanto, um ganho científico incalculável, permitindo que novas ferramentas e repositórios fossem criados e a disseminação do conhecimento fosse cada vez maior, e mais rápida.

1.4 Histórico

Comentário: Nova seção. Resolvi colocá-la, porém de maneira breve, para que sirva de uma introdução sobre o assunto e principalmente demonstrar a ligação da Ciência da Informação com a Ciência da Computação.

Os temas ”extração de informação” e ”machine learning” são resultantes da fusão de duas áreas do conhecimento bem complementares: a Ciência da Informação e a Ciência da Computação. Por esta proximidade, é muito comum encontrar artigos destes temas em ambas as áreas, porém com focos e objetivos diferentes.

Nesse sentido, essa união se complementa de maneira muito eficaz, visto a característica da fundamentação teórica da Ciência da Informação com a praticidade e desenvolvimento da Ciência da Computação. O resultado é um processo amplo e profundo, que permite que cada vez mais pesquisas sejam feitas e o ganho científico seja cada vez maior.

As pesquisas de *machine learning* se iniciaram muito antes de se tornarem viáveis do ponto de vista tecnológico. Os primeiros registros desta pesquisa são datados de 1956 por (SOLOMONOFF, 1956), onde as primeiras teorias de aprendizado automatizado foram formuladas, utilizando-se para tal de conceitos matemáticos, defendendo a ideia do aprendizado por repetição e utilização de padrões.

Esta fundamentação ainda é muito utilizada e é a base para o desenvolvimento tecnológico existente na área de ”Inteligência Artificial”. Apesar de ser amplamente aplicada, os conceitos por trás desta fundamentação evoluíram ao longo das décadas, juntamente com a capacidade computacional, tornando os resultados muito mais reais e precisos do que era possível imaginar na formulação desta ideia.

Diante do cenário atual, com base no acelerado ritmo de desenvolvimento da tecnologia, bem como da evolução das técnicas de *machine learning*, a utilização das ideias apresentadas décadas atrás ainda estão presentes e são amplamente utilizadas para o desenvolvimento de novas tecnologias e ferramentas.

Como é definido por (MOHRI; ROSTAMIZADEH; TALWALKAR, 2012), *machine learning* atua como uma forma de aprendizado com base em experiências passadas, através da utilização de dados eletrônicos coletados que analisados posteriormente seguindo padrões definidos à máquina, de modo a permitir que ela possa fazer previsões com exatidão com base em sua experiência passada.

Este tema é muito amplo e sua aplicabilidade é extremamente ampla, permitindo a utilização de suas técnicas e teorias em diversas atividades, como classificação de textos e documentos, processamento de linguagem natural, reconhecimento de fala, detecção de fraudes, diagnósticos médicos e sistemas de recomendações, assim como mecanismos de buscas e extração de informação, ponto principal de discussão deste trabalho.

Dessa forma, este assunto tem sido discutido e pesquisado muito amplamente na Ciência da Informação, tanto nas áreas de classificação de documentos até mesmo em aplicações de buscas, unindo os conhecimentos adquiridos desta área do conhecimento com a evolução computacional presente nos dias atuais, onde a Ciência da Computação ganha força e nos permite realizar experimentos mais eficazes.

1.5 Objetivos

Comentário: Os objetivos foram parcialmente reescritos, visando uma ligação maior com a metodologia, conforme observações feitas.

1.5.1 Objetivo Geral

Este trabalho possui como objetivo geral identificar quais são as melhores técnicas de extração de metadados em artigos científicos e suas melhores aplicações, com base em um conjunto de documentos pré-selecionados para testes, dos mais diversos padrões e de diversas áreas do conhecimento.

Com efeito, esta identificação permite que resultados sejam comparados e confrontados, permitindo decidir qual técnica é melhor utilizada para cada padrão visual, abrangendo um conjunto cada vez maior de dados e tendo resultados cada vez mais precisos.

1.5.2 Objetivos Específicos

Com base na diferenciação dos *layouts* de artigos científicos, este trabalho tem como objetivos específicos identificar também pontos em que as técnicas de extração de metadados necessitam de adaptações por parte de seus usuários e desenvolvedores, permitindo desta forma uma ampliação do universo de aplicação e garantindo assim uma cobertura mais abrangente dos artigos científicos analisados.

Acredita-se que os padrões de extração existentes no mercado são, de maneira geral, insuficientes para suprir todos os padrões de artigos existente, limitando a apenas uma pequena parcela destes, dentro de um *layout* específico, o que acaba gerando um desconforto e uma perda de credibilidade destas técnicas existentes atualmente.

1.6 Resultados Esperados

Comentário: Texto parcialmente alterado para prover uma maior identificação com o objetivo e o problema da pesquisa, de maneira a criar uma ligação entre eles.

As formas de extração de metadados em artigos científicos são geralmente baseadas em *layouts*, ou seja, em pequenos pedaços onde certos dados devem ser extraídos. Porém em virtude da grande diversidade de materiais produzidos, dos mais diversos padrões visuais existentes, este *layout* padrão - geralmente utilizado em artigos científicos - não se mostra eficiente na abrangência total das necessidades da comunidade científica mundial.

Sobre este aspecto, espera-se que certos artigos científicos não tenham seus metadados analisados de maneira eficaz por todas as técnicas de extração analisadas, uma vez que adaptações no código seriam necessárias a fim de permitir que outros padrões visuais de artigos fossem também analisados, aplicando então um dos fundamentos do *machine learning*, o aprendizado.

Por fim, espera-se que, com esta análise aprofundada das técnicas existentes, as mais eficazes na extração de metadados sejam identificadas e analisadas, elevando então o ganho científico neste tema de suma importância para o desenvolvimento tecnológico.

1.7 Estrutura

Esta pesquisa é estruturada iniciando com uma breve introdução sobre o tema; a apresentação do contexto da pesquisa, bem como sua origem e motivação; a definição do problema; a justificativa; o histórico do trabalho, demonstrando suas origens e fundamentos históricos; os objetivos gerais e específicos e, por fim, os resultados esperados.

O segundo capítulo tem como base o referencial teórico feito através de uma RSL (Revisão Sistemática de Literatura), tendo como base (KITCHENHAM, 2004), que propõe

um passo-a-passo para um revisão de literatura eficaz de modo a atingir os resultados desejáveis com a pesquisa.

No terceiro capítulo temos a metodologia para o desenvolvimento do trabalho, as técnicas que serão aplicadas e principalmente como serão feitas. Posteriormente, no capítulo quarto temos a análise e apresentação dos resultados, explicando como os testes foram realizados, os ambientes de teste criados e os resultados coletados.

No quinto capítulo temos a discussão/conclusão, trabalhos futuros e considerações finais sobre o trabalho apresentado.

2 Referencial Teórico

Visando obter uma revisão de literatura eficaz foi feita uma Revisão Sistemática de Literatura, com base no relatório técnico de (KITCHENHAM, 2004). Diante disso, o projeto se torna mais abrangente do ponto de vista de pesquisa literária e ao mesmo tempo mais restritivo, realmente realizando o estudo dos objetos que são importantes para a pesquisa e seus resultados.

Como o objetivo do trabalho tem foco na resultado prático, geralmente as bases relacionadas às áreas mais técnicas, como Ciência da Computação, devem ser bem focadas, como IEEE e ACM, por exemplo. Outras bases de natureza mais genérica também foram pesquisadas, geralmente através do site da CAPES, mas com caráter mais de apoio teórico realmente.

Embora a amplitude desta área de estudo, alguns detalhes necessitam ser observados para evitar assim redundância e perda de foco. A pesquisa necessitava ser focada em técnicas de extração de metadados em artigos científicos, somente. Estas técnicas necessitam ser reais, de maneira a existir realmente uma forma prática de serem testadas, fazendo assim com que os resultados obtidos sejam comparados e confrontados para verificar então a eficácia deste processo.

Como a pesquisa necessita de um resultado prático eficaz os critérios que serão adotados para demonstrar a relevância de um estudo serão seus resultados. Com base em uma técnica potencialmente eficaz, esta deve ser testada juntamente com um grupo de artigos previamente selecionados. Estes artigos já possuirão seus dados mapeados de maneira a poder comparar os resultados obtidos com cada uma das técnicas e seus respectivos resultados. Portanto os critérios utilizados serão de natureza explicitamente prática, com foco em resultados concretos, numericamente representados.

Seguindo as orientações propostas em (KITCHENHAM, 2004) foram mapeados alguns eventos para serem pesquisados, a fim de encontrar trabalhos relevantes para a área. Sendo assim, foram mapeados os seguintes eventos:

- KDIR (International Conference of Knowledge Discovery and Information Retrieval)
- ICDAR (International Conference on Document Analysis and Recognition)
- PDCAT (International Conference on Parallel and Distributed Computing, Applications and Technologies)
- IAPR (International Workshop on Document Analysis Systems)

- ACM Conference on Digital Libraries

Estes eventos foram analisados, observando todas as citações e referências utilizadas, a fim de encontrar também detalhes importantes para a pesquisa, sem perder o foco do trabalho.

2.1 Metadados

Comentário: Esta nova seção foi incluída para definir alguns conceitos que são apresentados mais a frente com a apresentação das técnicas.

2.1.1 Conceito de Metadado

Com base nas ideias apresentadas por (CATHRO, 1997), podemos definir inicialmente um metadado da seguinte maneira:

> *"[...] an element of metadata describes an information resource, or helps provide access to an information resource."*

Sobre este aspecto conclui-se que todo dado que agregue nova informação a um recurso pode ser considerado um metadado. Desta maneira, quanto mais metadados um recurso tiver, mais detalhado ele é, ou seja, mais dados sobre ele temos. Com efeito, podemos simplificar ainda mais a definição de metadado com sendo *um conjunto de dados sobre um determinado recurso.*

Temos contato diariamente com diversos metadados, sejam eles visíveis naturalmente ou não, mas de uma maneira ou outra, são utilizados em tarefas do cotidiano de maneira ampla; geralmente um conjunto enorme de novos dados e informações sobre recursos dos mais diversos tipos e modelos.

Podemos citar, por exemplo, a utilização de pequenos pedaços de dados sobre um conjunto de livros, dentro de um ambiente de biblioteca, por exemplo, o que é considerado uma coleção de elementos de metadados, conforme pode ser confirmado em (CATHRO, 1997). Além disso, o mesmo autor cita como exemplos de metadados os dados coletados por mecanismos de busca no momento em que as páginas da Internet são indexadas e então armazenadas.

Ante o exposto, podemos criar e identificar diversos conceitos do que podemos chamar de metadado, porém, em virtude de sua enorme aplicação e significância, alguns detalhes dentro do escopo deste trabalho devem ser direcionados, visando identificar e

explicar melhor os conceitos que serão aplicados na análise das ferramentas e técnicas que aqui serão apresentadas.

2.1.2 Padrões de Metadados

Diante da infinidade de dados que podem estar atrelados a um recurso, temos uma amplitude muito grande de características que podem ser definidas como sendo metadado. Por isso, foram definidos 15 (quinze) elementos de metadados para descreverem um recurso informacional, independente da disciplina em questão, estabelecendo então um padrão.

