UNIVERSIDADE FEDERAL DO ESTADO DO RIO DE JANEIRO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

ATHENAS: Uma Avaliação Experimental da Combinação de Métricas de Similaridade para o Alinhamento de Ontologias através de Mineração de Dados.

Alex Alves da Silva

Orientadores

Kate Cerqueira Revoredo

Fernanda Araújo Baião

RIO DE JANEIRO, RJ – BRASIL

Setembro de 2013

ATHENAS: UMA AVALIAÇÃO EXPERIMENTAL DA COMBINAÇÃO DE MÉTRICAS DE SIMILARIDADE PARA O ALINHAMENTO DE ONTOLOGIAS ATRAVÉS DE MINERAÇÃO DE DADOS.

Alex Alves da Silva

DISSERTAÇÃO APRESENTADA COMO REQUISITO PARCIAL PARA OBTENÇÃO DO TÍTULO DE MESTRE PELO PROGRAMA DE PÓS GRADUAÇÃO EM INFORMÁTICA DA UNIVERSIDADE FEDERAL DO ESTADO DO RIO DE JANEIRO (UNIRIO). APROVADA PELA COMISSÃO EXAMINADORA ABAIXO ASSINADA.

Aprovada por:

Kate Cerqueira Revoredo, D.Sc. – UNIRIO

Fernanda Araujo Baião Amorim, D.Sc. – UNIRIO

Bianca Zadrozny, D.Sc. – UFF e IBM

Sean Wolfgand Matsui Siqueira, D.Sc. – UNIRIO

RIO DE JANEIRO, RJ – BRASIL

SETEMBRO DE 2013

Silva, Alex Alves da.
S586 Athenas : uma avaliação experimental da combinação de métricas de similaridade para o alinhamento de ontologias através de mineração de dados / Alex Alves da Silva, 2013.
133 f. ; 30 cm

Orientadora: Kate Cerqueira Revoredo.
Coorientadora: Fernanda Araújo Baião.
Dissertação (Mestrado em Informática) - Universidade Federal do Estado do Rio de Janeiro, Rio de Janeiro, 2013.

1. Ontologias. 2. Alinhamento de ontologias. 3. Mineração de dados (Computação). 4. Aprendizado do computador. I. Revoredo, Kate Cerqueira. II. Baião, Fernanda Araújo. III. Universidade Federal do Estado do Rio de Janeiro. Centro de Ciências Exatas e Tecnológicas. Curso de Mestrado em Informática. IV. Título.

CDD - 006.312

"A grandeza não consiste em receber honras, mas em merecê-las".- Aristóteles

“Imagine o mundo em que cada pessoa tenha acesso gratuito à soma de todo o conhecimento da humanidade.” - Jimmy Wales, Fundador da Wikipédia.

Aos meus pais Pedro e Margarida

Agradecimentos

A Deus pelo dom da vida.

Aos meus pais que com toda simplicidade e humildade me educaram, mostrando-me que sonho e luta são essenciais para a conquista.

A minha esposa, Ana Paula que soube compreender minhas ausências e conseguiu dar conta dos nossos filhos enquanto estive dedicado ao Mestrado.

Aos meus lindos filhos João Pedro e Maria Luísa.

A todos familiares e amigos, pelo apoio e companheirismo que vivenciamos.

As professoras Kate e Fernanda pela orientação e ensinamentos, mas acima de tudo pela força e doses de motivação que injetaram durante os momentos mais difíceis. Pessoas maravilhosas que convivi durante esses dois anos e meio. Muitas experiencias e ensinamentos que levarei comigo durante a minha vida. Obrigado por tudo professoras.

Ao professor Sean que me ensinou e me incentivou no aprendizado do tema ontologia, durante as suas aulas no mestrado. E também durante os seminários com suas dicas sempre oportunas e cirúrgicas. Valeu Sean.

Aos meus Gerentes na Dataprev que compreenderam as minhas ausências durante o mestrado, em especial aos meus chefes: Nelson Simabuguro, André Luiz, Cláudia Maria Gama, Mônica Cavalcante e Edgar Prates. Aos amigos da Dataprev, em especial ao Elberth e Victor Padilha, e também Fritzen, Fabiano, Carlos Leão, Tosta, Rainer, Pablo, Victor Macedo e Alice que me ajudaram nessa caminhada, com dicas, codificação, cedendo as suas máquinas para eu terminar os experimentos a tempo. Obrigado a todos vocês.

Silva, Alex Alves. **ATHENAS: Uma Avaliação Experimental da Combinação de Métricas de Similaridade para o Alinhamento de Ontologias através de Mineração de Dados.** UNIRIO, 2013.2 133 Páginas. Dissertação de Mestrado. Departamento de Informática Aplicada, UNIRIO.

RESUMO

Ontologias têm atraído a atenção da comunidade científica, especialmente no campo da Web Semântica. Uma Ontologia descreve um domínio de conhecimento em termos de suas entidades (conceitos, relacionamentos e instâncias), permitindo assim o compartilhamento de conhecimento sobre um domínio e podendo servir como modelo conceitual para o desenvolvimento de Sistemas de Informação. No entanto, sua popularidade tem favorecido o surgimento de várias ontologias que descrevem um mesmo domínio, resultando no uso de diferentes entidades para representar o mesmo objeto no mundo real. Neste cenário, surgem problemas de ambiguidade e heterogeneidade semântica, dificultando o compartilhamento de informações. O alinhamento de ontologias estabelece correspondências entre entidades de ontologias distintas que representam o mesmo objeto do domínio, e vem sendo aplicado com sucesso na literatura para resolver os problemas mencionados. Várias métricas de similaridade têm sido consideradas para indicar a potencial correspondência entre entidades de duas ontologias, e alguns trabalhos evidenciam quando grupos de métricas são combinados. Entretanto, há cenários com resultados ainda não satisfatórios, demandando novas abordagens. Por outro lado, Mineração de Dados tem sido utilizada para descobrir automaticamente modelos que explicitem conhecimento implícito em dados. Neste trabalho, mineração de dados é aplicada a uma base de dados históricos de alinhamentos de ontologias para descobrir um modelo que combine quatro grupos diferentes de métricas de similaridade. A combinação dos quatro grupos de métricas é um diferencial da abordagem, que foi avaliada através de um experimento considerando três domínios diferentes. Os resultados mostram o potencial do trabalho com a melhora em precisão e cobertura dos alinhamentos quando comparados com o estado da arte.

Palavras-chave: Alinhamento de Ontologia, Ontologia, Aprendizado de Máquina, Mineração de dados.

ABSTRACT

Ontologies have attracted the attention of the scientific community, especially in the field of Semantic Web. An ontology describes a domain of knowledge in terms of its entities (concepts, relationships and instances), thus allowing the sharing of knowledge about a domain and can serve as a conceptual model for the development of Information Systems. However, the popularity of the subject has favored the emergence of several ontologies that describe the same domain, resulting in the use of different entities to represent the same object in the real world. In this scenario, there are problems of ambiguity and semantic heterogeneity that make information sharing difficult. The alignment of ontologies establishes correspondences between entities of different ontologies that represent the same object in the domain and has been successfully applied in the literature to solve previously mentioned problems. Several similarity metrics have been considered to indicate the potential correspondence between entities of two ontologies, and some studies show what happens when metric groups are combined. However, there are still not satisfactory scenarios results, requiring new approaches. On the other hand, Data Mining has been used to automatically discover models that explain implicit knowledge in data. In this master thesis, data mining is applied to historical database of alignments of ontologies to discover a model that combines four different groups of similarity metrics. The combination of the four groups of metrics is a novel approach, which was evaluated through an experiment considering three different domains. The results show the potential of the work, with the improvement in precision and recall of alignment when compared to state of the art.

Keywords: Ontology Alignment, Ontology, Machine Learning, Data Mining.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas

API	*Application Programming Interface*
CSV	*Comma-Separated Values*
GUI	*Graphical User Interface*
IA	Inteligência Artificial
KDD	*Knowledge Discovery in Databases*
MD	Mineração de Dados
ML	*Machine Learning*
MLP	*Multi Layer Perceptron*
OAEI	*Ontology Alignment Evaluation Initiative*
OWL	*Web Ontology Language*
PCA	Análise de Componentes Principais
PLN	Processamento de Linguagem Natural
RDF	*Resource Description Framework*
RDFS	*Resource Description Framework Schema*
RF	*Random Forest*
RNA	Rede Neural Artificial
RI	Recuperação de Informação
SVM	*Support Vector Machine*
URI	Identificador Uniforme de Recursos (*Uniform Resource Identifier*)
URL	*Uniform Resource Locator* (Localizador-Padrão de Recursos)
XML	*Extensible Markup Language*
W3C	*World Wide Web Consortium*

Capítulo 1 – Introdução

Este capítulo fornece uma visão geral da dissertação, bem como a motivação para tratar o alinhamento de ontologias. A proposta de solução para o problema exposto é brevemente descrita, apresentando as linhas de ação que norteiam esta dissertação. O objetivo, a hipótese e a metodologia usada para testá-la também são apresentadas.

1.1. Motivação e Objetivo

Ontologia é uma especificação formal explícita de uma conceitualização de um domínio de interesse [28] [29]. O significado das definições adicionado do termo "compartilhada" são explicados em Studer *et al.* [70]: O termo especificação significa que uma ontologia representa o conhecimento sobre um determinado domínio de interesse; formal refere-se ao fato que a ontologia deve ser processável por máquina; explícita significa que os tipos de conceitos usados e as restrições ao seu uso são explicitamente definidos; a conceitualização refere-se a um modelo abstrato de algum fenômeno do mundo por ter identificado os conceitos relevantes desse fenômeno; finalmente, compartilhada implica que uma ontologia reflete um acordo sobre a conceituação de domínio entre as pessoas em uma comunidade, reflete a noção de que uma ontologia captura conhecimento consensual, isto é, não é privado para alguns indivíduos, mas consenso de um grupo.

Quando consideradas sob a perspectiva de um artefato de engenharia e dentro de um domínio de conhecimento, ontologias consistem de uma estrutura formal de conceitos

e relações entre conceitos do domínio, e de um conjunto de axiomas que restringe a interpretação desta estrutura e permite a derivação de conhecimento novo a partir do conhecimento factual representado na estrutura [27]. Formalmente, uma ontologia O é uma tupla O = <C, R, I>, onde C é o conjunto de conceitos, que povoam o domínio de interesse, R o conjunto de relações entre os conceitos e I é o conjunto de instâncias que

instanciam os conceitos e relações e representam objetos do mundo real. Um elemento do conjunto {C ∪ R ∪ I} é denominado *entidade* [16].

Ontologias têm sido bastante utilizadas em diversas aplicações e na modelagem conceitual de domínio específico, porém, a criação dessas ontologias é comumente realizada de acordo com necessidades locais e muitas vezes sem a preocupação de reutilização, às vezes é simplesmente porque a comunidade difere de outra e portanto as entidades podem ser representadas de forma diferente. Desta forma, diferentes ontologias são criadas, contendo nomes diferentes para um mesmo conceito ou conceitos de diferente granularidade ou categorizações. Neste contexto, surge o problema da heterogeneidade semântica, que vem sendo investigado em cenários de integração de dados [13] [19], interoperabilidade entre aplicações [9] [52] [15] [67] e web semântica [4] [47] [55] [14].

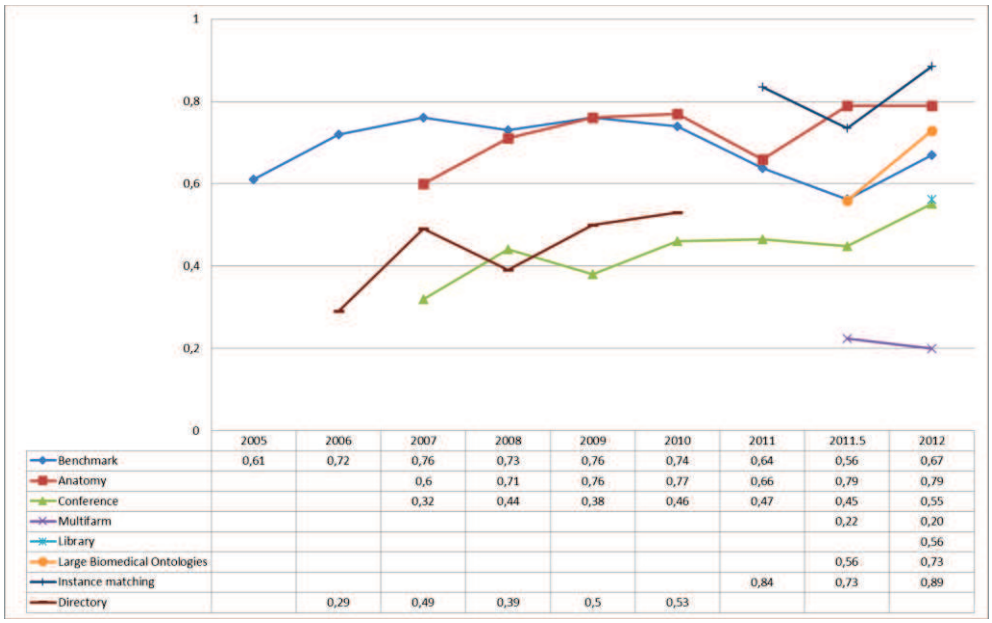
Por exemplo, considerando um conjunto de três ontologias descrevendo o domínio de organização de conferências, as pessoas presentes à conferência podem ter nomes diferentes, como *Participant*, *Conference_Participant* ou *Attendee*, e até mesmo conceituações diferentes, se por exemplo, *Participant* ou *Conference_Participant* incluem os palestrantes enquanto *Attendee* não. A heterogeneidade das ontologias provoca principalmente o problema de ambiguidade na interpretação de entidades e, consequentemente, impede o compartilhamento de conhecimento do domínio. Portanto, o alinhamento de ontologias torna-se uma tarefa crucial para essas aplicações.

O alinhamento entre duas ou mais ontologias é feito geralmente através da determinação da similaridade entre pares de entidades das ontologias sendo alinhadas. Essa similaridade é definida por métricas, que indicam a força da similaridade entre as duas entidades em questão. Em geral a força é definida como um valor entre 0 e 1, onde quanto mais próximo de 1 mais similares as entidades são. Quando a força supera um determinado limiar, as duas entidades são ditas correspondentes entre si. Para ilustrar

vamos considerar a métrica baseada em *strings MongeElkan* [50] e a baseada em linguagem *Soundex* [54], para resolver o problema da heterogeneidade entre as entidades: *Participant* e *Attendee*. A primeira a baseada em *string*, compara os caracteres das entidades (*strings*) e estabelece a similaridade entre elas pela similaridade máxima entre os caracteres. A ordem da sequência de caracteres não é importante. Já a segunda usa conceitos fonéticos e determina a similaridade entre as entidades utilizando aspectos relacionados à sonoridade. Se dois elementos tem sons parecidos, eles são considerados similares, independente da grafia. Utilizando essas duas métricas para verificar a similaridade de *Participant* e *Attendee*, os valores de força retornados pelas métricas foram: *MongeElkan* = 0.27 e *Soundex* = 0.5. Considerando como limiar o valor de 0.8 as duas palavras não seriam alinhadas por nenhuma das duas métricas. Entretanto sabemos que as duas palavras são semelhantes. A combinação entre as métricas pode ser uma solução para encontrar resultados melhores.

Alguns trabalhos recentes no campo de alinhamento de ontologias consideram a combinação de métricas de similaridade: GOMMA [41], LogMap [13, 62], CODI [35], [53], YAM ++ [16] e COMA [48], alguns como YAM ++, GOMMA e [32] empregam Mineração de Dados nos seus trabalhos. No entanto, a seleção das métricas de similaridade apropriadas, bem como o ajuste de configuração de sua combinação, são ainda problemas em aberto e relevantes. De fato, diferentes cenários de alinhamento podem exigir diferentes conjuntos de métricas de similaridade, o que, consequentemente, gera a necessidade de configurações diferentes em função da combinação das métricas. No entanto, a avaliação destas abordagens mostra que o problema ainda não se encontra plenamente resolvido, como mostra a Figura 1, com os resultados das médias nas avaliações da Medida-F entre as ferramentas participantes nos anos de 2005 e 2012, Medida-F é a média harmônica entre precisão e cobertura. As avaliações foram realizadas

nos domínios (*Benchmark, Anatomy, Conference, MultiFarm, Library, Large Biomedical Ontologies, Instance Matching e Directory*) da OAEI (*Ontology Alignment Evaluation Initiative*). A OAEI é uma iniciativa internacional coordenada, cujo objetivo é avaliar os pontos fortes e fracos de ferramentas de alinhamento, comparar o desempenho de técnicas e melhorar as técnicas de avaliação. Ela vem organizando campanhas de avaliação desde 2004 com o objetivo de avaliar as tecnologias de alinhamento de ontologia.



	2005	2006	2007	2008	2009	2010	2011	2011.5	2012
Benchmark	0,61	0,72	0,76	0,73	0,76	0,74	0,64	0,56	0,67
Anatomy			0,6	0,71	0,76	0,77	0,66	0,79	0,79
Conference			0,32	0,44	0,38	0,46	0,47	0,45	0,55
Multifarm								0,22	0,20
Library									0,56
Large Biomedical Ontologies								0,56	0,73
Instance matching							0,84	0,73	0,89
Directory		0,29	0,49	0,39	0,5	0,53			

Figura 1 - Domínio X Medida-F Média/Ano.

Desta forma, este trabalho propõe uma avaliação experimental da combinação de métricas de similaridade para o Alinhamento de Ontologias através de Mineração de Dados. Com o objetivo de gerar um modelo classificador de alinhamento de ontologias que oferece a possibilidade de combinar diferentes grupos de métricas de similaridade automaticamente. No presente trabalho, os algoritmos de Mineração de Dados supervisionados são utilizados para extrair o modelo ideal para combinação dos diferentes grupos de métricas de similaridade. Assim, o problema de alinhamento é transformado em uma tarefa Aprendizado de Máquina supervisionado. Um diferencial da abordagem foi considerar grupos diferentes de métricas quando comparado com o estado da arte.

Diante de tudo que foi apresentado, a hipótese foi formulada da seguinte forma: SE forem aplicado algoritmos de Aprendizado de Máquina para aprender a combinação de métricas similaridade para o alinhamento de ontologias ENTÃO os valores de força dos alinhamentos de ontologias serão melhores em termos de precisão e cobertura.

O objetivo da pesquisa é encontrar uma forma de melhorar a precisão e cobertura dos alinhamentos de ontologias de diferentes domínios. Com uma abordagem baseada em técnicas de Aprendizado de Máquina para gerar um modelo classificador de alinhamento de ontologias, tendo como base de dados os alinhamentos encontrados através de diferentes grupos de métricas de similaridade. O objetivo principal da pesquisa é geração de um classificador que seja capaz de indicar se existe uma correta correspondência entre entidades de 2 (duas) ontologias.

1.2. Metodologia de pesquisa

O primeiro passo desta pesquisa foi elaborar a revisão bibliográfica e obter o conhecimento atualizado ("estado da arte") nas áreas de ontologia, alinhamento de ontologia e das métricas de similaridades utilizadas. Foram criadas fichas de leitura para os trabalhos correlatos. A leitura crítica sobre estes trabalhos gerou uma lista de questionamentos, que fundamentaram e diferenciou a presente proposta das demais. A validação da proposta foi realizada através de experimentos comparativos com outras propostas existentes que são o estado da arte em alinhamento de ontologias. Os critérios para a escolhas dos trabalhos correlatos foram; utilizar preferencialmente a combinação de diferentes grupos de métricas de similaridade; os domínios da OAEI 2012 e Mineração de Dados para realizar o alinhamento de ontologias. Para então realizar um estudo comparativo entre as ferramentas escolhidas e a abordagem proposta, e verificar se a

combinação dos grupos de métricas de similaridade escolhidos obtém melhores resultados.

Para geração dos dados que serviram de entrada para a geração do modelo, foi desenvolvido um protótipo de sistema de informação, para apoiar a execução dos experimentos.

1.3. Organização da dissertação

Com o propósito de orientar o leitor sobre a distribuição dos assuntos, os capítulos dessa dissertação foram organizados da seguinte forma: a fundamentação teórica da proposta é apresentada na Capitulo 2. A proposta e a caracterização da contribuição são apresentadas no Capítulo 3. No Capítulo 4 são apresentados o planejamento dos experimentos, as bases de dados utilizadas, os resultados dos experimentos, bem como a avaliação e a conclusão sobre estes resultados. São discutidos alguns trabalhos relacionados com a proposta no Capítulo 5. Por fim, são apresentadas algumas contribuições da proposta, trabalhos futuros, limitações e a conclusão no Capítulo 6.

Capítulo 2 – Fundamentação Teórica

Neste capítulo serão apresentadas algumas definições basilares para a compreensão deste trabalho, como a definição de ontologia e alinhamento de ontologia, Descoberta de Conhecimento em Bases de Dados, e as formas de avaliação dos alinhamentos.

2.1. Ontologia

Ontologias modelam um domínio de conhecimento descrevendo os conceitos desse domínio. É comum que uma ontologia represente a taxonomia de um domínio, que é representada por classes de objetos e relações entre eles [50]. O formalismo adotado por elas permite representar o conhecimento sem ambiguidades. Ontologias, sob a perspectiva computacional, são modelos de referência concebidos para representar conceitos (conhecimento sobre o mundo) de forma padronizada e consistente, usando vocabulário comum e consensual [29]. Tradicionalmente, a modelagem do domínio de conhecimento é feita por especialistas de domínio e engenheiros de conhecimento que usam métodos, técnicas e linguagens da engenharia de ontologias para a definição de conceitos e relações entre estes conceitos do modelo. Na literatura há várias definições para ontologias. Neste trabalho adotamos a seguinte:

Definição 1. Uma ontologia é uma tupla: O = <C, R, I>,

Onde C, é um conjuntos de conceitos (classe), que povoam o domínio de interesse, R é o conjunto de relações (papeis) entre os conceitos do domínio; I é o conjunto de instâncias do conceito (indivíduos) que representam objetos do mundo real.

Neste trabalho quando menciona-se as “entidades” de uma ontologia [8], fala-se das classes, relações e dos indivíduos. A Figura 2 mostra um exemplo de ontologia que modela o domínio de transporte com os conceitos e seus relacionamentos. Como exemplo considere o conceito *Carro Grande* que possui um relacionamento “é-um” com o conceito *Carro.* Além disso o conceito *Locomotiva* tem uma relação de “possui” com o conceito *Cavalo-Vapor*.

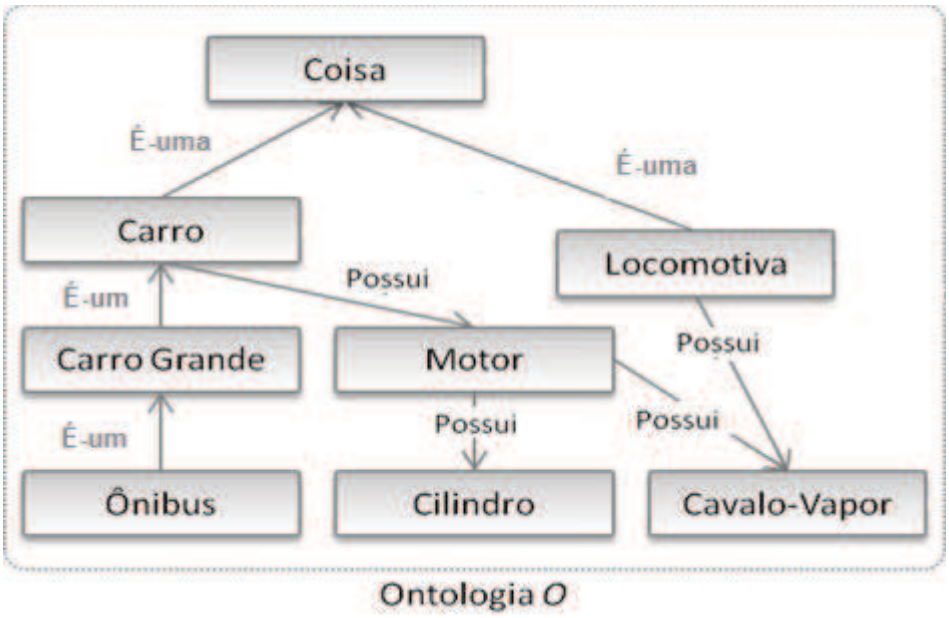


Figura 2 - Ontologia do domínio de transporte

As ontologias são descritas por linguagens padronizadas de metadados. Para se construir uma ontologia é necessário decidir qual linguagem melhor adequa-se à necessidade da modelagem, tendo em vista que, quanto maior a expressividade da linguagem, maior a riqueza para representação dos conceitos, porém, menor a decidibilidade e tratabilidade computacional.

Nesse âmbito, o XML[1] (*Extensible Markup Language*) oferece uma metalinguagem de marcação para a criação estruturada de documentos, por intermédio de marcadores

[1] http://www.w3.org/XML/

(*tags*). A criação de *tags* é livre e deve obedecer a um conjunto de regras sintáticas. Entretanto, a liberdade dos usuários para a criação de *tags* no XML pode gerar ambiguidades para a definição dos conceitos. Para eliminar esta ambiguidade, linguagens foram definidas e padronizados pela W3C (*WorldWide Web Consortium*), como, por exemplo, RDF (*ResourceDescription Framework*) [2] , RDFS (*ResourceDescription Framework Schema*)[3] e OWL (*Web Ontology Language*)[4]. Dessas linguagens, a OWL tem maior representatividade e permite a criação de esquemas de classes, hierarquias, propriedades, relações, restrições e padronizações terminológicas.

2.2. Alinhamento de Ontologia

Em sistemas abertos e distribuídos, como a web semântica, a heterogeneidade dos dados não pode ser evitada. Diferentes pessoas têm interesses e hábitos diferentes, utilizam ferramentas diferentes e possuem conhecimento, na maioria das vezes, em diferentes níveis de detalhe. Estas várias razões levam a diversas formas de heterogeneidade e, portanto, devem ser cuidadosamente levadas em consideração.