Para isso, foi criado o padrão Dublin Core (WEIBEL, 1997), que se originou após uma série de encontros feitos desde 1995, unindo bibliotecários e pesquisadores digitais e de conteúdo, visando identificar padrões para se representar um recurso eletrônico, seja ele uma página, um texto ou até mesmo um documento em um determinado formato. O nome ”Dublin” foi dado em virtude da primeira reunião do grupo, que foi realizada em Dublin, Ohio. Já o nome ”core” se deu em virtude dos elementos serem amplos e genéricos, sendo então utilizados para descrever uma grande variedade de recursos.

Os quinze elementos que fazem parte do padrão Dublin Core compartilham de um vasto conjunto de vocabulários de metadados, juntamente com diversas especificações técnicas, que são mantidas pela Dublin Core Metadata Initiative (DCMI), a agência responsável pela definição destes elementos.

Este padrão é utilizado nos dias atuais para se representar um recurso na Internet, por exemplo, podendo ser qualquer coisa que possa ser identificada (KUNZE; BAKER, 2007). Com base em suas características, mecanismos de busca podem indexar um recurso de maneira mais rápida e precisa, pois este é acompanhado de pequenas sinalizações sobre seu conteúdo, que são os metadados.

Para cada elemento descrito pelo padrão temos informações como:

- *label*, que é o texto para leitura e entendimento humano;
- *name*, que é usado para o processamento de máquina, ou seja, um identificador que a máquina utiliza para reconhecimento.

Os elementos que fazem parte da versão 1.1 do padrão Dublin Core (KUNZE; BAKER, 2007) podem ser vistos no Anexo A deste trabalho.

Abaixo podemos ver um exemplo de código para utilização de metadados Dublin Core em uma página da Internet, por exemplo, onde iremos referenciar os elementos *creator*, *title* e *language*.

```
<meta name="DC.Creator" content="Junior␣Grossi" >
```

```
<meta name="DC.Title" content="Análise Comparativa de Técnicas
   de Extração de Metadados em Artigos Científicos" >
<meta name="DC.Language" content="pt_BR" >
```

Desta forma, a indexação do autor (*DC.Creator*) da página, bem como seu título (*DC.Title*) e idioma (*DC.Language*), são feitos de maneira bem simplificada e direta. Para páginas da Internet, com as informações todas em formato HTML, a utilização destes metadados perderiam o sentido, visto que essas informações poderiam ser facilmente encontradas de outras formas, como a análise do próprio código HTML. Porém, para documentos binários - em formato PDF ou Word, por exemplo - a utilização destes metadados é de suma importância, visto que permite que essas informações básicas sejam capturadas sem a necessidade de análise do conteúdo dos arquivos, o que traria complexidade ao processo.

2.1.3 Técnicas de Extração de Metadados

Comentário: Lembrando que todo texto em ROXO não estava presente na última versão enviada e todo texto em VERMELHO foi alterado deste a última versão.

Algumas técnicas e algoritmos de extração de metadados são utilizadas em diversos projetos, de maneira a serem citadas em momentos onde exige-se uma precisão maior.

Estas técnicas se baseiam basicamente na classificação de dados com base nas suas representações escritas, tanto baseadas em padrões preestabelecidos ou até mesmo com base em um dicionário de palavras capaz de reconhecer ocorrências em diversas partes de um documento, o que agrega assertividade ao processo de extração.

2.1.3.1 Support Vector Machines (SVM)

Support Vector Machines (SVM) é uma técnica de *machine learning* que permite que um conjunto de dados sejam analisados, possibilitando que padrões sejam extraídos, criando uma espécie de memória (VAPNIK; CORTES, 1995). Seu objetivo inicial era ser uma técnica de classificação de dados, por ser focada em reconhecimento de padrão através de análises matemáticas.

Esta técnica se baseia na redução de erros com base em um resultado de treinos consecutivos que permitem a criação de um padrão e estabelecem um aprendizado por parte da distância entre ocorrências (VAPNIK; CORTES, 1995). Todas as análises realizadas são mapeadas, permitindo que um registro histórico em forma matemática seja realizado, levando o algoritmo à possibilidade de diferenciação entre um resultado e outro, de forma numérica.

Tabela 1 – Relação de classes utilizadas e comparação com o padrão Dublin Core.

Classe (Tag)	**Referência Dublin Core**	**Descrição**
Title	Title	Título do artigo
Author	Creator	Nome do autor do documento
Affiliation		Afiliação do autor
Address		Endereço do autor
Note		Frases de reconhecimentos, *copyright*
Email		Endereço de e-mail do autor
Date		Data da publicação
Abstract Introdution	Description	A parte de introdução do artigo
Phone		Telefone do autor
Keyword	Subject	As palavras-chave do documento
Web		Endereço na Internet do autor
Degree		Associação com o grau acadêmico
Pubnum		Número da publicação do documento
Page		O final da página

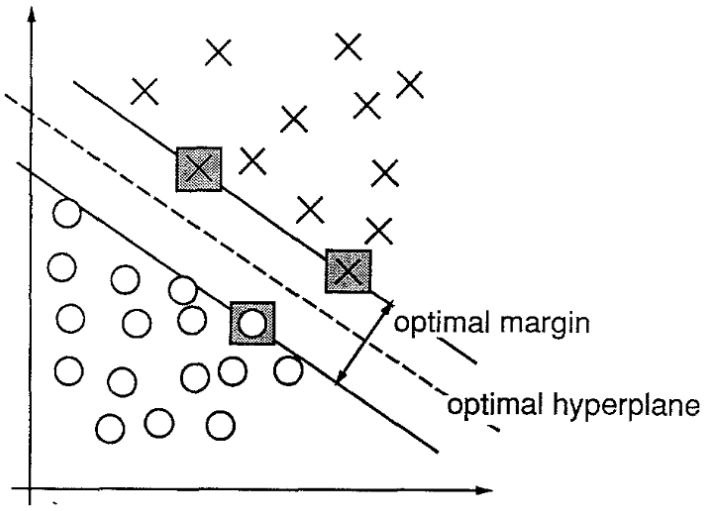
De acordo com Chieu (CHIEU; NG, 2002), ele sugere que a tarefa de extrair informação pode ser considerada um problema de classificação. Partindo deste pensamento foi que Han (HAN et al., 2003) decidiu utilizar técnicas de SVM para extração de metadados, utilizando das qualidades matemáticas do processo no reconhecimento de padrões, o que permitiu que novas descobertas fossem feitas, expandindo o estudo da extração de dados para um patamar mais amplo e elevado.

Várias ferramentas de extração se baseiam na utilização de SVM como técnica principal, visto sua eficiência no reconhecimento de padrões. Como descrito por (HAN et al., 2003) a utilização desta técnica é baseada na identificação de campos previamente selecionados no cabeçalho de um documento, do qual se deseja obter os metadados, por exemplo.

Esta técnica analisa diversos campos chamando-os de classes, e atribui a cada classe uma característica que a permite ser identificada. Deste modo, cada linha do cabeçalho do documento é classificada em uma ou mais classes, onde algumas fazem parte do padrão Dublin Core, detalhado na subseção 2.1.2.

Seymore et al. (SEYMORE; ROSENFIELD, 1999) definiu 15 (quinze) *tags* para esta definição do cabeçalho de um documento. Porém, destas 15 (quinze) *tags* definidas, somente 4 (quatro) correspondem ao padrão da Dublin Core e estão ilustradas na Tabela 1. Estas também foram as *tags* utilizadas por Han para a extração de metadados dos cabeçalhos de artigos científicos (HAN et al., 2003).

Figura 2 – Distância representando a separação entre classes na técnica de SVM.



A ocorrência dos campos é mapeada em uma representação bidimensional, permitindo identificar visualmente os padrões encontrados na análise de cada classe. Desta forma, com a visualização do posicionamento de cada ocorrência é possível obter uma distância clara entre os pontos de reconhecimento, chamada pelo autor de *hyperplanes* (VAPNIK; CORTES, 1995). Por sua vez, estes pontos são marcados como sendo os "support vectors", que permitem que o *hyperplane* entre eles determine a divisão entre as classes de forma clara e eficaz, como pode ser visto na Tabela 2.1.3.1. Esta divisão permite então a distinção entre os metadados, diferenciando os elementos analisados pelo algoritmo.

Além desta análise são utilizadas comparações de palavras dentro de um contexto previamente selecionado. Assim foi criado um *cluster* de palavras comuns que facilita na identificação destas classes nos cabeçalhos analisados. Este *cluster* basicamente é composto de:

- Dicionário online padrão em sistemas Linux;
- 8441 nomes e 19613 sobrenomes;
- Sobrenomes chineses;
- Nome dos estados do Estados Unidos e das províncias canadenses;
- Nomes das cidades dos Estados Unidos;
- Nome dos países do mundo, de acordo com World Fact Book[1];
- Nome dos meses e suas respectivas abreviações.

[1] Disponível em <https://www.cia.gov/library/publications/the-world-factbook/index.html>

Para cada uma das classes analisadas são feitas correlações com o tipo de dado esperado, de maneira a permitir que endereços de e-mail, por exemplo, sejam extraídos com base em expressões regulares utilizadas em linguagens de programação.

Support Vector Machine é uma técnica conhecida principalmente por sua boa performance e habilidade para com grandes quantidades de dados, sendo por isso considerada uma boa solução para problemas de classificação. Por essa característica sua principal funcionalidade é baseada em opções, e a que mais se assemelha com a análise realizada é então classificada. Por isso Han (HAN et al., 2003) decidiu utilizar este mesmo conceito de comparação para ser então aplicado na extração de dados, de modo a confrontar e comparar classes de metadados, a fim de identificá-los e diferenciá-los.

Han também encontrou alguns desafios na diferenciação de campos, como é o caso dos múltiplos autores. Em alguns casos a diferenciação de autores, que fazem do mesmo campo, poderiam estar até em linhas ou grupos diferentes. Para isso foram utilizados alguns elementos para representar a separação dos nomes (*chunks*), como pontuações e a palavra ”and”. Desta forma, os autores eram extraídos seguindo este padrão estipulado.

O resultado obtido pela utilização de SVM como técnica de extração de metadados foi bem relevante. O autor realizou uma comparação com a aplicação da técnica de Hidden Markov Models (HMM) - detalhada na subseção 2.1.3.2 - onde a SVM se mostrou mais eficaz na precisão da extração para algumas classes específicas, como é o caso dos títulos, autores e endereços, por exemplo. Para outras classes a utilização da técnica de HMM ainda demonstrou ser mais eficiente na extração de metadados.

2.1.3.2 Hidden Markov Models (HMM)

A teoria básica de Markov foi conhecida próximo dos anos 80 por engenheiros e matemáticos com grande aplicação inicialmente em processamento da fala, mas com vasta amplitude em outras áreas onde a descoberta de padrões pode ser aplicada (RABINER; JUANG, 1986).

O processo é baseado na identificação de modelos observáveis que representem e caracterizem a ocorrências de símbolos, ou seja, padrões, em determinados estados. Se um sinal foi observado ele pode ser utilizado para futuras referências, de acordo com o padrão.

Um exemplo prático citado por Rabiner e Juang (RABINER; JUANG, 1986) é o caso de uso do jogo *Cara e Coroa*. Toma-se um observador em um quarto fechado com uma cortina totalmente fechada para outro cômodo. Este observador não consegue ver nada que acontece no outro cômodo, onde está uma outra pessoa jogando uma moeda pra cima, relatando sempre o resultado obtido (cara ou coroa). Neste caso o problema é construir um modelo Hidden Markov Model (HMM) para explicar ao observador a sequência dos resultados obtidos.

Neste exemplo, o primeiro caso é baseado tanto no estado de cada resultado (cara ou coroa) e em probabilidades matemáticas de ocorrência destes estados, neste caso, 0.5, ou seja, dois estados. Assim desenha-se modelos onde os estados são representados com base nas inúmeras possibilidades existentes, levando inclusive em consideração a sequência dos últimos acontecimentos. Outra possibilidade seria a existência de duas moedas, o que daria ainda dois estados existentes, mas não em função da probabilidade de sair cara ou coroa, mas sim por serem consideradas duas moedas "justas", o que daria também uma probabilidade de 0.5 pra cada.

Neste último exemplo o grande detalhe do modelo é que este é oculto (*hidden*). Isso se deve ao fato de os dois estados, representados pelas duas moedas, serem totalmente independentes, o que não permite identificar qual moeda é a "justa" e então informar ao observador o resultado daquela rodada.

Por esta alteração de resultados e probabilidades, o fator decisivo na criação de cada modelo é a definição do número de estados que ele terá. Além disso, outro ponto que determina o sucesso do método é a utilização de um resultado anterior - *training data set* -, ou seja, uma memória, um conjunto de informações pré-identificadas que permitiriam ainda à associação dos estados e ocorrência dos símbolos (RABINER; JUANG, 1986).

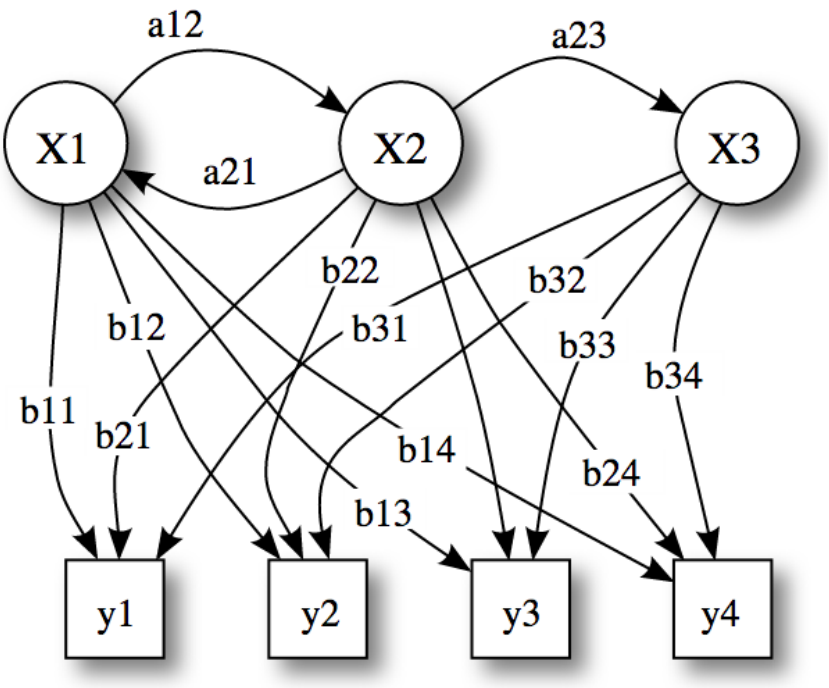
Um HMM pode ser formado por um conjunto de elementos, que formarão toda a teoria e aplicação dos algoritmos dentro do processo:

1. Um número `N` de estados, onde `N` é um inteiro finito;
2. Um intervalo temporal `t`, que determina a entrada em um novo estado, através de uma transição de probabilidade entre eles, levando em consideração sempre o estado anterior;
3. Após cada transição o observador registra um `símbolo` de acordo com a distribuição de probabilidade, que por sua vez depende do estado atual do modelo.

A utilização dos resultados passados - *training data set* - é muito importante para uma boa definição de um HMM, visto que permite adaptar os parâmetros do modelo para aquele conjunto de dados passados, que por sua vez fazem parte de um padrão já identificado.

Seguindo este padrão o HMM pode ser utilizado, por exemplo, para reconhecimento de palavras isoladas que, juntamente com a utilização de um vocabulário de palavras permite a criação de modelos de reconhecimento. Cada palavra deste vocabulário seria um modelo HMM, permitindo que a palavra escolhida fosse a pertencente ao modelo com maior probabilidade.

Figura 3 – Exemplo de modelo HMM, onde X são os estados, Y as observações possíveis, A as probabilidades de mudança de estado e B as saídas das probabilidades.



Já no âmbito da extração da informação, o HMM pode ser aplicado conforme é apresentado por Seymore et al. (SEYMORE; ROSENFIELD, 1999), onde um modelo construído manualmente contendo múltiplos estados por campos (título, autor, etc), pode ser mais eficiente do que um modelo com um estado por campo.

Um dos pontos positivos deste modelo é que por serem baseados em estatística eles são muito bem empregados em problemas de linguagem natural, aliando os resultados positivos à excelente performance computacional. Como desvantagem deste método podemos citar o fato de, por ser baseado em estatística, uma grande quantidade prévia de dados deve ser utilizada a título de treino para se obter padrões significativos para então ser aplicados de maneira final na criação do modelo.

Deste modo, para extração dos metadados, o HMM pode ser utilizado aplicando um marcador (*label*) em cada palavra do cabeçalho de um documento (artigo científico), relacionando cada palavra a uma classe, como título, autor, etc. Assim, uma opção seria a criação de um modelo com `N` estados, onde cada estado corresponderia a uma classe que se deseja extrair - por exemplo o título do documento. Porém, no caso da existência de sequências ocultas (*hidden sequences*) - o que alteraria a seleção do estado seguinte - a utilização de vários estados para cada classe traria resultados melhores (SEYMORE; ROSENFIELD, 1999).

Para isso seria necessário entender melhor a estrutura do modelo de acordo com dados de treino (*training data*), utilizando-se então de múltiplos estados para cada classe,

o que traria resultados melhores. Deste modo, este *training data* permitiria a construção de um modelo chamado *maximally-specific model*. Neste modelo, cada palavra do *training data* seria associada com seu próprio estado, com transição para o estado correspondente à palavra seguinte (SEYMORE; ROSENFIELD, 1999).

Este modelo - segundo os autores - poderia ser usado como o ponto de partida de uma variedade de técnicas para junção de estados (*state merging techniques*). Desta forma os autores propõe duas formas de junção de estados que permita a construção deste *maximally-specific model*:

1. ***Neighbor-merging:*** combina todos os estados que compartilham transições e possuem o mesmo nome de classe. Por exemplo, uma sequência de estados correspondentes ao título seriam juntados em apenas um estado. Assim, vários estados vizinhos com os mesmos nomes de classes seriam juntados em apenas um.

2. ***V-merging:*** combina quaisquer dois estados que possuem o mesmo nome de classe e compartilham transições "de" ou "para" um estado comum. Desde modo, ao invés de começar no estado inicial e decidir para qual estado correspondente ao título será feita a transição, esta técnica juntaria os estados "filhos" em um único estado, de maneira que somente uma transição do estado inicial para o estado de título iria existir. Desta forma, esta técnica poderia ser utilizada como modelo diretamente para extração de dados, podendo também criar novas combinações de estados, implicando em uma futura melhora deste modelo.

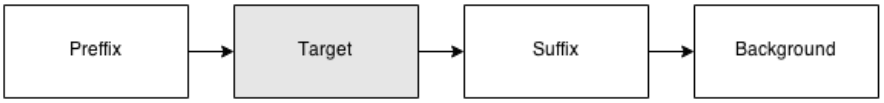
Como o objetivo de Seymore et al. é extrair informações relevantes de cabeçalho de artigos de Ciência da Computação, a área de análise nestes documentos se limita até o início da introdução, ou ao final da primeira página, o que ocorrer primeiro.

O resumo (*abstract*) é extraído facilmente com a utilização de expressão regular. Algumas classes de palavras especiais são identificadas também através de expressão regular e transformadas em *tags* ou *tokens*, como `<EMAIL>` ou `<YEAR_NUMBER>`, por exemplo. Em todos estes casos, todos os acentos e informações de novas linhas (
`n`) são removidos do texto.

Os resultados apontam como muito positiva a utilização de HMM para a extração de dados em cabeçalhos de artigos científicos. A precisão encontrada no experimento foi de 92,9% para todas as classes do cabeçalho e, mais especificamente, 97,2% para a extração dos autores. Também como resultado pode-se afirmar que modelos HMM com mais de um estado por classe são mais eficientes do que os que utilizam apenas um estado para cada classe analisada (SEYMORE; ROSENFIELD, 1999).

Uma outra utilização de HMM na extração de informação é descrita por (ZHANG, 2001), onde é realizado um experimento de extração de informação utilizando um conjunto

Figura 4 – Estados utilizados por (ZHANG, 2001) em seu modelo HMM.



de dados semi-estruturados, em formato HTML, contendo informações sobre restaurantes da cidade de Los Angeles.

Neste caso são estipulados quatro estados para o modelo: *Background*, *Prefix*, *Suffix* e *Target*. O *Target* é o estado responsável pela emissão do símbolo - chamado pelo autor de *token* - para o "campo-alvo". O *Preffix* e *Suffix* são estados que emitem símbolos que aparecem respectivamente antes e depois deste campo-alvo. Todos os demais símbolos são emitidos no estado *Background*. A relação entre os estados pode ser visualizada na Figura 4.

Os seguintes campos deveriam ser extraídos das informações dos restaurantes: *restaurant name* (nome do restaurante), *telephone number* (número de telefone), *hours (horas)* e *cuisine* (tipo de comida servida, como italiana, alemã, etc). Segundo Zhang (ZHANG, 2001) os campos *restaurant name* e *telephone number* deveriam ser mais fáceis de serem obtidos, visto que o nome do restaurante geralmente se encontra em destaque, com alguma diferenciação visual. Já o telefone possui um formato numérico, que permite mais facilmente uma identificação de um padrão. O campo *hours* também não foi complicado, embora se tenha uma variedade muito grande na disposição desta informação. Já o campo *cousine* foi mais difícil de ser extraído, visto a diversidade que existe na forma de um restaurante especificar e/ou representar sua cozinha, tanto na utilização de palavras diferentes quanto na própria identificação do estilo do restaurante por si só.

Como resultados esperados, os campos *restaurant name* e *telephone number* obtiveram muito êxito em sua extração, com resultados realmente consideráveis. Já os campos *hours* e *cousine* não tiveram resultados muito satisfatórios, o que pode ser explicado em função da característica dos HMMs de utilizar como modelo resultado de aprendizados anteriores, os *training data set* ou simplesmente *traning data*. Como nestes dois campos há uma grande possibilidade de representação diferente, os resultados não foram tão eficazes, o que poderia ser resolvido com uma alteração no modelo HMM que permitisse que as palavras identificadas como sendo do campo *cousine* fossem capturadas de maneira mais proveitosa, que estão, neste modelo sugerido, isoladas no estado *Background*.

Em função dos resultados obtidos, pode-se considerar a performance de HMM na extração de informação muito positiva, visto as possibilidades de variação do modelo, que permite um resultado mais acurado e próximo dos objetivos reais (ZHANG, 2001). Além disso, a utilização de resultados passados garante um aprendizado muito importante

para que o modelo seja estabelecido, o que garante ainda mais um ganho de eficiência na aplicação desta técnica.