O objetivo do alinhamento de duas ou mais ontologias é reduzir a heterogeneidade existente entre elas. A heterogeneidade não reside exclusivamente nas diferenças entre os objetivos das aplicações de acordo com o fim para o qual foram concebidas ou nos formalismos expressos nas ontologias na qual foram codificadas. O alinhamento de ontologias ocorre no sentido de identificar as relações entre entidades individuais de múltiplas ontologias, e é uma condição necessária para estabelecer a interoperabilidade entre elas [17].

---

[2] http://www.w3.org/RDF/
[3] http://www.w3.org/TR/rdf-schema/
[4] http://www.w3.org/TR/owl-features/

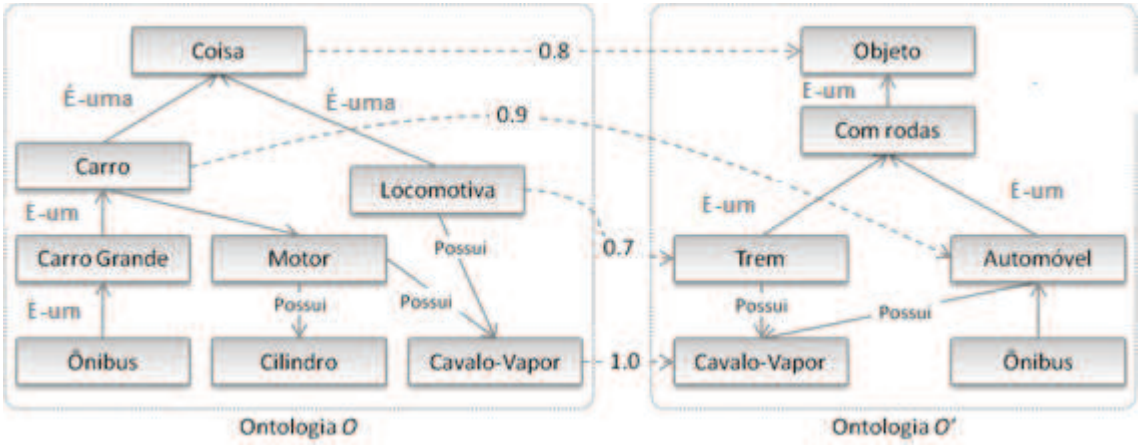


Figura 3 - Exemplo de alinhamento de duas ontologias.
Fonte: Adaptado de [1]

Sejam O, O' duas ontologias ilustradas na Figura 3. Alinhar as ontologias O com O' é encontrar um conjunto de correspondências A = {A1, A2, .., An}, que são representadas pelas linhas pontilhadas. Cada Ai ($1 \leq i \leq n$) é uma 5-tupla: <id,e1,e2,u,v> onde id representa o identificador único, e1 é uma entidade em O (ex. Coisa), e2 entidade é uma em O' (ex. Objeto), u é uma relação de equivalência, subsunção ou disjunção, entre e1 e e2, e v é a força de correspondência entre e1 e e2. A força é usualmente no intervalo [0,1], atribuído através de uma métrica que mede a similaridade entre duas entidades das ontologias [62]. No exemplo das entidades Coisa e Objeto a força que denota a confiança da similaridade do alinhamento entre as entidades é de 0.8.

2.2.1. Classificação das Técnicas de alinhamento de ontologia

Técnicas de alinhamento de ontologia, são algoritmos que recebem como entrada duas ontologias, geram correspondências entre as mesmas. Em [21][63], são apresentadas duas classificações sintáticas de técnicas de alinhamento com base no que os autores consideram ser as características mais importantes dessas técnicas. Estas duas classificações são apresentadas como duas árvores que compartilham suas folhas. As

folhas representam classes de técnicas de alinhamento. As duas classificações sintáticas são:

I. – **Por Granularidade e/ou Interpretação de entrada** - A classificação baseia-se (i) na forma como as ferramentas realizam o alinhamento (*matches)*, ou seja, no nível de elemento ou nível de estrutura, e em seguida, (ii) sobre como as técnicas geralmente interpretam as informações de entrada: terminológicas, estrutura interna, estrutura externa, extensional e semântica.

Nível de elemento vs. nível de estrutura – Euzenat e Shvaiko [21] separam as técnicas em duas vertentes: técnicas no nível de elemento e técnicas no nível de estrutura, as técnicas no nível de elemento consideram as entidades da ontologia de forma isolada, sem analisar seus relacionamentos com as demais entidades, já as técnicas do nível de estrutura ao contrário das técnicas do nível de elemento, comparam as entidades das ontologias a serem alinhadas considerando seus relacionamentos com as demais entidades.

II. – **Por tipo de classificação de entrada**- É baseado no tipo de entrada que é usada por meio de técnicas de alinhamento elementares que estão descritas a seguir:

Terminológica – compara *Labels* de entidades. Incluem técnicas de processamento de *String* (edição de distância, prefixo/sufixo e *N-Gram*), técnicas baseadas em linguagem (*tokenization, stemming e stopwords*) [39] ou em recursos linguísticos, como *thesaurus* e dicionários taxonômicos.

Estrutura Interna – compara evidências da estrutura interna de entidades (domínio, *range*, cardinalidade, etc.).

Estrutura Externa – compara as estruturas de relacionamento entre entidades, como a árvore taxonômica e o grafo cíclico formado por outras relações. Neste item, algoritmos para grafos são comuns.

Extensional – compara extensões das entidades. Extensões são entidades que formalmente, a princípio, não precisam estar declaradas nas ontologias. Exemplos são comentários e instâncias.

Semântica – compara a interpretação (modelo lógico) das entidades. Inclui técnicas de inferência lógica / simbólica sobre axiomas, como Lógica de Descrição [3].

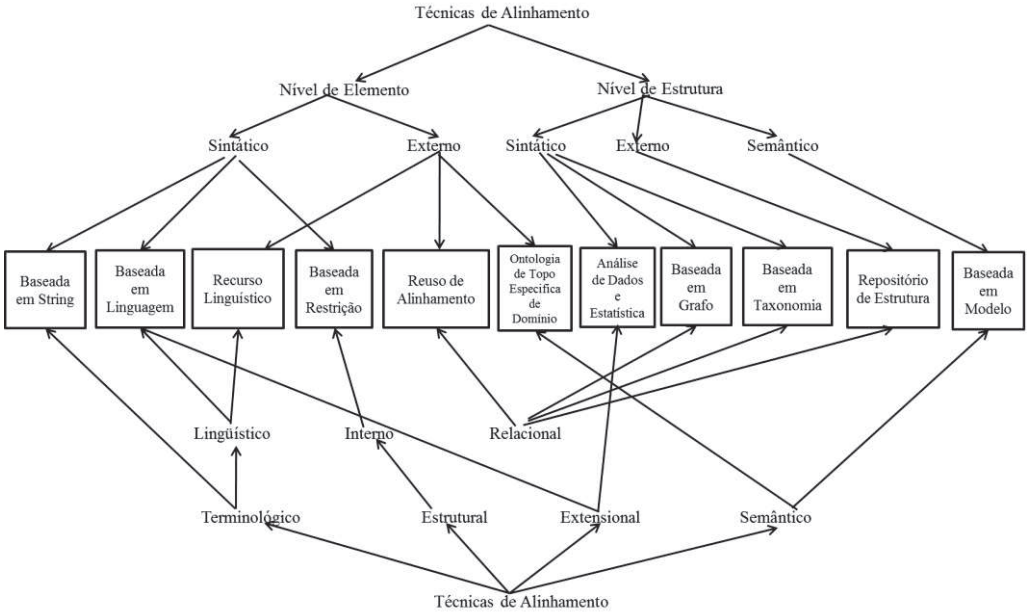


Figura 4 - Técnicas básicas de alinhamento de ontologias
Fonte Adaptado: [21]

A classificação global da Figura 4 pode ser lida tanto em ordem decrescente (com foco em como as técnicas interpretaram as informações de entrada) quanto ascendente (focando os tipos de objetos manipulados), a fim de alcançar a camada de técnicas básicas. Como [18] [21], classificamos as abordagens de mapeamento em função dos grupos de

métricas de similaridades em: baseadas em *Strings*, baseadas em linguagem, baseadas em recursos linguísticos, baseadas em restrição, baseadas em reuso de alinhamento, baseadas em ontologia de topo, baseadas em análise de dados e estatística, baseadas em grafo, baseadas em taxonomia, baseadas em repositório de estrutura e baseadas em modelo. A seguir serão explicadas cada uma dessas técnicas.

- **Técnicas Baseadas em *String* -** Comparam e associam nomes e *Labels* das entidades das ontologias. Essas técnicas se aproveitam da estrutura da *String*, como distancia medida pelas mudanças necessárias na *String* para chegar a outra. Divide partes da *String* para realizar as comparações. Exemplos de métricas de similaridade pertencentes a esse grupo são:

  - ***Jaro*** **[**36**]** - Introduz um comparador de *String* que responde por inserções, exclusões e transposições. O algoritmo básico Jaro tem três componentes: (1) calcular os comprimentos de *Strings*, (2) encontrar o número de caracteres comuns nas duas S*trings* e (3) encontrar o número de transposições (alteração na ordem das palavras). A métrica de distância Jaro leva em consideração os desvios ortográficos típicos.

$$jaro(s,t) = \frac{1}{3}.\left(\frac{|s'|}{|s|}\right) + \frac{|t'|}{|t|} + \frac{|s'| - T_{s',t\prime}}{2|s'|}$$

    Resumidamente, para duas *Strings* s e t, onde s' são os caracteres em s que são comum em t, e que t' são os caracteres em t que são comum em s. Um caractere ‘a’ em s é comum em t se o mesmo caractere ‘a’ está presente no mesmo lugar em t.

    Para o exemplo dos nomes MARTHA e MARHTA aplicando a formula:

$$jaro(s,t) = \frac{1}{3}.\left(\frac{6}{6} + \frac{6}{6} + \frac{6-2}{12}\right) = \frac{1}{3}.(1.0 + 1.0 + 0.33) = 0.77$$

• ***Levenshtein*** [44] - A distância de *Levenshtein* é uma métrica para medir as diferenças entre duas cadeias de caracteres. Informalmente, a distância entre duas palavras para a métrica de *Levenshtein* é o número mínimo de edições de um único caractere (inserção, deleção e substituição) necessária para transformar uma palavra na outra. Matematicamente, distância de *Levenshtein* entre duas *Strings* a, b é dada por $lev_{a,b}(|a|,|b|)$ onde:

$$\operatorname{lev}_{a,b}(i,j) = \begin{cases} \max(i,j) & \text{if } \min(i,j) = 0, \\ \min \begin{cases} \operatorname{lev}_{a,b}(i-1,j) + 1 \\ \operatorname{lev}_{a,b}(i,j-1) + 1 \\ \operatorname{lev}_{a,b}(i-1,j-1) + [a_i \neq b_j] \end{cases} & \text{otherwise.} \end{cases}$$

Por exemplo, a distância Levenshtein entre as palavras inglesas "*kitten*" (gato) e "*sitting*" (sentando-se) é 3, já que com apenas 3 edições conseguimos transformar uma palavra na outra, e não há maneira de o fazer com menos de três edições:

1. *kitten*
2. *sitten* (substituição de 'k' por 's')
3. *sittin* (substituição de 'e' por 'i')
4. *sitting* (inserção de 'g' no final)

- ***SmithWaterman*** [68] - Determina as regiões semelhantes entre duas sequências de caracteres. Em vez de olhar para a sequência total, compara segmentos de todos os comprimentos possíveis e otimiza a medida de similaridade.

  Uma matriz H é construída como se segue:

  $H(i,0) = 0, 0 \leq i \leq m$

  $H(0,j) = 0, 0 \leq j \leq n$

$$se\, w(a_i, b_j) = w(acerto). \quad ou\ se\ a_i \,!=\, b_j. \quad w(a_i, b_j) = w(erro)$$

$$H\ (i,j) = \max \begin{Bmatrix} 0 \\ H(i-1, j-1) + w(a_i, b_j)\ acerto/erro \\ H(i-1, j) + w(a_i, -)\ deleta \\ H(i, j-1) + w(-, b_j)\ insere \end{Bmatrix}\ ,1 \le i \le m, 1 \le j \le n$$

Onde:

a, b = cadeias de caracteres sobre o Alfabeto Σ

m = comprimento de (a)

n = comprimento de (b)

$H(i,j)$ - é o escore máximo de similaridade entre o sufixo de a[1...i] e o sufixo de b [1...j]

$w(c,d), c, d \in \sum \cup \{' - '\}$ ' - é o esquema de escore para lacunas (penalidades)

Exemplo:

Sequência 1 = ACACACTA

Sequência 2 = AGCACACA

w(gap) = 0 / (penalidade para a lacuna)

w(*correspondência*) = +2

$$H = \begin{bmatrix} & - & A & C & A & C & A & C & T & A \\ - & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A & 0 & 2 & 1 & 2 & 1 & 2 & 1 & 0 & 2 \\ G & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 \\ C & 0 & 0 & 3 & 2 & 3 & 2 & 3 & 2 & 1 \\ A & 0 & 2 & 2 & 5 & 4 & 5 & 4 & 3 & 4 \\ C & 0 & 1 & 4 & 4 & 7 & 6 & 7 & 6 & 5 \\ A & 0 & 2 & 3 & 6 & 6 & 9 & 8 & 7 & 8 \\ C & 0 & 1 & 4 & 5 & 8 & 8 & 11 & 10 & 9 \\ A & 0 & 2 & 3 & 6 & 7 & 10 & 10 & 10 & 12 \end{bmatrix}$$

Para obter o alinhamento ótimo, começamos com o maior valor na matriz (i,j). Então, nós vamos para trás para uma das posições (i-1,j), (i,j-1) ou (i-1,j-1), dependendo da direção de movimento usado para construir a matriz. Mantemos o processo até chegar a um célula da matriz com valor zero, ou o valor na posição (0,0). No exemplo, o valor mais alto corresponde à célula na posição (8,8). A caminhada de volta corresponde a (8,8), (7,7), (7,6), (6,5), (5,4), (4,3), (3,2), (2,1), (1,1), e (0,0). Uma vez que tenhamos terminado, reconstruímos o alinhamento da seguinte forma: começando com o último valor, chegamos a (i,j) usando o caminho previamente calculado. Um salto na diagonal implica que há um alinhamento (ou uma correspondência ou uma não correspondência). Um salto de cima para baixo implica que há uma deleção. Um salto da esquerda para a direita implica que há uma inserção.