Além da utilização de HMM de forma natural, algumas variações de seu algoritmo são também citadas na literatura, como é o caso dos MEMMs (*Maximum Entropy Markov Models*) (BERGER; PIETRA; PIETRA, 1996). Nos MEMMs cada estado possui um modelo exponencial que utiliza as características de observação como entrada de dados (*input*) (LAFFERTY; MCCALLUM; PEREIRA, 2001). Estes modelos são baseados na técnica de HMM, se diferenciando na maneira como os estados se relacionam, bem como a relação entre suas transições, o que leva à citações de maneira independente por alguns autores, porém, com herança lógica dos conceitos de HMM.

2.1.3.3 Word Clustering

Técnicas de classificação de texto geralmente utilizam palavras extraídas do texto como a principal fonte de recursos para a representação. Por outro lado, os *clusters* de palavras tem sido uma proposta eficaz para a redução da dimensionalidade e da dispersão, melhorando assim a performance desta classificação (HAN et al., 2005).

Basicamente o conceito de *clusters* compreende um conjunto de palavras formando um banco de dados de domínio (*domain database*), aliado com um conjunto de propriedades ortográficas de demais palavras dentro daquele contexto específico. A utilização destes *clusters* juntamente com outras técnicas tem mostrado um ganho de 6.6% na performance de classificação de elementos de um cabeçalho de artigo científico, e ainda 8,4% de ganho de performance para a extração da bibliografia destes documentos (HAN et al., 2005).

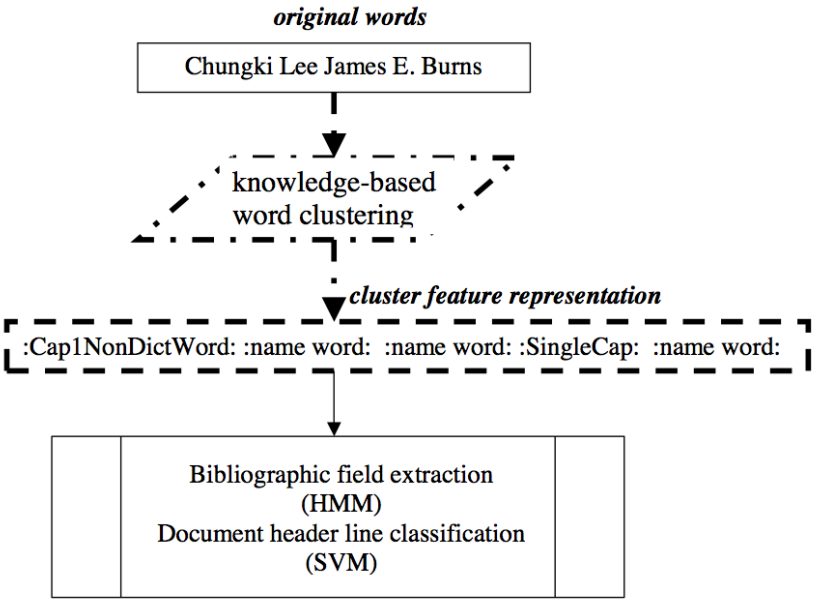
A utilização destes grupos de palavras demonstra uma relação entre palavras semelhantes dentro de um determinado contexto, permitindo que a extração dos metadados possa ocorrer de maneira mais eficaz, com resultados mais eficazes.

Sendo assim, Han et al. apresentou uma ideia de um *cluster* de palavras para promover a extração de metadados de artigos da área de Ciência da Computação, indo de maneira relativamente contrária às propostas tradicionais, que baseiam, geralmente, apenas na ocorrência e estatísticas de palavras isoladas dentro do texto original.

Han et al. agrupou bases de dados de domínios diversos incluindo também propriedades ortográficas de palavras, com base em um conhecimento prévio de classes específicas, como autor, título, etc. Deste modo, palavras encontradas nos documentos vão sendo comparadas com palavras deste *cluster* construído, permitindo identificar, por grupos, características de metadados. Para cada classe cria-se um *cluster*, com suas palavras e propriedades ortográficas específicas.

Como exemplo, a palavra ”Mary” faz parte do contexto de ”nomes”. Portanto existe uma probabilidade maior de ela, juntamente com seu grupo de palavras ao redor, fazer

Figura 5 – Workflow da extração de metadados usando *cluster* de palavras



parte da classe ”autor”, por exemplo. Esta lógica é apresentada para outras classes, como ”e-mail” por exemplo, que pode ser identificado, em sua grande maioria, com a presença do caractere ”@”, o que levaria ao conceito de utilização de expressões regulares para encontrar padrões de ocorrências.

A utilização de *word clustering* tem demonstrado um custo computacional muito baixo, o que pode ser considerado uma vantagem sobre demais técnicas computacionais (HAN et al., 2005).

Han et al. ainda utiliza da técnica de SVM (subseção 2.1.3.1) para classificação de linhas de um cabeçalho de um documento, tanto em função dos bons resultados obtidos quanto também pela boa performance apresentada. Deste modo, cada linha obtida se transforma em um vetor de palavras, que é comparado com seu respectivo *cluster*, melhorando os resultados de classificação.

Além disso é utilizada também a técnica de HMM (subseção 2.1.3.2) para a extração das referências bibliográficas, observando sempre a ocorrência de padrões como título e autor para identificação dos mesmos em referências existentes no documento.

Basicamente a *Rule-based Word Clustering* se resume em 3 (três) etapas. A primeira etapa se resume na construção das bases de dados, assim como (HAN et al., 2003), onde foram utilizadas também como bases externas os nomes apresentados em seu *cluster* (nomes de estados americanos, países, cidades, nomes de dicionários, nomes de pessoas, códigos postais, etc), de maneira a unir, não somente estas bases externas, mas também bases construídas dentro de um domínio específico, como palavras que pertencem a uma classe específica.

A segunda etapa é chamada de *Cluster Design*. Nesta etapa é onde os *clusters* são arquitetados, de maneira a contemplar também propriedades ortográficas das palavras, como se funcionasse de maneira geral como uma expressão regular de palavras, formando então este *cluster* com base nas características apresentadas.

Já a terceira etapa é chamada de *Rule Design*, que resume-se basicamente na combinação de palavras em diferentes domínios, na verificação ortográfica destas palavras e na classificação delas no seu *cluster* correto. Por exemplo, nomes devem começar com a primeira letra maiúscula para então serem classificadas como pertencentes ao *cluster* "nomes".

Além disso, foi observada também a presença de certas palavras que faziam parte de diversos *clusters* ao mesmo tempo, o que permitiu que elas fossem classificadas em seu próprio grupo de palavras, tornando-se então independentes.

A representação através de *clusters* utilizada por Han et al. (HAN et al., 2005) conseguiu reduzir um texto original de 11.223 palavras em um *cluster* de 588 elementos, permitindo ainda que ele fosse distribuído entre as classes distintas, o que tornou o processo muito menos trabalhoso mas, ao mesmo tempo, eficaz.

Como resultado, a utilização desta técnica de clusterização permitiu um ganho considerável de performance, além de permitir uma precisão maior dos resultados, visto que eles são apresentados com base nestas bases focadas em um domínio específico, perfazendo um contexto mais definido e com resultados mais garantidos.

Por outro lado a utilização desta técnica possui uma falha na semântica dos dados. No momento da classificação das classes, na separação e criação dos *clusters*, um dígito ou conjunto deles é substituído por `:number:`. Isso faz com que ele se torne apenas um número qualquer, sem um contexto específico, ou seja, pode ser tanto uma referência a alguma página do documento ou até mesmo um mês de um ano.

2.1.3.4 Conditional Random Fields (CRFs)

CRFs é um framework proposto por Lafferty et al. (LAFFERTY; MCCALLUM; PEREIRA, 2001) criado para construir modelos probabilísticos e dados marcados em sequência (*label sequence data*), geralmente utilizados no reconhecimento de padrões e aprendizado de máquinas (*machine learning*).

Esta técnica oferece algumas vantagens ao se comparar com técnicas mais tradicionais, como HMM (subseção 2.1.3.2), se destacando a habilidade de diminuir pressupostos independentes feitas nestes modelos (LAFFERTY; MCCALLUM; PEREIRA, 2001).

Modelos baseados em HMM possuem uma fraqueza, que é problema denominado *bias problem*. As transições de um dado estado somente competem entre elas, e não com todas as transições presentes no modelo. Elas são feitas com base probabilística de acordo

com o estado inicial e a sequência de observação (*observation sequence*).

Desta forma, devido ao *bias problem*, em um caso extremo, um estado que tiver como opção de transição somente um outro estado, pode simplesmente ignorar a sequência de observação, o que traria efeitos contrários aos objetivos do processo.

Laffarty et al. realiza comparações funcionais e práticas entre CRFs, HMM e MEMM (*Maximum Entropy Markov Models*), uma variação da técnica de HMM, detalhada na subseção 2.1.3.2. A diferença crítica entre CRFs e MEMMs é que MEMM utiliza como probabilidade de ocorrência de um próximo estado modelos exponenciais para cada um deles, enquanto a técnica de CRF possui um modelo exponencial único para toda a sequência de *labels*, com base na sequência de observação. Segundo Laffarty et al. (LAFFERTY; MCCALLUM; PEREIRA, 2001) pode pensar na CRF como um modelo de estado finito sem normalização das probabilidades de transição.

Uma das vantagens de se utilizar CRFs sobre HMM é que ela absorve todas as qualidades deste último, mas com a particularidade de resolver o *bias problem*. Outra grande vantagem ao ser comparada com HMMs e MEMMs é que a CRF possui resultados melhores quando a distribuição dos dados possui grande dependência do modelo, o que geralmente ocorre em casos mais práticos.

Um exemplo para entender o *bias problem* foi apresentado por Lafferty et al. (LAFFERTY; MCCALLUM; PEREIRA, 2001). Ele propõe um modelo cujo objetivo é distinguir duas palavras: `rib` e `rob`, tendo como sequência de observação as letras `r i b`. O problema é identificado quando uma das duas palavras é mais comum no *training set*, o que acarreta nas transições do estado inicial preferirem suas transições correspondentes, o que acarreta sempre na vitória da palavra relacionada aquele estado.

Visando a comprovação e reconhecimento da eficiência da técnica de CRF na extração de informação, foram aplicados dois tipos de experimentos:

1. Verificação direta do *bias problem*;
2. Geração de dados utilizando HMM aleatórios.

Os resultados apontam que HMMs superam MEMMs em virtude do *bias problem*. Por sua vez os CRFs superam os HMMs, sendo então considerada a CRF a melhor técnicas a ser utilizada com base no *training set* utilizado (LAFFERTY; MCCALLUM; PEREIRA, 2001). Outro ganho apresentado pode ser verificado ao agregar algumas características ortográficas à utilização de CRFs, aumentando o poder destes modelos condicionais.

De modo geral as CRFs utilizam a mesma lógica que modelos baseados em Markov (HMMs e MEMMs), se diferenciando nos aspectos probabilísticos para com as transições entre os estados, acarretando em resultados comparativos melhores.

Além disso, CRFs são utilizadas também em marcação e parseamento de dados sequenciais, com uma ordem determinada, como linguagem natural, sequências biológicas (como os genes) ou estados computacionais.