- ***Jaccard*** [7] - Mede a sobreposição de dois conjuntos. É uma estatística utilizada para comparar a semelhança (desambiguação necessária) e diversidade de conjuntos de amostras. O coeficiente de *Jaccard* mede a semelhança entre conjuntos de amostras e é definido como o tamanho da interseção dividida pelo tamanho da união dos conjuntos da amostra.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

Para o exemplo dos nomes MARTHA e MARHTA aplicando a formula:

$$J(A,B) = \frac{6}{6} = 1.0$$

- ***JaroWinkler*** [75] - É uma medida de similaridade entre duas *Strings*. É uma variação da métrica *Jaro*. Essa métrica estabelece que dadas duas *Strings* s1 e s2, sua distância dj é:

$$d_j = \begin{cases} 0, & se\ m = 0 \\ \frac{1}{3}, \left(\frac{m}{s1} + \frac{m}{s2} + \frac{m-t}{3m}\right) & caso\ contrário \end{cases}$$

Onde:

m é o número de correlações entre caracteres;

s1 e s2 são os tamanhos de s1 e s2, respectivamente;

t é o número de transposições (alteração na ordem das palavras).

Exemplo das *Strings* MARTHA e MARHTA:

m = 6, s1 =6 e s2 =6, e dois caracteres onde há a transposição. *Jaro-Winkler* calcula o valor de força da seguinte maneira:

$$d_j = \frac{1}{3}\left(\frac{6}{6} + \frac{6}{6} + \frac{6-2}{18}\right) = 0.74$$

- ***MongeElkan*** [50] - É uma variante da distância de Smith-Waterman, com parâmetros de custo em particular (penalidades) e dimensionado para o intervalo [0,1]. Para a comparação das *Strings* é utilizado o Gotoh [53] - Algoritmo de programação dinâmica com base na distância para o alinhamento de pares onde diferença de comprimento k é penalizado em a+bk - a é penalidade do gap de abertura, b é o gap de continuação da penalidade.

$$Comparar\ (A, B) = \frac{1}{|A|}\sum_{i=1}^{|A|} max_{j+i}^{|B|}\ comparar(A_i, B_j).$$

Os custos são os seguintes: caracteres descasados = -3, correspondência = +5 (*case insensitive*), correspondência aproximada = 3, para correspondência em {dt} {gj} {lr} {mn} {bvp}} {aeiou}{,. }, gap inicial = +5, continuar gap = 1.

- ***TagLinkToken*** [7] - É uma métrica híbrida de *Strings*. Es*cores* de *token* são computadas pelo método baseado em caracteres. Pares simbólicos correspondentes são definidos e para o cálculo de distância das *Strings* é necessário um métrica baseada em caracteres.
- ***ChapmanLengthDeviation*** [7] **-** O desvio do comprimento das cadeias de entrada é usado para determinar se as cadeias são semelhantes em tamanho. Esta abordagem não se destina a ser utilizada sozinha, mas sim, juntamente com outras abordagens.

$$sim = \frac{A}{B}$$

Em que A é o tamanho da cadeia de caracteres da String 1 e B o tamanho da cadeia de caracteres da String 2. Para o exemplo dos nomes MARTHA e MARHTA a aplicação da formula retornaria 1 sim = 6/6 =1

- ***ChapmanMeanLength*** [7] - Fornece uma medida de semelhança entre duas cadeias de caracteres, esta abordagem pode ser usada para determinar quais métricas pode melhor se aplicar, ou seja, as cadeias de comprimentos semelhantes são mais semelhantes independentemente do conteúdo. A similaridade é calculada da seguinte forma: 500 menos a soma do comprimento total das duas cadeias de caracteres. Assim, fazendo o cálculo para duas cadeias de caracteres A e B *sim* = (500 – A+B) /500. O resultado é, então, multiplicado por ele mesmo 4 vezes *sim* = (sim*sim*sim*sim). Para o exemplo dos nomes MARTHA e MARHTA a aplicação da formula *sim* = (500 – 12) /500 = 0.976 continuando com *sim* = (0.976*0.976*0.976*0.976) = 0.907

- ***SmithWatermanGotoh*** [7] - implementa o *Smith Waterman* com a melhoria do ***Gotoh*** [25]

- ***SmithWatermanGotohWindowedAffine*** [7] - implementa o *Smith Waterman* com a melhoria do *Gotoh* [25] com o modelo de penalidade *gap affine*[5].

  O modelo *Affine gap* inclui variáveis de *gap* de custos, tipicamente baseadas no comprimento do intervalo l. Comparando duas sequências $A(= a_1, a_2, a_3 \dots a_n)$ e $B(b_1, b_2, b_3 \dots b_m)$

  $D_{ij} \max\{D_{i-1,j-1} + d(a_i, b_j), \max\{D_{i-kj} - W_i\}, \max\{D_{i,j-l} - Wl\}, 0\}$

  D_{ij} É a máxima similaridade para dois seguimentos terminando em a_i e b_j respectivamente.

  O custo *Affine gap* são especificados de duas formas: com custos para iniciar um gap, e por outro lado com um custo para continuar o *gap*.

- ***NeedlemanWunch*** [52] - O algoritmo calcula o alinhamento ótimo de similaridade global entre duas sequências. É utilizado o método de programação dinâmica e permite que modelos de pontuação arbitrárias (utilização de custos gerais de *gap*) para inserir e excluir operações. É semelhante a métrica *Levenshtein,* o que adiciona ajuste do custo de *gap.* Como se segue:

  $D(i-1, j-1) + d(si, tj) - Substituição/copia$

  $D(i,j) = min\, D(i-1, j) + G - Inserção$

  $D(i, j-1) + G - Deleção$

  Onde G = "custo gap" e d (i, j) é mais uma função de distância arbitrária em caracteres.

[5] *Affine* - Métrica de edição de distância. Atribuir um custo relativamente mais baixo para uma sequência de inserções ou exclusões.

- ***MatchingCoefficient*** [7] – É uma abordagem baseada em vetores muito simples, que simplesmente conta o número de termos, (dimensões), em que ambos os vetores não são zero. Assim, para o vetor do conjunto X e do conjunto Y o coeficiente correspondência é | X e Y |. Isto pode ser visto como contagem de base em vetores de termos relacionados [71].
- ***CosineSimilarity*** [7] **-** É uma medida da similaridade entre dois vetores de um espaço com um produto interno que mede o cosseno do ângulo entre elas. O cosseno 0° é 1, e é inferior a 1 para qualquer outro ângulo. É, portanto, um julgamento de orientação e não de magnitude: dois vetores com a mesma orientação têm uma similaridade de um cosseno, dois vetores de 90° tem uma similaridade de 0, e dois vetores diametralmente opostos têm uma similaridade de -1, independente da sua magnitude. A similaridade cosseno é particularmente utilizada no espaço positivo, em que o resultado é perfeitamente delimitado em [0,1].

Técnicas Baseadas em Linguagem - Consideram nomes como palavras em uma linguagem natural, como português, espanhol ou inglês. São baseadas em técnicas de Processamento de Linguagem Natural (PLN) [10], explorando propriedades morfológicas das palavras que são fornecidas como entrada. Exemplos de métricas de similaridade pertencentes a esse grupo são:

- ***Soundex*** [54] - É um algoritmo fonético para indexação de nomes pelo som, como pronunciado em Inglês. O objetivo é que homófonos[6] a sejam codificados na mesma representação, de modo que eles possam ser alinhados,

[6] As palavras homófonas são palavras de pronúncias iguais.

apesar de pequenas diferenças quando são soletradas. A codificação do *Soundex* para um nome composto por uma letra seguida de três dígitos numéricos: a primeira letra do nome e os dígitos que são codificados com as consoantes restantes. Consoantes de som similares partilham os mesmo dígitos. Segue a codificação do algoritmo: Manter a primeira letra do nome e remover todas as outras ocorrências de a, e, i, o, u, y, h, w.

Substitua consoantes com dígitos como segue (após a primeira letra):

b, f, p, v => 1
c, g, j, k, q, s, x, z => 2
d, t => 3
l => 4
m, n => 5
r => 6

Até que o tamanho seja igual a quatro. Caso não seja possível atingir o tamanho de quatro dígitos, são inseridos zeros (0) até que o tamanho seja atingido.

Para o exemplo dos nomes MARTHA e MARHTA os códigos seriam:

M~~A~~RT~~HA~~ = M630

M~~A~~R~~H~~T~~A~~ = M630

Que retornaria uma similaridade idêntica.

- ***ChapmanOrderedNameCompoundSimilarity*** [7] - O algoritmo de Similaridade pelo qual os termos são combinados e testados contra o algoritmo S*oundex* padrão, este se destina a fornecer uma melhor classificação para as listas de nomes próprios.
- ***ChapmanMatchingSoundex*** [7] – O algoritmo em que os termos são combinados e testados contra o algoritmo *soundex* padrão, este se destina a fornecer uma melhor classificação para as listas de nomes próprios.

- ***QGramsDistance*** [42] - Compara os elementos contando o número de ocorrências de diferentes q-grams. A cadeia é dividida em tokens com comprimento igual a 2. Um q-gram neste contexto refere-se a uma sequência de letras de uma determinada palavra. As *Strings* serão parecidas a medida que tenham mais q-grams em comum. Para o exemplo dos nomes MARTHA e MARHTA de 5 q-grams apenas 2 são iguais.

{MA, AR, RT, TH, HA}

{MA, AR, RH, HT, TA}

Técnicas Baseadas em Restrições - Utilizam algoritmos que lidam com as restrições internas aplicadas às definições das entidades, como tipo de dado e cardinalidade de atributos, entre outras [21].

Técnicas Baseadas em Recursos Linguísticos - Usam recursos externos como tesauros específicos do domínio. Exemplos de recursos linguísticos pertencentes a esse grupo são: *Wordnet*[7] [56] é um banco de dados léxico para o idioma inglês e UMLS *Unified Medical Language System,* nesses recursos são utilizados sinônimos, generalizações e especializações.

Técnicas Baseadas em Reuso de Alinhamentos – É a forma alternativa de explorar recursos externos, que armazenam resultados de alinhamento anteriores entre ontologias. Por exemplo [48], [39], [38]: Têm-se armazenados os resultados do alinhamento entre p e p', e entre p e p'', podemos reutilizar estes resultados no alinhamento de p' e p''.

[7] WordNet - http://wordnet.princeton.edu/ - WordNet ® é um grande banco de dados léxico do Inglês. Substantivos, verbos, adjetivos e advérbios são agrupados em conjuntos de sinônimos cognitivos (*synsets*), cada um expressando um conceito distinto. *Synsets* estão interligadas por meio de relações conceituais-semânticas e léxicas.

Técnicas Baseadas em Ontologias de Topo – Descrevem conceitos gerais, como: espaço, tempo, matéria, objeto, evento, ação, etc. As ontologias de Topo contém um glossário central que permite descrever termos em vários domínios, que contém noções gerais independentes para o problema ou domínio [26]. Podem ser utilizadas como fontes externas de conhecimento comum. A principal característica destas ontologias é que elas são baseadas em lógica, e deste modo, técnicas que exploram estes recursos são baseados em semântica.

Técnica Baseadas em Ontologias formais Específicas do Domínio - Também podem ser usadas como recursos externos, pois usam termos com sentidos relevantes apenas para este domínio, e que não estão relacionados a conceitos similares em outros domínios [61].

Técnica Baseadas em grafo – Aqui são aplicadas as técnicas que consideram as ontologias como grafos rotulados. A análise de similaridade entre um par de nós representando os conceitos nas duas ontologias baseia-se em suas posições dentro do grafo, considerando-se que, se dois nós de duas ontologias são similares, então seus nós adjacentes, também devem apresentar certo grau de similaridade [16], [55].

Técnicas Baseadas em Taxonomia - Também consiste em técnicas baseadas em grafo, que levam em consideração apenas a relação de especialização. Da mesma forma que as técnicas baseadas em grafo, se uma ligação do tipo é-um (*is-a*) conecta termos já considerados similares, é possível que seus nós adjacentes também sejam similares [20] [60].

Técnicas Baseadas em Repositórios de estruturas - Repositórios que controla as versões de ontologias e entidades de interesse, bem como os diferentes tipos de

mapeamentos. Essa técnica é utilizada para reduzir o tempo de execução durante a tarefa de alinhamento e reduzir a utilização dos recursos de memória [41] [25].