Sua aplicação na extração de metadados foi apresentada por Peng et al. (PENG; MCCALLUM, 2004), como uma maneira eficaz de extrair padrões em cabeçalhos e referências de artigos científicos. Deste modo, através da identificação destes padrões sequenciais pode-se determinar os tipos de dados existentes e então identificá-los, seguindo uma lógica/ordem pré-determinada.

A utilização de CRFs na extração de metadados mostra-se muito eficaz por reduzir bastante os erros em algumas métricas, aumentando o sucesso da aplicação desta técnica nesta área.

2.2 Revisão de Estado da Arte na Extração de Metadados

2.3 Técnicas de Algoritmos

2.4 Ambientes de Extração de Metadados

2.5 Trabalhos Comparativos

Comentário: Falar sobre as comparações realizadas por Granitzer, tanto no que diz respeito a layouts (GRANITZER et al., 2012), quanto no que diz respeito a dicionários (GRANITZER, 2011).

2.6 Ferramentas e Projetos de Destaque

Alguns projetos baseiam sua extração em padrões pré-definidos de maneira a identificar dados relevantes dentro de uma região específica, facilitando a procura e consequentemente aumentando a velocidade no resultado.

Estes projetos geralmente permitem uma variedade muito grande de layouts, embora nem todos já estejam previamente definidos. Geralmente novos layouts são inseridos em novas versões ou até mesmo por contribuições das mais diversas, como é o caso dos projetos de código livre, os chamados projetos *open source*.

Abaixo segue uma relação dos principais projetos relacionados à área de extração de metadados de artigos científicos, com informações sobre seu funcionamento e algoritmos que são utilizados.

2.6.1 Cermine

Um destes projetos é o recente CERMINE (TKACZYK et al., 2014), uma biblioteca *open source* desenvolvida na linguagem de programação Java que permite que sejam extraídos os metadados de artigos científicos em formato digital PDF, oferecendo ainda a possibilidade de cruzamento de dados por meio de referências e títulos, permitindo assim identificar citações bem como a relevância de um determinado documento.

O CERMINE ainda possui um mecanismo de aprendizagem da própria máquina, permitindo que na medida que dados forem sendo alterados ele consiga absorver os detalhes e permitir assim uma mudança de sua maneira de extrair os dados. Deste modo ele permite que seja adaptado para novos padrões de layouts, o que permite de maneira geral que uma grande gama de modelos seja então abrangida.

Seu grande diferencial em comparação com as demais técnicas é que ele não somente extrai os metadados de um artigo, mas sim analisa todo o seu conteúdo, incluindo citações a outros artigos, que podem ser facilmente cruzados por meio de informações como título e autor(es).

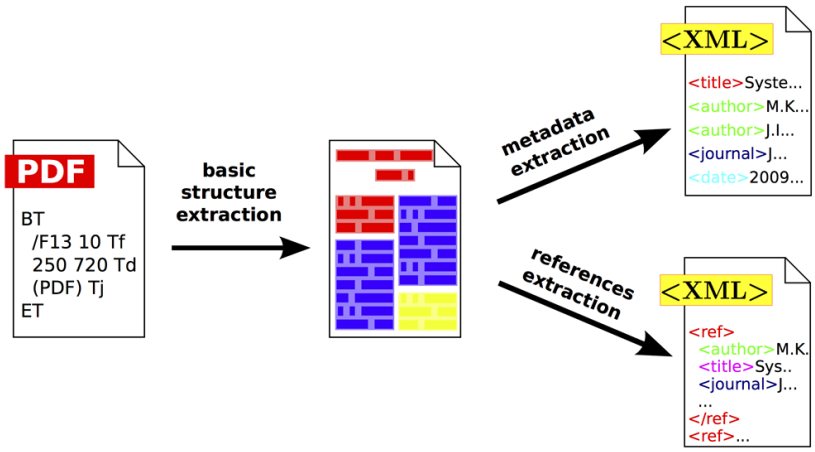
Seu mecanismo considera somente arquivos PDF com texto gerado de maneira pura, sem a utilização de imagens. Basicamente ele considera regiões, linhas e páginas como pontos estratégicos para a extração de informações. As bases destas regiões possuem padrões que são utilizados juntamente com técnicas de SVM (HAN et al., 2003). Com base nisso ele condensa um layout onde as informações geralmente estão dispostas, permitindo assim que em um determinado local do arquivo esteja, provavelmente, o título e o nome dos autores.

Com estas regiões definidas o CERMINE extrai as informações com base em padrões preestabelecidos, de maneira a gerar então sua saída para os metadados e sua saída para as referências encontradas. A saída trabalhada pelo projeto é no formato XML, permitindo assim que possa ser compartilhado com outros sistemas por possuir uma leitura semântica e ao mesmo tempo fácil de ser interpretada pelas linguagens de máquinas. A Figura 6 demonstra como o processo de extração do CERMINE funciona.

Com o mapeamento definido ele identifica regiões de acordo com seu conteúdo, as quais ele chama de *zones*. Esta regiões são determinadas a fim de extrair as informações relevantes para cada uma, de maneira a separar, por exemplo, a área destina aos metadados do arquivo. O CERMINE divide estas *zones* da seguinte maneira:

- **Metadata:** É a região mais ao alto do documento, onde obtém os metadados, que seriam o resumo, *bib_info*, tipo, título, afiliação, autores, datas, editores e palavras-chaves.
- **References:** Região responsável por identificas detalhes de referências que foram

Figura 6 – CERMINE Extraction Workflow



utilizadas no artigo, como título e autores, por exemplo

- **Body:** O texto geral do artigo, incluindo equações, imagens e tabelas.
- **Other:** Outros detalhes menos significantes semanticamente, como número das páginas, dentro outros.

A extração das referências abrange também seus próprios metadados. Tanto no texto corrido (*Body*) quanto na lista de referências do artigo o *parser* do CERMINE analisa linha a linha, permitindo uma extração de dados mais eficaz. Das referências são extraídos os seguintes dados: autor, título, nome do *journal*, volume, *issue*, páginas, *publisher*, localização e o ano.

2.6.2 TeamBeam

Outros projeto de destaque é o TeamBeam (KERN et al., 2012), cuja base ideológica possui objetivos bem sociais, de contribuir com o compartilhamento de conhecimento. Basicamente o objetivo do projeto é extrair metadados de artigos científicos, porém focado apenas nestes, de maneira a extrair título, nome do *journal*, resumo e informações sobre os autores, como nome, endereço de e-mail e afiliações.

O projeto também é de código livre (*open source*) e é baseado na extração de pequenos blocos de texto. A manipulação dos arquivos PDF são feitas pela biblioteca PDFBox [2], que fornece meios eficazes de extrair textos com base nas regiões desejadas.

O algoritmo do TeamBeam utiliza o algoritmo de *Maximum Entropy* (BERGER; PIETRA; PIETRA, 1996), que utiliza basicamente de tarefas de classificação sequencial como ferramenta principal para obtenção de padrões. A base deste algoritmo está na

[2] Biblioteca de manipulação de arquivos PDF mantida pela Fundação Apache. Disponível em <https://pdfbox.apache.org/>

utilização de CRFs (LAFFERTY; MCCALLUM; PEREIRA, 2001), principalmente no que diz respeito à extração dos metadados (PENG; MCCALLUM, 2004).

O processo de extração é feito basicamente em duas etapas. A primeira é a etapa de classificação de blocos de texto (*text block classification*), onde geralmente já é possível obter algum dado concreto de resultado. Neste etapa o objetivo é associar certos blocos de texto a um dos seguintes marcadores: *Title Block*; *Sub-Title Block*; *Journal Block*; *Abstract Block*; *Author Block*; *E-Mail Block*; *Affiliation Block*; *Author-Mixed Block*; e *Other Block*.

Dependendo do layout do artigo alguns metadados podem vir divididos em blocos de texto diferentes, necessitando de um processamento posterior, como é o caso, geralmente, dos blocos com informações sobre os autores. Neste caso também é realizada a etapa de classificação de token (*token classification*), que se resume na classificação de palavras individualmente de acordo com um dos seguintes marcadores: *Given Name*; *Middle Name*; *Surname*; *Index*; *Separator*; *E-Mail*; *Affiliation-Start*; *Affiliation*; e *Other*.

Kern et al. defendem a ideia dos excelentes resultados do TeamBeam ao ser comparado com outros projetos. Este fato é dado em virtude das características que são levadas em consideração no processamento do algoritmo, utilizando de dicionários, informações de layout e modelo de linguagem.

2.6.2.1 Resultados Comparativos

A fim de analisar os resultados obtidos com a aplicação das técnicas descritas no TeamBeam, os autores comparam as técnicas utilizadas com outras técnicas de outras ferramentas, que utilizam processos diferentes de análises.

Para fins de comparação de resultados, os autores citam as técnicas ParsCit (subseção 2.6.7), Layout-CRFs (subseção 2.6.3) e Mendeley Desktop (subseção 2.6.4). Como conclusão ele compara estas três técnicas separadamente, por não abrangerem todos os detalhes que o TeamBeam possui, o que tornaria a comparação desleal.

Assim, ele chega à conclusão que em virtude das diferentes formas de processamento dos dados feitas por cada umas das ferramentas, chega-se a resultados mais precisos para cada campo extraído. Por exemplo, o Layout-CRF's (subseção 2.6.3) é mais abrangente e mais assertivo para extração de títulos, por ser baseado em CRF. Já as ferramentas baseadas em dicionário apresentam melhores resultados para extração de autores, visto se basearem em *data-sets*[3] já consolidados.

Comentário: Devo explicar aqui o que seriam esses Data-Sets? Exemplo: E-Prints, Mendeley e PubMed.

[3] Bases de dados com nomes dos autores mais citados e outras informações já catalogadas que são armazenadas para consulta pública.

2.6.3 Layout-CRFs

Comentário: Escrever sobre.

2.6.4 Mendeley Desktop

Comentário: Escrever sobre.

2.6.5 CiteULike

Comentário: Escrever sobre CiteULike, com base no artigo de seus autores (EMAMY; CAMERON, 2007).

2.6.6 CiteSeer

Um dos projetos mais específicos encontrados é o CiteSeer (GILES; BOLLACKER; LAWRENCE, 1998). Seu objetivo não é apenas extrair dados em artigos, mas sim também analisar citações de outros documentos no conteúdo textual encontrado. Assim, ele é capaz de identificar quais documentos são citados, quantas vezes são citados e onde são citados, se assemelhando muito ao processo de citação natural.

Desta forma, com estas informações identificadas, ele consegue criar um *ranking* dos documentos citados, incluindo seus autores e *journals*, criando a partir de um documento fonte todo um conjunto de relações com outros documentos de uma maneira estatística bem eficaz.

Os indexes de citações (*citation indexes*), que são utilizados no projeto, foram originalmente criados para a recuperação de informação, porém sua utilidade é tamanha que permite que citações sejam indexadas de maneira eficaz, permitindo inclusive que referências de documentos em idiomas diferentes sejam identificadas. Desta maneira, essa técnica pode ser utilizadas de diversas formas, não somente na identificação de citações propriamente ditas, mas envolvendo um conjunto de informações eficazes, como inclusive identificar a reputação de um determinado artigo científico no meio em que se encontra, simplesmente analisando onde é referência em relação ao número de vezes em que é citado.