Técnicas Baseadas em Modelo - Algoritmos baseados em modelo (ou semanticamente fundamentados) manipulam entradas baseando-se na sua interpretação semântica. A idéia por trás deste tipo de técnica é que, se duas entidades são similares, elas compartilham a mesma interpretação lógica. Técnicas de inferência de lógica descritiva são exemplos de técnicas baseadas em modelo [16] [21] [38] [55] [35].

Técnicas Baseadas em Estatísticas e Análise de Dados - Técnicas que aproveitam amostras (preferencialmente grandes) de uma população com o objetivo de encontrar regularidades e discrepâncias. Isto ajuda a agrupar itens ou computar distâncias entre eles. Análise de correspondência e distribuições de frequência são exemplos destes tipos de técnicas [64]. Exemplos de métricas de similaridade pertencentes a esse grupo são:

- ***OverlapCoefficient*** [7] - O coeficiente de sobreposição é uma medida de similaridade relacionada com o coeficiente de *Jaccard*, que calcula a sobreposição entre os dois conjuntos de caracteres. Se o conjunto X é um subconjunto de Y ou o inverso, então o coeficiente de sobreposição é igual a um. Segue sua definição:

$$overlap(x, y) = \frac{|X \cap Y|}{\min(|X|, |Y|)}$$

- ***DiceSimilarity*** [7] - É uma medida estatística utilizada para comparar a semelhança de duas amostras. Uma medida em que a similaridade é definida como duas vezes o número de termos comuns nas entidades comparadas, dividido pelo número total de termos de ambas as entidades. Onde n_t é o

número de caracteres bigramas encontrados em ambas as cadeias, n_x é o número de bigramas na cadeia de x e n_y é o número de bigramas em cadeia y. Por exemplo, para calcular a semelhança entre: MARTHA e MARHTA

$$s = \frac{2n_t}{n_x + n_y}$$

Teríamos que encontrar o conjunto de bigramas para as *Strings*:

{MA, AR, RT, TH, HA}

{MA, AR, RH, HT, TA}

Como o conjunto tem 5 elementos, e na interseção desses dois conjuntos tem apenas dois elementos: MA e AR.

Calculando a formula: $s = \frac{2\ x\ 2}{5+5} = 0.4$

2.2.2. Avaliação do Alinhamento de Ontologia

A natureza dos conjuntos de dados que serão utilizados em um projeto de avaliação de alinhamento de ontologias deve atender dois requisitos: (i) cobertura de aspectos relevantes e (ii) imparcialidade da avaliação. Este conjunto de dados consiste tipicamente de pelo menos duas ontologias e um alinhamento esperado entre essas duas ontologias, chamado de alinhamento de referência (R) [17]. O alinhamento de referência é o alinhamento feito manualmente por um especialista no domínio que as ontologias representam [5]. O alinhamento obtido é utilizado como o padrão de referência para avaliar a qualidade do resultado determinado automaticamente pelas ferramentas de alinhamentos [37].

O critério mais comumente utilizado para a avaliação e compreensão dos alinhamentos de ontologia em relação as ferramentas de alinhamentos é a utilização do alinhamento de referência [19]. Medidas como Precisão, Cobertura e Medida-F têm sido

utilizadas para avaliar sistemas de alinhamento de ontologias [13] [32] e também como base nas campanhas OAEI para avaliação das ferramentas de alinhamento.

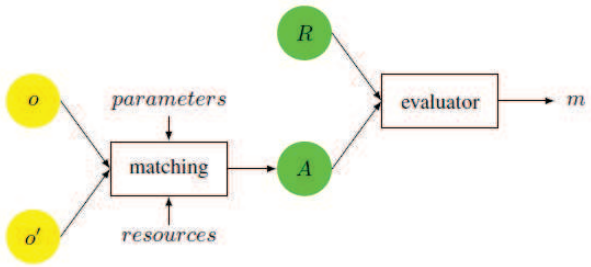


Figura 5 - Modelo básico para avaliação
Fonte: [19].

A Figura 5 ilustra uma ferramenta recebendo como entrada duas ontologias (O e O´) e gerando uma alinhamento A usando um conjunto de recursos e parâmetros. Um componente de avaliação recebe este alinhamento A e calcula a qualidade da medida m (tipicamente Precisão, Cobertura e Medida-F) comparando o alinhamento de referência R com todos os alinhamentos possíveis entre as duas ontologias de entrada.

2.3. Descoberta de Conhecimento em Base de Dados

Em sua busca pelo conhecimento, o homem desenvolve uma crescente capacidade de aumentar o conhecimento: conhecimento gerando conhecimento. Esse princípio norteia a Descoberta de Conhecimento em Banco de Dados (Knowledge Discovery in Databases (KDD)) [66].

A KDD visa buscar padrões desconhecidos existentes nas bases de dados e envolve diversas áreas de conhecimento, tais como a Estatística, Inteligência Artificial, Aprendizado de Máquina, Banco de Dados, Reconhecimento de Padrões, Armazenagem de Dados (Data Warehousing), Visualização de Dados entre outras [24]. Através da exploração das bases de dados, consideradas verdadeiros “depósitos” de conhecimento

em potencial, novos conceitos são extraídos gerando um novo conhecimento consistente, útil e compreensível, a ser incorporado ao conhecimento já existente. A Figura 6 mostra a sequência de passos, a fim de alcançar o conhecimento, que deve ser novo e útil. No entanto, para que o conhecimento extraído de bases de dados seja correto e represente informações verdadeiras capazes de gerar novos conhecimentos, a base de dados deve estar consistente.

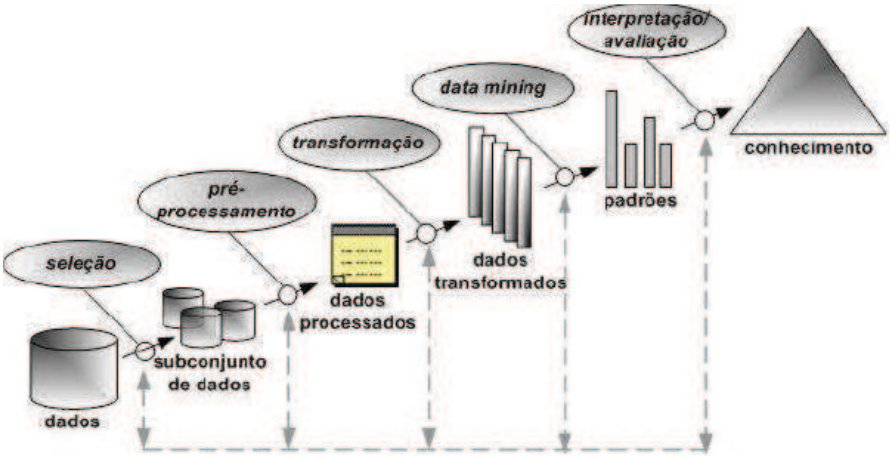


Figura 6 - Processo KDD proposto por Fayyad *et al.* (1996)

Para gerar um novo conhecimento alguns desses passos de KDD podem ser aplicado as bases de dados. Por exemplo, técnicas de pré-processamento, como a limpeza, tratamento de valores ausentes, a detecção de ruído dos dados e *outliers* podem ser aplicadas. Também é nessa etapa que os dados relevantes são coletados e atributos contínuos são normalizados, se necessário. Normalização evita que os atributos com grandes intervalos de valores sejam preteridos em detrimento de outros [31]. Outra etapa nesse grande processo chamado KDD é a aplicação de Mineração de Dados (MD), onde os algoritmos de Aprendizado de Máquina [49] são utilizados para aprender um modelo que reflete o conjunto de dados, explicitando padrões escondidos. Finalmente, o modelo aprendido passa através de um passo de pós-processamento, onde os padrões interessantes são filtrados, apresentados visualmente e interpretados. KDD é um processo iterativo [22].

Portanto, pode-se voltar ao passo anterior sempre que necessário. O processo é concluído quando o conhecimento for encontrado.

2.3.1. Pré-processamento

Esta etapa é de suma importância para o processo. A maioria das bases de dados, embora com grandes massas de informações, compreendem muitos valores incorretos ou imprecisos, causando grande "poluição" à base. A etapa de pré-processamento visa tratar os dados aumentando a qualidade dos mesmos, de forma a prepará-los adequadamente para a mineração, permitindo a extração de conhecimento consistente.

Segundo GOLDSCHMIDT [24], PANG-NING *et al.* [55] e ZHANG *et al.* [77] algumas das principais atividades desta fase são: redução da dimensionalidade, agregação, amostragem, discretização e binarização, construção de atributos, correção de prevalência, limpeza dos dados e transformação de atributo.

2.3.1.1. Redução de Dimensionalidade

De acordo com a compreensão do problema e dos objetivos em termos de conhecimento a serem alcançados pelo processo na base em questão, é necessária a realização da redução da dimensionalidade dos dados a serem considerados no processo de KDD, visto que o conjunto de dados selecionados é sempre um subconjunto dos dados existentes na base.

Quanto ao enfoque de redução da dimensionalidade dos dados, pode-se optar pela redução baseada em atributos, conhecida com redução de dados vertical (colunas), ou pela redução baseada em registros, chamada de redução de dados horizontal (tuplas). Atributos bem selecionados conduzem a modelos mais concisos e de maior precisão. E a eliminação de um atributo significa maior redução do que eliminar um registro, com isso

levando a redução do tempo computacional para a mineração. Bases com registros consistentes e coesos conduzem a um processo de melhores resultados [23]. Existem técnicas que podem ser aplicada em ambos os casos. No caso de redução de dados vertical as técnicas de *Filter* ou *Wrapper*, com as seguintes estratégias: Seleção Sequencial para Frente (*Forward Selection*), Seleção Sequencial para Trás (*Backward Selection*), Híbrida e Análise de Componentes Principais (PCA). Para os casos de redução de dados horizontal as técnicas de: Segmentação de Banco de Dados, Eliminação direta de casos, Amostragem Aleatória e Agregação de Informações.

Conforme literatura e apontamentos de FERLIN [23], é possível também optar-se por uma seleção híbrida onde os dois tipos de redução são realizados, o que pode tornar o processo de seleção mais complexo e custoso em termos de processamento.

2.3.1.2. Discretização e Binarização

O objetivo desta etapa é adaptar os valores da amostragem às limitações do algoritmo de mineração a ser usado, ou seja, garantir que os dados estejam em conformidade com determinado método. Segundo GOLDSCHMIDT [24], duas técnicas podem ser aplicadas: codificação numérico-categórica, que mapeia valores originalmente numéricos em categorias (ex: substituir o valor 1 por “ativo” e 0 por “inativo” no campo status), e codificação categórico-numérica, que mapeia atributos qualitativos em atributos quantitativos, onde os valores que definem o domínio de um atributo são transformados na representação binária de números discretos (ex: substituir o valor “ativo” por 1 e “inativo” por 0 no campo status).

2.3.1.3. Limpeza de Dados

Esta etapa visa identificar e corrigir erros nas bases de dados. Entre os possíveis

problemas temos: dados incompletos, dados discrepantes, dados inconsistentes. Na literatura, de forma geral classifica-se como dado incompleto aquele que não está suficientemente detalhado ou está ausente. Dado discrepante é aquele com "ruído", ou seja, com algum valor atípico e dado inconsistente é aquele com desvio semântico. Para cada problema existe uma ação mais indicada.

Na complementação de dados, os dados faltantes podem ser preenchidos de forma manual, embora impraticável na maioria dos casos devido ao volume e desconhecimento dos dados, ou através de técnicas de imputação baseadas em regras e realizadas através de métodos de mineração de dados que estimam valores o mais adequados possíveis para serem inseridos, sempre de acordo com o tipo do atributo e com seu domínio, avaliado no conjunto de dados existentes.

A detecção de ruídos busca identificar dados em que os valores originais foram modificados. E tratar *outliers*: dados com valores que destoam muito da tendência geral de valores para o atributo. Para tanto, técnicas de inspeção e agrupamento são utilizadas de forma a reconhecer os valores típicos, isolá-los e/ou suavizá-los. FERLIN [23] cita métodos de agrupamento, como o encaixotamento e a regressão.

2.3.1.4. Transformação de Atributo

No contexto do processo de busca de conhecimento em base de dados, a natureza dos dados pode levar a padrões tendenciosos, pois o domínio dos valores de cada atributo, mesmo focando apenas atributos numéricos contínuos, é muito diversificado, Sendo assim, alguns valores podem gerar desvios e invalidar o processo de extração de conhecimento como um todo. Para evitar isso, recorre-se à normalização de dados para redefinir grandeza de valores de atributos para uma escala mais apropriada ao processo. É bastante comum mapear-se os valores originais para uma escala de intervalo [0.0, 1.0]

ou [-1.0, 1.0].

2.3.2. Mineração de Dados

Mineração de Dados (MD) é uma etapa do processo de *KDD* [31] responsável pela aplicação de algoritmos de Aprendizado de Máquina (AM) [49] capaz de encontrar um modelo que reflete o conjunto de dados, explicitando padrões escondidos. Dependendo das características de domínio, e objetivos a serem alcançados pode-se escolher entre uma variedade de algoritmos como: Baseados em Regras, Redes Neurais, Árvores de Decisão, *Support Vector Machine* e Regressão Linear [31].