Uma proposta interessante em que se baseia o CiteSeer é a de Cameron (CAMERON, 1997). Seu objetivo é de formar uma Base de Citações Universal, onde todos os artigos estariam ligados entre si, independente de qualquer fator externo como idioma, por exemplo. A diferença entre esta proposta e o trabalho realizado com o CiteSeer é que este último

permite que os documentos sejam analisados sem nenhum esforço extra, ou seja, sem a intervenção dos autores dos documentos, como é proposto por Cameron. Neste caso, os documentos seriam analisados de maneira automática, diminuindo o tempo necessário para o relacionamento de todos eles e aumentando a eficácia e eficiência do processo.

O funcionamento do CiteSeer é relativamente simples, porém o trabalho realizado por detrás do processo envolve muito estudo e dedicação. Basicamente ele é capaz de fazer o *download* de *papers* utilizando a Internet, convertê-los em texto e realizar a análise de todas as suas citações. Este resultado é armazenado em um banco de dados para consultas e relacionamentos futuros. Um dos pontos interessantes desta análise é que como ela é feita com base em referência textual, identificado origem e destino, ela pode ser facilmente aplicada tanto no sentido natural de leitura (um artigo cita outros) quanto no sentido inverso (um artigo é citado por outros).

O projeto também possui suas desvantagens, como por exemplo, a não cobertura de alguns *journals* importantes, indexando todos seus documentos online e de maneira automática. Além disso o projeto original não é capaz de identificar mais de um autor nos documentos, sendo a identificação apenas pelo campo autor, tendo ele apenas um ou mais de um pesquisador envolvido.

Basicamente o projeto analisa o documento em partes:

- **URL:** a URL onde o documento foi obtido;
- **Cabeçalho:** o bloco de título e autor do documento;
- **Resumo:** o bloco de resumo do documento;
- **Introdução:** a introdução do texto do documento;
- **Citações:** a lista de referências a outros documentos citadas no decorrer do texto;
- **Contexto de citação:** o contexto no qual um documento cita outro;
- **Texto completo:** o texto completo do documento e suas respectivas citações como um todo.

Um dos detalhes importantes do projeto é a identificação das *tags* de citações, que são basicamente as representações visuais quando um outro documento é referenciado, como por exemplo: [4], [Giles997] e "Marr 1982". Estes pequenos pedaços de textos que permitem ao CiteSeer identificar a relação (e em que momento do texto) entre documentos, permitindo assim que suas análises sejam realizadas de maneira eficaz.

Uma das dificuldades relatadas durante o desenvolvimento do projeto é a identificação de artigos iguais com formas de escrita e informações divergentes. Alguns artigos

podem vir com autores com sobrenomes utilizando abreviação, ou até mesmo em ordem diferente. A fim de aumentar esta identificação alguns passos a mais são realizados, como a conversão para caixa baixa todas as letras, remoção de hifens e das próprias *tags* de citação, expansão de abreviaturas e remoção de algumas palavras externas como "*volume*", "*pages*" e "*number*" por exemplo.

O CiteSeer é um projeto bem maduro e abrangente. Após cerca de 18 (dezoito) anos desde sua criação ele ainda se encontra ativo na Internet, abrangendo cada vez mais documentos e aprimorando cada vez mais suas técnicas de identificação de documentos.

2.6.7 ParsCit

Assim como o CiteSeer o ParsCit é um projeto baseado na identificação de citação de documentos. Porém eles utiliza um modelo de CRF *(Conditional Random Fields)* para identificar sequências nas referências.

O projeto encontra-se ainda ativo e possui atuais contribuições de desenvolvedores ao longo do mundo. Desenvolvido na linguagem de programação *Perl*, o projeto pode ser executado tanto na forma de um *web service*[4] quanto de maneira *standalone*, com execução do código direto quando necessário, dentro do servidor.

Um detalhe bastante interessante do projeto é a descrição de seu processamento antes e depois da análise do documento, que o autor chama de *Pre-Processing Steps* e *Post-Processing Steps*.

Inicialmente o processamento inicia buscando certos *tokens* que podem estar ligados a alguma referência, seja em formato numérico, seja na citação dos nomes dos autores nos mais diversos formatos. Com essa referência coletada o próximo passo é buscar dentro do artigo onde ela está localizada, com base em um conjunto de heurísticas previamente definidas. Para isso é necessária a conversão total do conteúdo do artigo para o formato texto, que deve estar codificado em UTF-8[5].

Desta forma a fase de pré-processamento se inicia com esta analise puramente textual, em busca de padrões comuns que podem ter sido utilizados para representações de referências à artigos científicos. Com os resultados coletados um modelo CRF é então aplicado aos dados encontrados para processamento futuro.

Com este modelo CRF definido algumas etapas são aplicadas a fim de normalizar os dados encontrados. Nomes de autores podem estar escritos de maneira diferente, com abreviações em nomes de maneira diferente, dependendo do modo como as referências foram escritas no artigo analisado.

[4] É a disponibilização de algum serviço na Internet permitindo que outros projetos consultem suas informações e as obtém de maneira simplicada.

[5] Formato de exibição de caracteres que englobam os formatos mais utilizados no mundo, com acentuação e caracteres especiais.

Esta análise posterior dos nomes dos autores é feita com base na inspeção de cada palavra individualmente, respeitando então um padrão de normalização definido, sempre com as iniciais dos nomes seguidas do sobrenome escrito normalmente.

A utilização deste projeto é feita de maneira muito simples, podendo ser executado por linha de comando, o que facilitam os testes e a extração de dados. Os resultados obtidos desta análise é então exibido (ou gravado em arquivo) em formato XML, que permite utilização posterior em qualquer tecnologia. Como informado anteriormente os resultados também podem ser coletados em formato de *Web Service*, mas para os fins desta pesquisa apenas sua execução em linha de comando é suficiente.

2.6.7.1 Resultados Comparativos

Visando comparar os resultados obtidos com alguns projetos em que o ParsCit foi baseado, os autores realizam comparações com os resultados obtidos pelo processo descrito por Peng (PENG; MCCALLUM, 2004), onde de maneira geral obteve uma melhora na extração e comparação das referências em torno de 5% (precisão de 0.91 passou para 0.95), demonstrando a eficácia do projeto.

Visando também analisar outros projetos a mesma comparação de resultados foi feita com o projeto CiteSeer (GILES; BOLLACKER; LAWRENCE, 1998), onde o aumento da qualidade dos dados extraídos foi de aproximadamente 19% levando em consideração algumas limitações do CiteSeer que não são compreendidas como no ParsCit.

Comentário: Seria necessário falar também do Weka (*The open-source machine learning framework*)? Até que ponto ele seria relevante para a pesquisa em si? Neste caso tenho medo de fugir muito do escopo e detalhar demais técnicas.

3 Metodologia

3.1 Escolha do Corpus

3.2 Desenho do Experimento

3.3 Ambiente Tecnológico

Este trabalho tem como metodologia uma pesquisa de caráter não-experimental e quantitativa, por se tratar de coleta de informações e comparação de resultados de técnicas de extração de metadados em artigos científicos.

Desta maneira, a pesquisa de modo padrão não traz alteração nos ambientes pesquisados, apenas os analisa e os compara com base em padrões estabelecidos como sendo de resultado adequado. Assim, o projeto em si trata muito mais de pré-seleção de documentos e técnicas a serem testadas como também dos resultados já analisados e que são necessários para uma técnica ser considerada produtiva.

Primeiramente, são filtradas as técnicas encontradas a fim de analisar realmente as que são necessárias dentro do objetivo da pesquisa, fazendo do projeto o mais conciso possível. Desta forma, diversos elementos serão utilizados a fim de se obter os resultados desejados. Os elementos se relacionam entre si e estão identificados na Figura 7.

3.4 Seleção das Técnicas/Ferramentas

As técnicas que foram analisadas dentro do capítulo de **Revisão de Literatura** compreendem um universo atuante e de caráter livre, independente da linguagem de programação que foi utilizada no desenvolvimento, exceto pelos detalhes explicados nas limitações do trabalho.

Em função do grande número de ferramentas disponíveis como resultado desta pesquisa, algumas foram selecionadas a fim de se obter ao máximo os testes e os resultados esperados.

Conforme mencionado anteriormente, ferramentas pagas ou que não disponibilizam o código para instalação não foram testadas, visto que a necessidade do atual trabalho visa identificar os resultados obtidos dentro de um conjunto de infra estrutura única, de maneira justa com cada projeto.

Assim sendo, a Tabela 2 lista as seguintes ferramentas que foram selecionadas para

Figura 7 – Processo de Metodologia

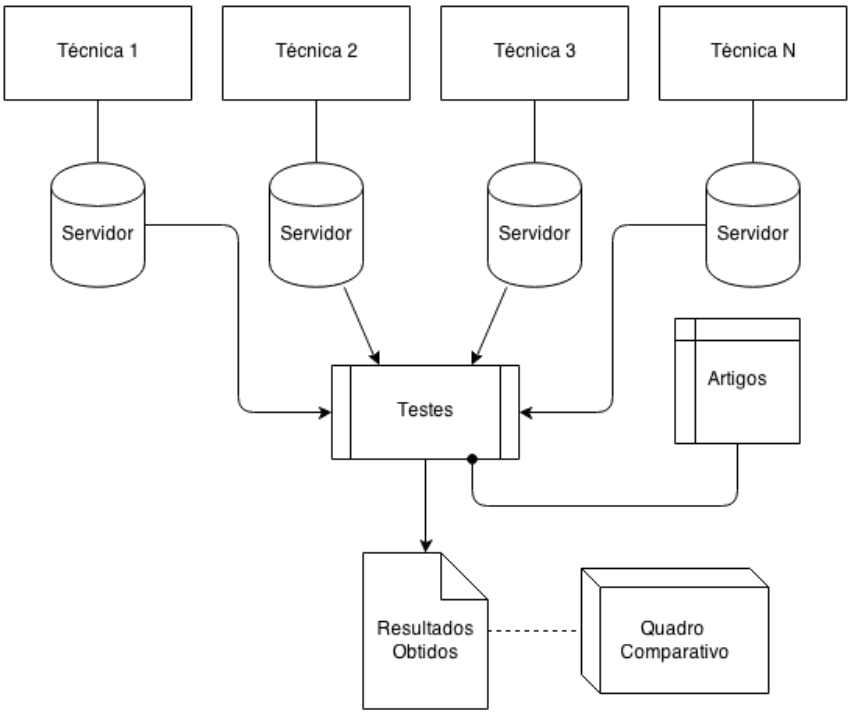


Tabela 2 – Ferramentas selecionadas para testes

Ferramenta	**Linguagem**
Ferramenta 1	Java
Ferramenta 2	Perl
Ferramenta 3	Python
Ferramenta 4	Java

teste dentro do escopo proposto neste trabalho.

3.5 Seleção de Artigos

Visando provar a eficiência destas técnicas, desejamos ter informações de saída consideradas corretas para que seus resultados possam ser comparados e verificados com exatidão. Assim, foi selecionada uma série de artigos científicos das mais diversas áreas de pesquisa, de diversos eventos distintos, com padrões visuais totalmente diferentes e que são possível de ser analisados e coletados.

Desde modo a lista destes artigos compreende um total de 100 documentos variados, com seus metadados já extraídos manualmente e todos catalogados a fim de ter o resultado da aplicação das técnicas comparado com os resultados obtidos manualmente. Na Tabela 3 pode-se ver o número de artigos que foram selecionados por cada área do conhecimento. A relação de todos os artigos analisados pode ser obtida no Apêndice A deste documento.

Tabela 3 – Seleção de artigos científicos para testes (ainda em desenvolvimento)

Área de Conhecimento	**Total de Artigos**
Arquitetura e Urbanismo	7
Música	7
Ciência da Computação	8
Ciência da Informação	9
Ciências Biológicas	7
Direito	7
Engenharia Civil	8
Letras	7
Matemática Computacional	7
Medicina	9
Odontologia	8
Psicologia	9
Sociologia	7
Total	**100**

Os artigos foram selecionados tomando como base a principal forma de análise das técnicas selecionadas para testes: o layout, ou seja, a disposição dos elementos nos artigos científicos. Esta seleção foi feite com base a abranger um maior número de representações, com disposições diferentes, tipografias diferentes e inclusive ordem diferentes nas exibições. Desta forma as técnicas poderão ser confrontadas e os resultados comparados com os resultados esperados.

Todos os artigos selecionados foram escritos na língua inglesa. Esta decisão foi tomada em virtude de além de ser a língua inglesa universal para disseminação de conhecimento, ela é a mais utilizada no meio acadêmico, de maneira a ter um universo muito maior de artigos escrito em inglês do que outras línguas.

Além disso a abrangência de outros idiomas entraria em um aspecto que não é objetivo deste trabalho abordar, visto a diversificação de culturas e símbolos, fazendo com que línguas orientais, como o mandarim ou japonês, tenham análises diferenciadas em função de suas diferenças na forma de representação.

3.6 Infraestrutura Computacional

Para que os testes sejam feitos de maneira adequada e independente, sem interferência de outras técnicas nos resultados, serão utilizados N servidores, sendo N o número total de técnicas a serem avaliadas.

Deste modo, para cada técnica avaliada será configurado um servidor com as linguagens de programação necessárias e todos os pré-requisitos que a técnica necessita

para funcionar. Estes servidores serão definidos através de máquinas virtuais, o que traz benefícios não somente de performance mas de flexibilidade quanto das tecnologias necessárias para o funcionamento de cada técnica.

3.6.1 Testes In Logo

Algumas técnicas de extração de metadados disponibilizam acesso online a uma ferramenta gratuita para testes. Deste modo, para estes casos específicos não será necessária a criação de servidores e instalação dos pacotes, visto que o ambiente de testes poderá ser feito dentro da ferramente fornecida pelas desenvolvedoras das técnicas.

Este fato garante uma maior precisão nos resultados, inclusive pelo fato de o ambiente de testes estar 100% funcionando e ter sido disponibilizado pela mesma equipe de desenvolvedores do projeto, garantindo a eficácia nos resultados que essa técnicas fornece.

3.7 Quadro Comparativo

Visando uma comparação dos resultados eficaz todos os resultados serão inseridos em um documento formato planilha para serem comparados manualmente e as conclusões então obtidas. Para tal esta planilha de resultados a chamaremos de 'Quadro Comparativo"e será exclusiva para comparação dos resultados das técnicas analisadas.

Com o objetivo de facilitar o acesso às informações este documento é disponibilizado como anexo deste projeto e ainda tem seu acesso liberado em formato online. Desta forma qualquer pessoa poderá ter acesso aos resultados obtidos pelos testes e comparar elas mesmas os dados contidos na planilha.

3.8 Metadados, Pesos e Resultados

A extração de metadados de artigos científicos engloba um processo onde os resultados obtidos, mais especificamente os metadados propriamente ditos, possuem características diferenciadas que podem influenciar em uma busca por artigos, feita por um pesquisador.

Desde modo atribuímos pesos para cada um dos metadados analisados, de maneira a identificar os mais importantes e que podem contribuir com um número maior de resultados de busca.

Alguns metadados são mais importantes que outros no que diz respeito à funcionalidade de pesquisa. Geralmente quando vamos buscar artigos, seja na Internet, ou em algum outro local, geralmente buscamos primeiro pelo título do artigo (quando procuramos por

Tabela 4 – Os metadados e seus pesos atribuídos

Metadado	**Relevância**	**Peso**
Título	Um dos termos mais buscados quando se pesquisa um artigo	5
Autor(es)	O segundo termo mais pesquisado	4
E-mail(s)	Pouco relevante no quesito pesquisa de artigos	1
Resumo	Importante por conter palavras chaves e o resumo propriamente dito	3
Referências	Muito importante e necessário, pois será utilizada na referência inversa de autores	5

um artigo em específico) ou então pelo nome do autor (quanto procuramos artigos de um determinado autor).

Além disso, utilizamos também o título, juntamente com o resumo, para buscar de palavras chaves ou palavras que podem ser relevantes na pesquisa pelos documentos. Assim sendo alguns metadados devem ser mais considerados no resultado destas extrações, por serem mais importantes no ponto de vista da busca.

Assim sendo apresentamos a Tabela 4, que demonstra como cada metadado teve sua importância interpretada e qual o peso que lhe foi atribuído, sendo utilizado o inteiro 1 para o peso mais baixo e o 5 para peso mais alto, sendo consequentemente o mais importantes.

Outro detalhe importante é a precisão de cada resultado para cada metadado analisado. Em alguns casos o título, por exemplo, não é extraído em 100% mas alguma variação dele.

Deste modo consideramos 3 (três) resultados possíveis para um resultado analisado:

1. **Preciso:** Quando um resultado atinge acima de 95% de precisão, ou seja, o campo foi extraído em 95% ou mais de sua totalidade.

2. **Satisfatório:** Quando um resultado atinge entre 90 e 94%, o que pode ser considerado satisfatório e a maioria do conteúdo consegue ser analisada sem maiores problemas.

3. **Inaceitável:** Quando o resultado atinge abaixo de 90%, ou seja, entre 0% e 89%. Este resultado no âmbito do presente projeto é considerado inaceitável.

Assim, temos o valor de cada resultado possível, que será também utilizado no processo de análise, conforme consta na Tabela 5. Cada valor, independente de sua classificação será utilizado para o cálculo final da nota de cada ferramenta.

Tabela 5 – Resultados obtidos em cada metadado e sua precisão

Resultado	**Precisão**
Preciso	1
Satisfatório	0.60
Inaceitável	0

Tabela 6 – Descrição de cada variável no Índice de Confiabilidade

Variável	**Descrição**
PT	Precisão na obtenção do título
PA	Precisão na obtenção do(s) autor(es)
PE	Precisão na obtenção dos e-mails dos autores
PR	Precisão na obtenção do resumo
PRF	Precisão na obtenção das referências

3.8.1 Índice de Confiabilidade

Considerando que cada metadado possui um peso diferente necessitamos calcular o índice de acertos a ser utilizado em cada resultado coletado para cada técnica aplicada. Assim sendo chegamos em uma fórmula matemática à qual chamamos "Índice de Confiabilidade", que calcula o resultado obtido através dos pesos que foram atribuídos a cada campo.

Este índice utiliza os pesos anteriormente definidos e a precisão dos resultados obtida, de maneira a permitir chegar em uma única nota final para cada técnica aplicada.

Basicamente a fórmula atua como uma média ponderada dos resultados alcançados na extração de cada campo do artigo, seguindo os pesos apresentados em Tabela 4. Cada peso é atribuído ao resultado encontrado em cada ferramenta testada.

A título de exemplo, após o teste de uma ferramenta, supondo que ela conseguiu extrair 87% dos títulos dos artigos com sucesso, sua precisão com relação ao título será 87, que será multiplicado pelo peso correspondendo ao seu campo, neste caso, o inteiro 5. Isso ocorre para todos os campos apresentados, seguindo seus respectivos pesos.

A descrição de cada variável no Índice de Confiabilidade poderá ser obtida de acordo com a Tabela 6, onde temos as variáveis que fazem parte da fórmula descrita.

$$IC = \frac{5(PT)+4(PA)+PE+3(PR)+5(PRF)}{18}$$

4 Análise e Apresentação de Resultados

Com base no objetivo de identificar como as ferramentas atuais se comportam nos mais diversos tipos de padrões visuais foram realizados vários experimentos práticos a fim de obter os resultados numéricos desejados para que o objetivo fosse alcançado.

Desta maneira, foram separadas algumas ferramentas/técnicas implementáveis tecnicamente para que os artigos selecionados (citados no capítulo anterior) pudessem ser analisados e seus dados extraídos segundo os propósitos de cada uma.

Para que os resultados pudessem ser medidos e comparados foi utilizado o "Índice de Confiabilidade"detalhado no capítulo de **Metodologia**, a fim de obter um valor numérico para cada resultado apresentado com a utilização das ferramentas, permitindo que os valores obtidos fossem comparados e então confrontados para se obter o comportamento de cada ferramenta.

4.1 Ambiente de Testes

Para que esta análise pudesse ser feita com exatidão e garantir assim os resultados esperados, foi-se criado um ambiente de testes abrangendo um conjunto de tecnologias para que as ferramentas pudessem ser instaladas e testas seguindo o propósito de cada uma.

4.1.1 Servidores de Teste

Para cada teste de ferramenta foi criada uma Máquina Virtual dentro do Ambiente de Testes, com o objetivo de ter plataformas operando independentemente, com apenas as tecnologias necessárias para seu correto funcionamento, sem influência de códigos terceiro e/ou programas desnecessários.

Por restringir apenas os testes utilizando ferramentas de código livre, todos os testes realizados ocorreram em máquinas Linux[1], de acordo com a Tabela 7, respeitando os requisitos necessários para cada ferramenta.

[1] Sistema operacional de código livre com ampla utilização em servidores de todo o mundo.

Tabela 7 – Ferramentas e Ambientes de Testes

Ferramenta	**Ambiente**
Ferramenta 1	Linux Ubuntu 12.04, Java 7, Sqlite, Tomcat 6
Ferramenta 2	Linux Ubuntu 12.04, Perl, MySQL, Nginx
Ferramenta 3	Linux Ubuntu 12.04, Perl, PostgreSQL, Apache
Ferramenta 4	Linux Ubuntu 12.04, Java 6, MySQL, Tomcat 6

5 Discussão / Trabalhos Futuros

5.1 Contribuições

5.2 Trabalhos Futuros

5.3 Considerações Finais

Referências

BERGER, A. L.; PIETRA, S. A. D.; PIETRA, V. J. D. A maximum entropy approach to natural language processing. 1996. Citado 2 vezes nas páginas 31 e 37.

CAMERON, R. D. A universal citation database as a catalyst for reform in scholarly communication. v. 2, n. 4, 1997. Citado na página 39.

CATHRO, W. Metadata: an overview. 1997. Disponível em: <http://www.nla.gov.au/openpublish/index.php/nlasp/article/view/1019/1289>. Citado na página 21.

CHIEU, H. L.; NG, H. T. *A Maximum Entropy Approach to Information Extraction from Semi-Structured and Free Text*. 2002. Citado na página 24.

EMAMY, K.; CAMERON, R. Citeulike: A researcher's social bookmarking service. 2007. Disponível em: <http://www.ariadne.ac.uk/issue51/emamy-cameron/>. Citado na página 39.

GILES, C. L.; BOLLACKER, K. D.; LAWRENCE, S. Citeseer: An automatic citation indexing system. p. 89–98, 1998. Citado 2 vezes nas páginas 39 e 42.

GRANITZER, M. *Results and Raw Experimental Data on the E-Prints Data Set for A Comparison of Metadata Extraction Techniques for Crowed-source Bibliographic Metadata Management*. London: Mendeley, 2011. Citado na página 35.