Aprendizado é qualquer mudança num sistema que melhore o seu desempenho na segunda vez que ele repetir a mesma tarefa. Aprendizado de Máquina (AM) é uma parte da Inteligência Artificial (IA) responsável pelo desenvolvimento de teorias computacionais focadas na criação do conhecimento artificial [31]. Softwares desenvolvidos com esta tecnologia possuem a característica de tomarem decisões com base no conhecimento prévio acumulado através da interação com o ambiente. AM (do inglês, *Machine Learning*) é a área de IA cujo objetivo é o desenvolvimento de técnicas computacionais sobre processo de aprendizado [49].

Técnicas de AM serão utilizadas, neste trabalho, para apoiar a geração de um classificador, que seja capaz de indicar se existe correspondência entre pares de entidades de duas ontologias. Classificação é a tarefa de aprender uma função alvo f que mapeie cada conjunto de atributo x para um dos atributos alvos y pré-determinados. A função alvo também é conhecida informalmente como modelo de classificação. Um modelo de classificação é útil para os seguintes propósitos: (i) modelagem descritiva, pode servir como ferramenta explicativa para se distinguir entre objetos e classes diferentes. (ii) Modelagem preditiva, pode ser usado para prever o rotulo da classe de registros não

conhecidos. Um modelo de classificação pode ser tratado como uma caixa preta que atribui automaticamente um rótulo da classe quando recebe o conjunto de atributos de um registro desconhecido. O modelo classificador gerado pelo algoritmo de aprendizagem deve se adaptar bem aos dados de entrada e prever corretamente os rótulos de classes de registros que ele nunca viu antes [76]. Portanto, um objetivo chave do algoritmo de aprendizagem é construir modelos que prevejam com precisão os rótulos de classes de registros não conhecidos previamente. O aprendizado do modelo é chamado supervisionado sempre que a classe exemplo é conhecida e levada em consideração durante o processo de aprendizagem. O modelo aprendido pode ser um classificador ou um preditor, dependendo do tipo da variável de classe. Se a variável de classe é nominal (isto é, *true / false*), o modelo é um classificador. Caso contrário, é um preditor [31].

Existem vários algoritmos de AM, tendo em conta cada hiperparâmetros do algoritmo, que são as variâncias dos componentes não-observáveis presente no modelo [40]. Com base no estudo de Thornton *et al.* [72] foram selecionados os algoritmos de *Support Vector Machine* (*SVM*), *Multi Layer Perceptron* (*MLP*) e *Random Forest* (*RF*), algoritmos que serão descritos da Seção 2.3.2.1 até a 2.3.2.3.

2.3.2.1. Multi Layer Perceptron

As redes *Multi Layer Perceptron* (*MLP*) têm sido aplicadas em uma variedade de áreas, desempenhado tarefas tais como: classificação de padrões, controle e processamento de sinais [33]. Uma Rede Neural Artificial (RNA) do tipo *MLP* (Figura 7) é constituída por um conjunto de nós de origem, os quais formam a camada de entrada de rede. Uma camada de entrada: consiste em uma camada com os sinais de entrada (estímulo da rede), esta camada não possui neurônios. Após a camada de entrada, pode haver uma ou mais camadas escondidas ou intermediárias que consistem em camadas que

se encontrem. Não existem limites para quantidade de camadas escondidas, e também não é obrigatória a existência delas. Finalmente, há também uma camada de saída, que consiste em uma camada de neurônios que geram a saída da rede [30] (resposta da rede a um estímulo). O valor do Bias, que é usado para aumentar os graus de liberdade, permitindo uma melhor adaptação por parte da rede neural ao conhecimento a ela fornecido, é ajustado da mesma forma que os pesos sinápticos, e permite que um neurônio apresente saída não nula ainda que todas as suas entradas sejam nulas. Os pesos sinápticos são usados para armazenar o conhecimento, uma sinapse é o nome dado à conexão existente entre os neurônios. A Bias é incluída no somatório da função de ativação, com o objetivo de aumentar o grau de liberdade desta função e, consequentemente, a capacidade de aproximação da rede [11].

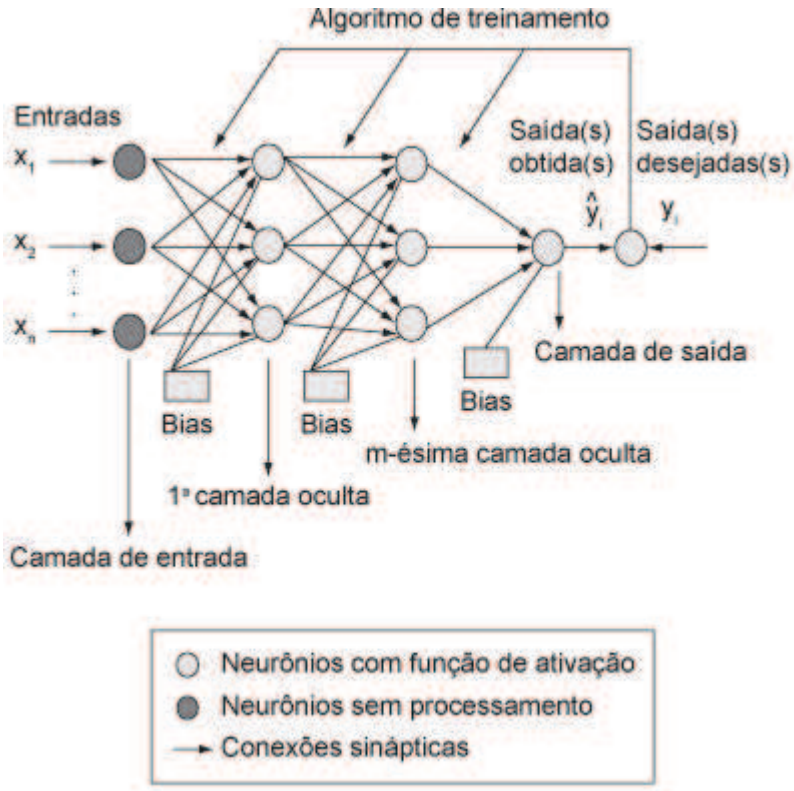


Figura 7 - Arquitetura de uma Rede *MLP*. Fonte [11]

A *MLP* da Figura 7 é uma rede multi camada fortemente conectada com conexões *feedfoward*. Ou seja, é uma rede em que as camadas estão organizadas em uma ordem e

os neurônios da camada estimulam todos os neurônios da camada seguinte (fortemente conectada). Nenhum neurônio pode estimular um neurônio da mesma camada ou de camadas anteriores (*feedfoward*). O aprendizado supervisionado da rede se dar com os pares de conjuntos de entrada e saída desejada. Quando é apresentada à rede este conjunto de entrada, esta retorna um conjunto de valores de saída, que é comparado com o conjunto de valores de saída desejado. O processo é repetido para todos os pares de entrada e saída que constituem o conjunto de treinamento da rede, até que a taxa de acerto seja considerada satisfatória.

O algoritmo de treinamento de uma *MLP* mais famoso é o proposto por (RUMELHART e MCCLELLAND, 1986), conhecido como *back-propagation* (retro-propagação). Este algoritmo consiste em um algoritmo supervisionado estático (não auto organizável), onde a arquitetura da rede deve ser previamente conhecida, e esta não é alterada durante o treinamento. Os únicos parâmetros alterados são os pesos da rede.

2.3.2.2. SVM

É uma Poderosa metodologia para resolver problemas de aprendizagem de máquina e que tem recebido considerável atenção. Foi proposto por VAPNIK [73]. Possui seus fundamentos na teoria de aprendizagem estatística e tem mostrado resultados empíricos promissores em muitas aplicações práticas. Esse algoritmo encontra um tipo especial de modelo linear, chamado hiperplano de margem máxima, que busca separar corretamente todos os casos de uma base entre as classes. Aqueles casos mais próximos do hiperplano de margem máxima são os vetores de suporte. Existe sempre pelo menos um vetor de suporte para cada classe, frequentemente existindo mais [76] [55] [57]. Consiste em um método de aprendizado que tenta encontrar a maior margem para separar diferentes classes de dados em um hiperplano como mostra a Figura 8.

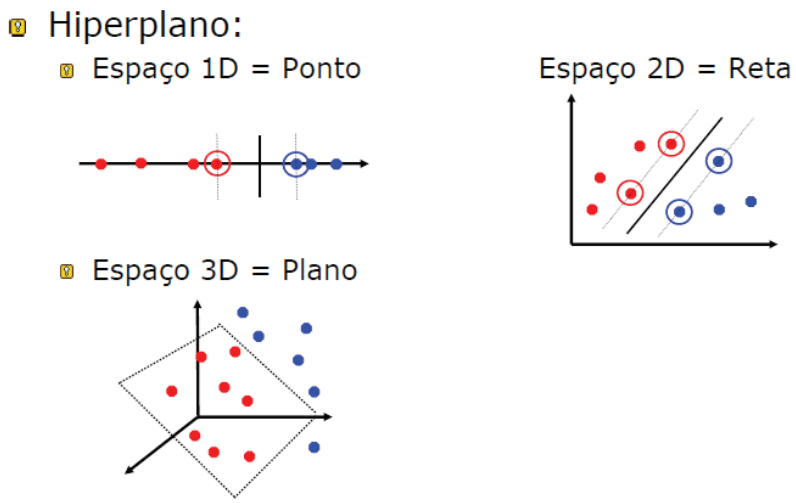


Figura 8 - Hiperplanos ponto, reta e plano.

Fonte: [46]

A essência do *SVM* é a construção de um hiperplano ótimo. O hiperplano ótimo é aquele que maximiza a largura da margem, sendo esta a distância que separa as duas classes. Isso é feito minimizando-se a distância de um caso mal classificado de sua margem do hiperplano. De modo que ele possa separar diferentes classes de dados com a maior margem possível [55] [57] [51] chamada Soft Margin, (Figura 9). A separação ótima entre classes ocorre por meio de um hiperplano condicional (L), tal que este plano é orientado para maximizar a margem (distância entre as bordas, L1 e L2) e pelo ponto mais próximo de cada classe.

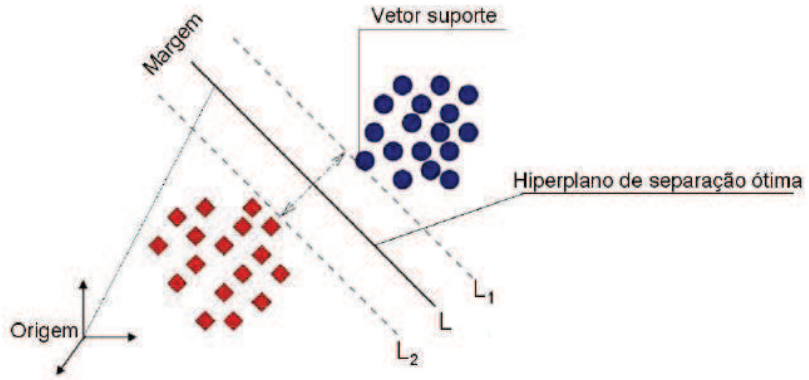


Figura 9 - Hiperplano Ótimo. Fonte: [51]

O *SVM* foi originalmente concebido para lidar com classificações binárias, entretanto na maior parte dos problemas reais requer múltiplas classes. Para se utilizar uma *SVM* para classificar múltiplas classes é necessário transformar o problema multiclasse em vários problemas de classes binárias [9] [45]. Outro fator importante a considerar é que em muitos problemas reais as classes não são linearmente separáveis mesmo utilizando a margem de folga, nestes casos a abordagem utilizada pelo *SVM* para resolver esse tipo de problema consiste em mapear os dados para um espaço de dimensão maior [11]. Na Figura 10 a temos o exemplo do plano 1D em que os dados não são linearmente separáveis, e na Figura 10 b temos esses dados sendo mapeados para um dimensão maior (2D), onde eles são linearmente separáveis.

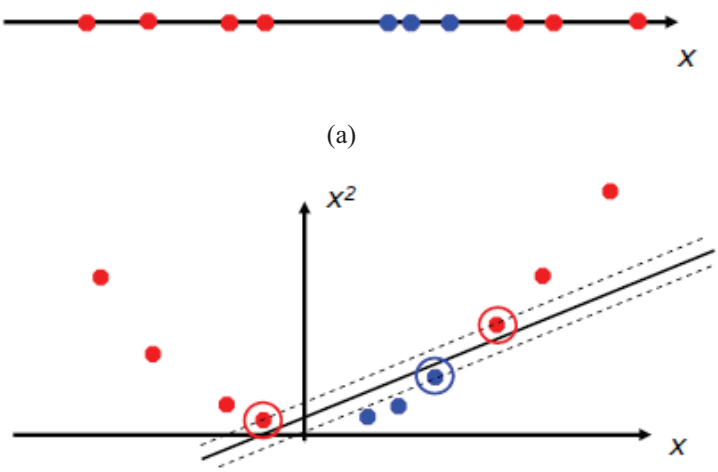


Figura 10 - *SVM* Mapeando os Dados para um Espaço de Dimensão Maior. Fonte: [46]

Mais do que simplesmente adequar a curva aos dados, o algoritmo de *SVM* utiliza uma função *kernel* para mapear os dados em um espaço diferente em que o hiperplano pode ser utilizado para fazer a separação linear [11]. O conceito de utilização de uma função *kernel* é poderoso porque permite que o modelo realize separações mesmo com fronteiras bastante complexas. O *SVM* possui quatro funções *kernel*, sendo elas (i) linear,

(ii) quadrática, (iii) polinomial e (iv) função de base radial. Neste estudo será utilizada a função de base radial (FBR). Esta função foi escolhida, pois segundo indicação do trabalho de [51] todas as outras três funções são variações da FBR e segundo é citado em [51] os trabalhos de Brown *et al*. (2000), Huang *et al*. (2002), Melgani e Bruzzone (2004) indicam que o FBR apresenta os melhores resultados na separação ótima de classes.