GRANITZER, M. et al. A comparison of layout based bibliographic metadata extraction techniques. In: . [S.l.]: ACM, 2012. Citado na página 35.

HAN, H. et al. Automatic document metadata extraction using support vector machines. 2003. Citado 4 vezes nas páginas 24, 26, 32 e 36.

HAN, H. et al. Rule-based word clustering for document metadata extraction. p. 1049–1053, 2005. Citado 3 vezes nas páginas 31, 32 e 33.

KERN, R. et al. Teambeam — meta-data extraction from scientific literature. v. 18, n. 7/8, 2012. Disponível em: <http://www.dlib.org/dlib/july12/kern/07kern.html>. Citado na página 37.

KITCHENHAM, B. *Procedures for Performing Systematic Reviews*. [S.l.], 2004. Citado 2 vezes nas páginas 18 e 20.

KUNZE, J.; BAKER, T. *The Dublin Core Metadata Element Set*. [S.l.], 2007. Disponível em: <http://www.ietf.org/rfc/rfc5013.txt>. Citado na página 22.

LAFFERTY, J.; MCCALLUM, A.; PEREIRA, F. C. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: ACM (Ed.). *Proceedings of the 18th International Conference on Machine Learning 2001 (ICML 2001)*. [S.l.]: ACM, 2001. p. 282–289. Citado 4 vezes nas páginas 31, 33, 34 e 38.

MOHRI, M.; ROSTAMIZADEH, A.; TALWALKAR, A. *Foundations of Machine Learning*. [S.l.: s.n.], 2012. ISBN 978-0-262-01825-8. Citado na página 17.

PENG, F.; MCCALLUM, A. Accurate information extraction from research papers using conditional random fields. In: *HLT-NAACL04*. [S.l.: s.n.], 2004. p. 329–336. Citado 3 vezes nas páginas 35, 38 e 42.

RABINER, L. R.; JUANG, B. H. An introduction to hidden markov models. 1986. Citado 2 vezes nas páginas 26 e 27.

SEYMORE, K.; ROSENFIELD, R. Learning hidden markov model structure for information extraction. p. 37–42, 1999. Citado 3 vezes nas páginas 24, 28 e 29.

SOLOMONOFF, R. An inductive inference machine. 1956. Disponível em: <http://world.std.com/~rjs/indinf56.pdf>. Citado na página 16.

TKACZYK, D. et al. Cermine — automatic extraction of metadata and references from scientific literature. 2014. Citado na página 36.

VAPNIK, V.; CORTES, C. Support-vector networks. In: PUBLISHERS, B. K. A. (Ed.). *Machine Learning*. [S.l.]: Kluwer Academic Publishers, Boston, 1995. v. 20, p. 273–297. Citado 2 vezes nas páginas 23 e 25.

WEIBEL, S. The dublin core: A simple content descrip- tion format for electronic resources. In: SCIENCE, B. of the American Society for I. (Ed.). [S.l.: s.n.], 1997. p. 9–11. Citado na página 22.

ZHANG, N. R. Hidden markov models for information extraction. 2001. Citado 3 vezes nas páginas 7, 29 e 30.

Apêndices

APÊNDICE A – Listagem de artigos analisados para realização do experimento

#	**Título *(Autores)***	**Área/Disciplina**
1	Benefits of extra-pair mating may depend on environmental conditions—an experimental study in the blue tit (Cyanistes caeruleus) *(Aneta Arct, Szymon M. Drobniak, Edyta Podmokta, Lars Gustafson, Mariusz Cichoń)*	Ciências Biológicas
2	Cost-Effective Suppression and Eradication of Invasive Predators *(Peter W. J. Baxter, John L. Sabo, Chris Wilcox, Michael A. McCarthy, Hugh P. Possingham)*	Ciências Biológicas
3	Model selection and model averaging in behavioural ecology: the utility of the IT-AIC framework *(Shane A. Richards, Mark J. Whittingham, Philip A. Stephens)*	Ciências Biológicas
4	Territory size of the flavescent warbler, Basileuterus flaveolus (Passeriformes, Emberizidae), in a forest fragment in Southeastern Brazil *(Charles Duca, Miguel Â. Marini)*	Ciências Biológicas
5	Effectiveness of Corridors Relative to Enlargement of Habitat Patches *(Mattew R. Falcy, Cristián F. Estades)*	Ciências Biológicas
6	High fidelity on islands: a comparative study of extrapair paternity in passerine birds *(Simon C. Griffith)*	Ciências Biológicas
7	Parentage and Relatedness in Polyandrous Comb-Crested Jacanas Using ISSRs *(S. M. Haig, T. R. Mace, T. D. Mullins)*	Ciências Biológicas
8	Factors influencing double brooding in Eurasian Hoopoes Upupa epops *(Jael Hoffmann, Erik Postma, Michael Schaub)*	Ciências Biológicas
9	The importance of fission–fusion social group dynamics in birds *(Matthew J. Silk, Darren P. Croft, Tom Tregenza, Stuart Bearhop)*	Ciências Biológicas
10	Genetic monogamy in two long-lived New Zealand passerines *(Sabrina S. Taylor, Sanne Boessenkool, Ian G. Jamieson)*	Ciências Biológicas
11	A Probabilistic Framework for Semi-Supervised Clustering *(Sugato Basu, Mikhail Bilenko, Raymond J. Mooney)*	Ciência da Computação

12	Document Clustering: The Next Frontier *(David C. Anastasiu, Andrea Tagarelli, George Karypis)*	Ciência da Computação
13	Enhancing Semi-Supervised Document Clustering with Feature Supervision *(Yeming Hu, Evangelos E. Milios, James Blustein)*	Ciência da Computação
14	Instance-Level Constraint-Based Semisupervised Learning With Imposed Space-Partitioning *(Jayaram Raghuram, David J. Miller, George Kesidis)*	Ciência da Computação
15	Alternatives to the k-means algorithm that find better clusterings *(Greg Hamerly, Charles Elkan)*	Ciência da Computação
16	Towards Parameter-Free Data Mining *(Eamonn Keogh, Stefano Lonardi, Chotirat Ann Ratanamahatana)*	Ciência da Computação
17	Learning Nonparametric Kernel Matrices from Pairwise Constraints *(Steven C. H. Hoi, Rong Jin, Michael R. Lyu)*	Ciência da Computação
18	Enhancing Semi-Supervised Clustering: A Feature Projection Perspective *(Wei Tang, Hui Xiong, Shi Zhong, Jie Wu)*	Ciência da Computação
19	Integrating Constraints and Metric Learning in Semi-Supervised Clustering *(Mikhail Bilenko, Sugato Basu, Raymond J. Mooney)*	Ciência da Computação
20	On Weighting Clustering *(Richard Nock, Frank Nielsen)*	Ciência da Computação
21	Masculinity, Sexuality and the Body of Male Soldiers *(Nyameka Mankayi)*	Psicologia
22	Consumption of alcohol and depression in students of a public school of CoatzaCoalCos, VeraCruz, Mexico *(Brenda Alicia Hernández-Cortaza, Leticia Cortaza-Ramirez, Moacyr Lobo da Costa Junior)*	Psicologia
23	Psychometric Properties of the Spanish Language Version of the Stress in Children Questionnaire (SiC) *(Alejandra Caqueo-Urízar, Alfonso Urzúa, Walter Osika)*	Psicologia
24	Facial information processing in schizophrenia *(João Paulo Machado de Sousa, Jaime Eduardo Cecílio Hallak)*	Psicologia
25	Intelligence and socioeconomic success: A meta-analytic review of longitudinal research *(Tarmo Strenze)*	Psicologia

26	The Science of Sex Differences in Science and Mathematics *(Diane F. Halpern, Camilla P. Benbow, David C. Geary, Ruben C. Gur, Janet Shibley Hyde, Morton Ann Gernsbacher)*	Psicologia
27	Hardiness and Burnout Syndrome: A Cross-Cultural Study among Portuguese and Brazilian Nurses *(Mary Sandra Carlotto, Cristina Queirós, Sofia Dias, Mariana Kaiseler)*	Psicologia
28	Social Psychology: Fundamentals and Fundamentalisms *(Marcus Eugênio Oliveira Lima)*	Psicologia
29	Computerized Assessment of Food Preferences in Adolescents in the Stimulus Equivalence Paradigm *(Gisele Straatmann, Sebastião Sousa Almeida, Julio C. de Rose)*	Psicologia

Anexos

ANEXO A – Elementos do padrão Dublin Core, versão 1.1

Name	**Label**	**Definition**	**Comment**
title	Title	A name given to the resource.	
creator	Creator	An entity primarily responsible for making the resource.	Examples of a Creator include a person, an organization, or a service. Typically, the name of a Creator should be used to indicate the entity.
subject	Subject	The topic of the resource.	Typically, the subject will be represented using keywords, key phrases, or classification codes. Recommended best practice is to use a controlled vocabulary. To describe the spatial or temporal topic of the resource, use the Coverage element.
description	Description	An account of the resource.	Description may include but is not limited to: an abstract, a table of contents, a graphical representation, or a free-text account of the resource.
publisher	Publisher	An entity responsible for making the resource available.	Examples of a Publisher include a person, an organization, or a service. Typically, the name of a Publisher should be used to indicate the entity.
contributor	Contributor	An entity responsible for making contributions to the resource.	Examples of a Contributor include a person, an organization, or a service. Typically, the name of a Contributor should be used to indicate the entity.

date	Date	A point or period of time associated with an event in the lifecycle of the resource.	Date may be used to express temporal information at any level of granularity. Recommended best practice is to use an encoding scheme, such as the W3CDTF profile of ISO 8601 [W3CDTF].
type	Type	The nature or genre of the resource.	Recommended best practice is to use a controlled vocabulary such as the DCMI Type Vocabulary [DCTYPE]. To describe the file format, physical medium, or dimensions of the resource, use the Format element.
format	Format	The file format, physical medium, or dimensions of the resource.	Examples of dimensions include size and duration. Recommended best practice is to use a controlled vocabulary such as the list of Internet Media Types [MIME].
identifier	Identifier	An unambiguous reference to the resource within a given context.	Recommended best practice is to identify the resource by means of a string conforming to a formal identification system.
source	Source	A related resource from which the described resource is derived.	The described resource may be derived from the related resource in whole or in part. Recommended best practice is to identify the related resource by means of a string conforming to a formal identification system.
language	Language	A language of the resource.	Recommended best practice is to use a controlled vocabulary such as RFC 4646 [RFC4646].
relation	Relation	A related resource.	Recommended best practice is to identify the related resource by means of a string conforming to a formal identification system.

coverage	Coverage	The spatial or temporal topic of the resource, the spatial applicability of the resource, or the jurisdiction under which the resource is relevant.	Spatial topic and spatial applicability may be a named place or a location specified by its geographic coordinates. Temporal topic may be a named period, date, or date range. A jurisdiction may be a named administrative entity or a geographic place to which the resource applies. Recommended best practice is to use a controlled vocabulary such as the Thesaurus of Geographic Names [TGN]. Where appropriate, named places or time periods can be used in preference to numeric identifiers such as sets of coordinates or date ranges.
rights	Rights	Information about rights held in and over the resource.	Typically, rights information includes a statement about various property rights associated with the resource, including intellectual property rights.