O algoritmo pode ser descrito da seguinte forma: dadas 'Z' amostras de treinamento {xi, yi}, com i =1,2..., Z, onde $xi \in \Re^M$ é uma representação vetorial de um conjunto e yi $\in$ {-1,1} é sua classe associada. Neste processo existe uma distribuição de probabilidade P(x, y) desconhecida da qual os dados de treinamento serão retirados. Ou seja, o processo de treinamento consiste em treinar um classificador de forma que este aprenda um mapeamento $\chi \rightarrow y$ por meio de exemplos (classes) de treinamento {xi, yi} de forma que seja capaz de classificar um exemplo (x, y) ainda não visto que siga a mesma distribuição de probabilidade (P) dos exemplos de treinamento.

2.3.2.3. Random Forest

Árvores de decisão são técnicas de modelagem populares, mas modelos de uma única árvore podem ser instáveis e excessivamente sensíveis a dados específicos dentro das bases de treinamento. Modelos que combinam diversas árvores resolvem esse problema, combinado os resultados para determinar a classe de cada caso, sendo a floresta aleatória (*random forest*) um desses modelos. As florestas aleatórias [6] são um conjunto de classificação não podada (*unpruned*) ou árvores de regressão. A Floresta aleatória gera muitas árvores de classificação, a quantidade de arvores pode ser configurada. Cada árvore é construída por uma amostra de *Bootstrap,* que são novas amostras geradas a partir dos dados originais, utilizando um algoritmo de classificação de árvore. Após a

floresta ser formada, e um novo objeto precise ser classificado, esse objeto é colocado para que cada uma das árvore na floresta faça a classificação. Cada árvore dá um voto que indica a decisão da árvore sobre a classe que o objeto pertence. A floresta então escolhe a classe com o maior número de votos para o objeto.

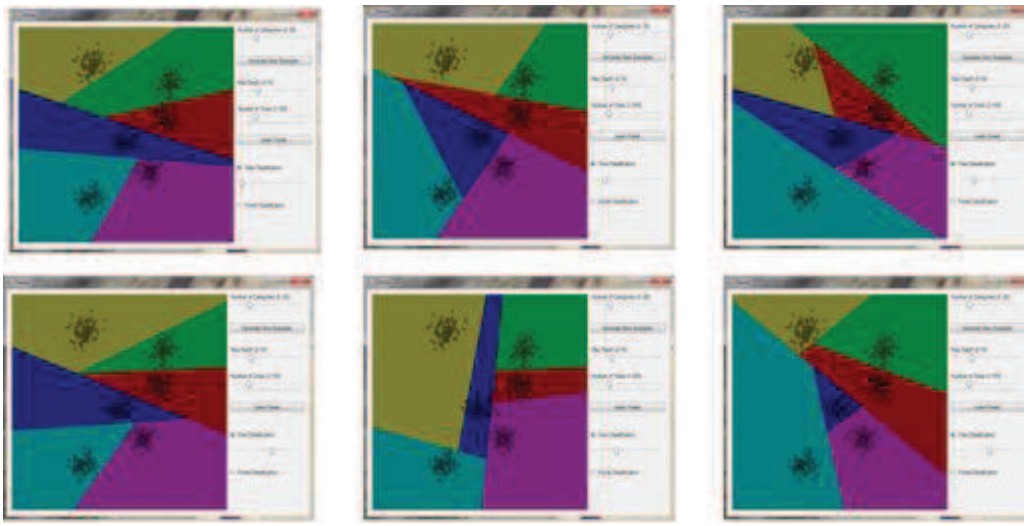


Figura 11 - Diferentes Classificações Pelas Árvores da Floresta Aleatória. Fonte: [2]

No exemplo da Figura 11 Árvores diferentes dentro de uma floresta pode dar origem a diferentes classificações (votos): Amarelo, Verde, Vermelho, Azul Marinho e Azul Claro. Breiman (2001) demonstrou que o um melhor desempenho poderia ser conseguido através da injeção de aleatoriedade, a fim de minimizar a correlação entre os modelos de base, mantendo a precisão. Em *RF*, isto é conseguido através da combinação de duas fontes de aleatoriedade. Em primeiro lugar, os casos utilizados para cada árvore são amostrados aleatoriamente, sem reposição, do conjunto de treino inicial. Em segundo lugar, *RF* consiste em usar os recursos selecionados aleatoriamente em cada nó. Usando a forte *Law of Large Numbers*[8] [34], Breiman demonstrou que *RF* convergem sempre de

[8] Na teoria da probabilidade, *Law of Large Numbers* é um teorema que descreve o resultado da realização da mesma experiência de um grande número de vezes. De acordo com a lei, a média dos resultados obtidos a partir de um

modo que *overfitting* não é um problema, isto é, *RF* nunca causa *overfitting* quanto mais árvores são adicionadas. Múltiplos estudos empíricos [6] demonstram *RF* de ser competitivo em precisão com os melhores algoritmos de classificação e regressão em certo número de domínios de aplicação.

2.4. Pós-Processamento

Nesta etapa é feito o tratamento e a avaliação do conhecimento extraído da base de dados envolvendo as tarefas de análise, interpretação e avaliação. Nessa etapa é medida a acurácia e verificadas as medidas de erro, que irão indicar se o que foi aprendido, de fato, ajuda quando novos objetos são examinados. Uma técnica para avaliar a acurácia é dividir a base de dados original em dois conjuntos: conjunto de treinamento – dados que serão utilizados na construção do modelo de conhecimento e conjunto de testes – dados que serão utilizados na validação do modelo. Existem diferentes técnicas para realizar a divisão, tais como: *Holdout*, Validação Cruzada com K Conjuntos, Validação Cruzada com K Conjuntos Estratificada*, Leave-One-Out* e *Bootstrap*. [24], [55], [76], [23].

Para avaliar a proposta foram utilizadas as medidas de Precisão, Cobertura e Medida-F, e a divisão do conjunto de amostras dos dados foi utilizado a técnica SMOTE (*Synthetic Minority Oversampling*) [10] para realizar o balanceamento [55].

A validação cruzada com K conjuntos (*k-folds*) foi utilizada. A validação cruzada é uma técnica para avaliar a capacidade de generalização de um modelo, a partir de um conjunto de dados. Esta técnica é amplamente empregada em problemas onde o objetivo da modelagem é a predição [76], [43]. Busca-se então estimar o quão acurado é este modelo na prática, ou seja, o seu desempenho para um novo conjunto de dados. Método

---

grande número de ensaios deve ser próximo do valor esperado, e tenderão a tornar-se mais próximos à medida que mais experimentos são realizados.

de validação cruzada denominado *k-fold* consiste em dividir o conjunto total de dados em k subconjuntos mutualmente exclusivos do mesmo tamanho e, a partir disto, um subconjunto é utilizado para teste e os k-1 restantes são utilizados para estimação dos parâmetros e calcula-se a acurácia do modelo [43]. A seguir são explicadas as medidas que foram utilizadas para a avaliação da acurácia.

A seleção das métricas para a avaliação da acurácia é uma tarefa que deve levar em consideração o critério de não favorecer qualquer abordagem. Para a tarefa de alinhamento de ontologias, tipicamente têm sido utilizadas as seguintes métricas [17]:

- ✓ Verdadeiros Positivos (VP): É o conjunto de correspondências encontradas, que fazem parte do conjunto de correspondências do alinhamento de referência;
- ✓ Falsos Positivos (FP): É o conjunto de correspondências encontradas, que não fazem parte do conjunto correspondências do alinhamento de referência;
- ✓ Falso Negativos (FN): É o conjunto de pares de elementos que não foram identificados como correspondências possíveis e que estão presentes no conjunto de correspondências do alinhamento de referência;
- ✓ Precisão (*Precision*): A precisão mede a proporção de correspondências corretas que foram encontradas, ou seja, dentre as correspondências encontradas quantas realmente são? Dado um alinhamento de referência *R*, a precisão de um alinhamento *A* pode ser calculada da seguinte forma:

$$P(A,R) = \frac{|\mathrm{VP}|}{|\mathrm{VP}+\mathrm{FP}|}$$

- ✓ Cobertura (*Recall)*: A cobertura mede a proporção de correspondências corretas encontradas dentre todas as possíveis. Dado um alinhamento de referência *R*, a cobertura de um alinhamento *A* pode ser calculada da seguinte forma:

$$R(A,R) = \frac{[\mathrm{VP}]}{[\mathrm{VP} + \mathrm{FN}]}$$

- ✓ Medida-F (*F-Measure*): A Medida-F representa a harmonização entre a precisão e a cobertura. Pode ser a medida principal para avaliar a qualidade de um alinhamento. Dado um alinhamento de referência *R*, a precisão e a cobertura, a medida-F pode ser calculada da seguinte forma:

$$F(A,R) = \frac{(b^2+1).P(A,R).R(A,R)}{b^2.\mathrm{P(A,R)}+\mathrm{R(A,R)}}$$

Com *b*=1sendo um fator de peso padrão, chega-se a:

$$F_1(A,R) = \frac{2.P(A,R).R(A,R)}{P(A,R)+R(A,R)}$$

Capítulo 3 – Abordagem Proposta

Neste capítulo é apresentada a abordagem proposta para geração de um classificador para alinhamento de ontologias, a partir da combinação de grupos de métricas de similaridades, utilizando Mineração de Dados. O objetivo é melhorar precisão e cobertura no processo de alinhamento de ontologias, com a combinação de diferentes grupos de métricas de similaridade.

3.1. Considerações Gerais

Diversas soluções de alinhamento de ontologias vêm sendo propostas nos últimos anos, dentre elas as que utilizam a combinação de métricas de similaridade. Alguns trabalhos como o *GOMMA* [41], *LogMap* [11] [62], CODI [35] [53], YAM ++ [16] e COMA [48] utilizam a combinação de métricas baseadas em *Strings*, baseadas em linguagem, baseadas em recursos linguísticos, baseadas em análise de dados e estatística, baseadas em modelo, baseadas em reuso de alinhamento, baseados em grafos e baseadas em ontologia de topo. Porém, a combinação de várias métricas de similaridade para obter resultados mais precisos é uma tarefa muito complexa, e em alguns casos com a necessidade da intervenção do usuário para a validação dos alinhamentos. Outro fator importante é que, em algumas dessas abordagens, é necessário definir um valor de *threshold* (corte / limiar) para considerar válido, ou não, o alinhamento de acordo com o valor de força retornado pelas métricas de similaridade. Diante dos fatos citados, as abordagens automáticas são necessárias para apoiar o alinhamento de ontologias. A Mineração de Dados pode ser utilizada para extrair o modelo para combinação de

métricas. Assim, o problema de alinhamento é transformado em uma tarefa aprendizado de máquina.

A abordagem proposta cria um classificador que combina automaticamente grupos de métricas de similaridade através da utilização de técnicas de Mineração de Dados. Os grupos de métricas de similaridade foram selecionados de acordo com a Tabela 1. Existem diversas métricas de similaridades que são empregadas nas pesquisas em alinhamento de ontologia. Nesse trabalho selecionamos as técnicas mais populares de similaridades que já estão implementadas na API de alinhamento [37] e no projeto SimMetrics[9], por que essas ferramentas são consenso entre os pesquisadores da área. Este trabalho não tem por objetivo avaliar qualquer ferramenta de alinhamento ou métrica de similaridade individualmente.

As métricas que serão usadas para a geração do classificador foram agrupadas de acordo com os quatro grupos de métricas apresentados no Capítulo 2 (baseadas em *Strings*, Linguagem, Recurso Linguístico e Análise de Dados e Estatística), como mostra a coluna 1 da Tabela 1.

[9]http://sourceforge.net/projects/simmetrics/ - SimMetrics é uma Biblioteca de Métrica de Similaridade, e.g. de edição de distância *(Levenshtein, Gotoh, Jaro, etc.*) para outras métricas, (por exemplo, *Soundex, Chapman*). Trabalho fornecido pela *UK Sheffield University*.

Tabela 1 - Métricas de similaridades utilizadas na proposta

Grupo de Métricas	#	Métricas de Similaridade	Origem
Baseada em Recursos Linguísticos	M1	JWNL Alignment	API Alinhamento
Baseada em Linguagem	M2	QGramDistance	Projeto SimMetrics
	M3	ChapmanMatchingSoundex	Projeto SimMetrics
	M4	ChapmanOrderedNameCompoundSimilarity	Projeto SimMetrics
	M5	SmithWatermanGotohWindowedAffine	Projeto SimMetrics
	M6	Soundex	Projeto SimMetrics
	M7	TagLinkToken	Projeto SimMetrics
Baseada em String	M8	Levenshtein	Projeto SimMetrics
	M9	SmithWaterman	Projeto SimMetrics
	M10	SmithWatermanGotoh	Projeto SimMetrics
	M11	JaroWinker	Projeto SimMetrics
	M12	MongeElkan	Projeto SimMetrics
	M13	Jaccard	Projeto SimMetrics
	M14	Euclidean	Projeto SimMetrics
	M15	ChapmanLengthDeviation	Projeto SimMetrics
	M16	BlockDistance	Projeto SimMetrics
	M17	Jaro	Projeto SimMetrics
	M18	ChapmanMeanLength	Projeto SimMetrics
	M19	MatchingCoefficient	Projeto SimMetrics
	M20	CosineSimilarity	Projeto SimMetrics
	M21	NeedlemanWunch	Projeto SimMetrics
Baseado em Análise de Dados e Estatística	M22	DiceSimilarity	Projeto SimMetrics
	M23	OverlapCoefficient	Projeto SimMetrics

3.2. Proposta

O presente trabalho tem como objetivo a criação de um classificador para o alinhamento de ontologias, utilizando a combinação de diferentes grupos de métricas de similaridade, com a utilização de técnicas de Mineração de Dados. Um processo sistemático foi seguido para a obtenção do classificador, conforme a visão geral apresentada na Figura 12.

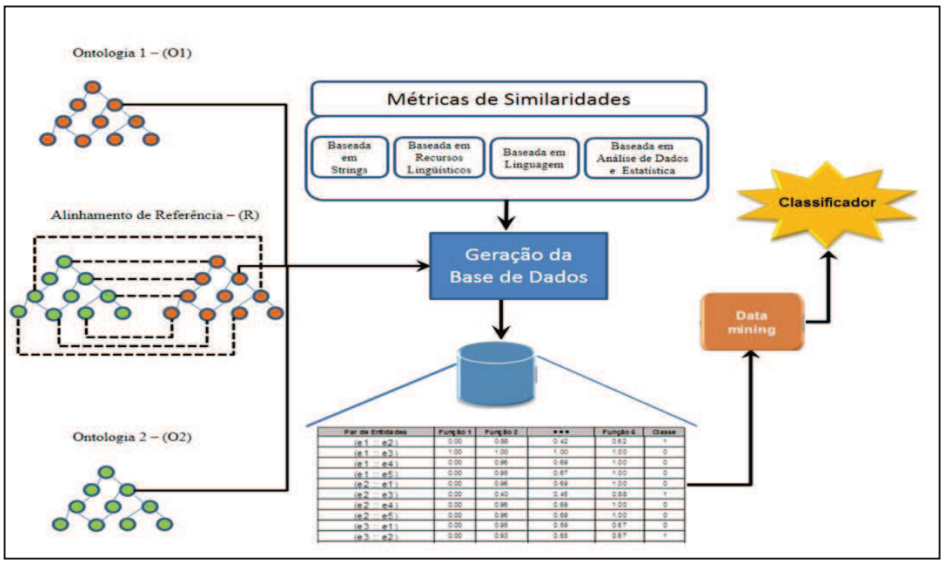


Figura 12 - Visão da Abordagem Proposta

Como ilustrado na Figura 12, são necessários alguns insumos para o aprendizado do classificador: (i) duas ontologias de entrada a serem alinhadas, O1 e O2, (ii) um alinhamento de referência R entre as duas ontologias de entrada O1 e O2, e (iii) a seleção das métricas de similaridades (dentre as descritas na Tabela 1).

A partir destes insumos, a base de dados para aprendizado do classificador é obtida da seguinte forma, conforme ilustra a Figura 13.

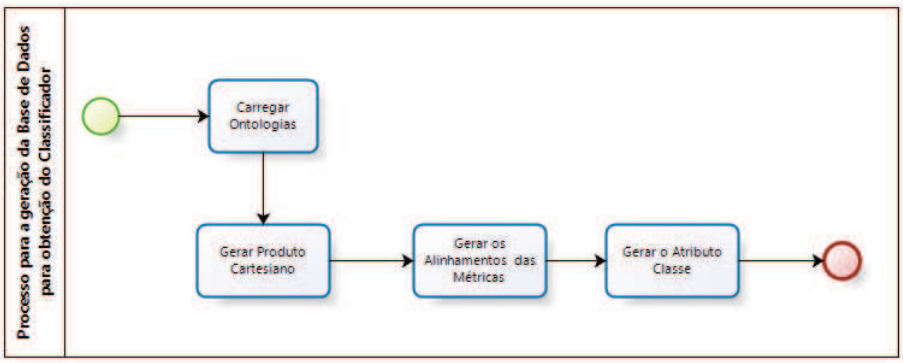


Figura 13 - Passos para Geração da Base de Dados para Obtenção do Classificador

São carregadas as ontologias O1 e O2 e gerado o produto cartesiano entre todas as suas entidades, resultando, portanto, em um conjunto de pares (e1, e2) varrendo todas as combinações onde e1 é uma entidade de O1 e e2 é uma entidade de O2. A geração do produto cartesiano foi definida durante experimentos realizados no trabalho [65] que foi publicado com a proposta inicial. Isso determina o número de instâncias no conjunto de dados, representados na primeira coluna da Tabela 2. Para cada par de entidades, calcula-se o valor de força (no intervalo [0,1]) retornado por cada métrica de similaridade considerada, representados pelas colunas intermediárias da base de dados de entrada (colunas 2-6 da Tabela 2). Finalmente, o atributo classe ou alvo (última coluna da Tabela 2) da base de dados é preenchido com base no alinhamento de referência R de entrada: é atribuído o número 1 aos pares de elementos contidos no alinhamento de referência R e 0 para os pares que não estão presente no alinhamento de referência.

Tabela 2 - Exemplo do conjunto de dados com diferentes métricas de similaridades e o produto cartesiano das entidades das ontologias.

Par de Entidades	Função 1	Função 2	Função 3	Função 4	Função 5	Função 6	Classe
(e1 :: e2)	0,00	0,88	0,43	0,62	0,42	0,62	1
(e1 :: e3)	1,00	1,00	1,00	1,00	1,00	1,00	0
(e1 :: e4)	0,00	0,96	0,67	1,00	0,69	1,00	0
(e1 :: e5)	0,00	0,95	0,63	1,00	0,67	1,00	0
(e2 :: e1)	0,00	0,96	0,67	1,00	0,69	1,00	0
(e2 :: e3)	0,00	0,40	0,37	0,95	0,45	0,88	1
(e2 :: e4)	0,00	0,96	0,67	1,00	0,69	1,00	0
(e2 :: e5)	0,00	0,96	0,67	1,00	0,69	1,00	0
(e3 :: e1)	0,00	0,95	0,73	0,67	0,59	0,67	0
(e3 :: e2)	0,00	0,93	0,73	0,73	0,65	0,67	1
(e3 :: e4)	0,00	0,48	0,14	0,30	0,00	0,25	1
(e3 :: e5)	0,00	0,95	0,59	1,00	0,71	1,00	1
(e4 :: e1)	0,00	0,96	0,71	1,00	0,71	1,00	0
(e4 :: e2)	0,00	0,72	0,27	0,38	0,17	0,33	1
(e4 :: e3)	0,00	0,79	0,56	0,60	0,47	0,46	1
(e4 :: e5)	0,00	0,30	0,53	0,67	0,44	0,41	1
(e5 :: e1)	1,00	1,00	1,00	1,00	1,00	1,00	1
(e5 :: e2)	1,00	1,00	1,00	1,00	1,00	1,00	1
(e5 :: e3)	1,00	1,00	1,00	1,00	1,00	1,00	1
(e5 :: e4)	0,00	0,49	0,36	0,93	0,42	0,83	1
(e6 :: e1)	0,00	0,88	0,43	0,62	0,42	0,62	1
(e6 :: e2)	1,00	1,00	1,00	1,00	1,00	1,00	1
(e6 :: e3)	0,00	0,97	0,75	1,00	0,67	1,00	0
(e6 :: e4)	0,00	0,97	0,75	1,00	0,67	1,00	0
(e6 :: e5)	1,00	1,00	1,00	1,00	1,00	1,00	1
(e7 :: e1)	1,00	1,00	1,00	1,00	1,00	1,00	0
(e7 :: e2)	1,00	1,00	1,00	1,00	1,00	1,00	1
(e7 :: e3)	1,00	1,00	1,00	1,00	1,00	1,00	1

No conjunto de dados que será usado para o experimento desta dissertação, cada métrica de similaridade representa um atributo que contém os valores encontrados (força / confiança) para cada correspondência. A idéia é gerar um modelo classificador que combine várias métricas de similaridade.

Para a aplicação dos algoritmos de mineração é aplicada a técnica de redução de dados vertical, excluindo-se a primeira coluna da tabela gerada inicialmente, uma vez que ela representa o identificador da tupla e, por conseguinte, não contribui para o modelo de classificação a ser aprendido.

O passo seguinte é o aprendizado do classificador, onde pode ser utilizada qualquer ferramenta de Mineração de Dados. Neste experimento, foi utilizada a bem conhecida ferramenta de Aprendizado de Máquina Weka (versão 3.6.10 Linux, disponibilizada pela universidade de *Waikato* Nova Zelândia), em que os dados dos alinhamentos com as diferentes métricas de similaridades são carregados, é aplicada a partição dos dados, e são executado os algoritmos de aprendizado de máquina utilizados na proposta, o *MLP* [61], *RF* [6] e *SVM* [73]. Tais algoritmos foram selecionados porque no trabalho de Thornton *et al.* [72] ele considerou o problema de selecionar, simultaneamente, um algoritmo de aprendizagem, definindo os seus hiperparâmetros e mostrou como que este problema pode ser resolvido por uma abordagem totalmente automatizada. Com base neste estudo, foram selecionados os algoritmos *SVM, MLP* e *RF*, que foram os algoritmos que obtiveram os melhores resultados, quando testados em 21 bases de dados de diferentes domínios e tamanhos.

Capítulo 4 – Experimentos

Neste capítulo serão descritos o planejamento dos experimentos com a dinâmica das atividades realizadas, as bases de dados utilizadas e os cenários de todos os experimentos realizados, bem como os resultados das avaliações dos experimentos.

4.1. Base de Dados dos Experimentos

O OAEI [19] é uma iniciativa internacional coordenada, cujo objetivo é avaliar os pontos fortes e fracos de ferramentas de alinhamento, comparar o desempenho de técnicas e melhorar as técnicas de avaliação. Essa base dispõe de uma grande variedade de domínios com os seus alinhamentos de referência, geralmente denominados de trilhas para as competições anuais. Os conjuntos de dados fornecidos pelo OAEI estão em conformidade com os critérios exigidos para a conclusão de uma avaliação do projeto de alinhamento de ontologias, e possuem as características necessárias para serem utilizados como cenários para os experimentos. Em particular, foram escolhidos os domínios de *Benchmark, Conference* e *MultiFarm* fornecidos pela campanha OAEI em 2012 para executar nosso experimento. Cada domínio irá gerar uma base de dados, que serão descritos na Seção a seguir. Esses domínios foram escolhidos levando em consideração o histórico das avaliações desde 2005 até 2012 como é visto na Figura 1, onde nota-se que eles obtiveram as piores médias na avaliação da Medida-F.

4.1.1. Base de Dados *Benchmarck*

O objetivo do teste de *Benchmark* é oferecer um conjunto de testes que sejam de ampla cobertura em recursos, progressivo e estável. Eles servem o propósito de avaliar a força e a fraqueza de *matchers* (por ser progressivo e de ampla cobertura) e medir o progresso dos *matchers* (por ser estável e reutilizável ao longo dos anos). O objetivo desta série de *Benchmark* é de identificar as áreas em que cada algoritmo de alinhamento é forte e fraco. Esse domínio proporciona resultados fortemente comparáveis e permite testar escalabilidade [19].

O teste do domínio de *Benchmark* consiste de um conjunto de dados que é construídos a partir de ontologias de referência de diferentes tamanhos e de diferentes domínios. A ontologia bibliográfica tem sido a principal ontologia de referência desde o início das campanhas OAEI.

O domínio do primeiro *Benchmark* é de referências bibliográficas. Sua ontologia de referência é baseada em uma visão subjetiva do que deve ser uma ontologia bibliográfica. Pode haver muitas classificações diferentes de publicações (com base na área, qualidade, etc.), mas foi escolhido o mais comum entre os estudiosos com base na média de publicações, a ontologia resultante é semelhante ao *BibTeX.* A ontologia de referência foi elaborada com base no primeiro Concurso de Alinhamento de Ontologias EON[10] (*Ontology Alignment Contest*), que contém 33 classes nomeadas, 24 propriedades do objeto, 40 propriedades de dados, 56 indivíduos nomeados e 20 indivíduos anônimos. Esta ontologia de referência só combina classes e propriedades nomeadas, e principalmente usa a relação "=" com a força de 1. A ontologia completa de referências bibliográficas é a do teste # 101, as outra ontologias são variações da 101.

[10] http://oaei.ontologymatching.org/2004/Contest/

Cada conjunto de dados é composto, geralmente, de 111 testes individuais que confrontam uma ontologia de referência (#101) com uma versão modificada da mesma ontologia. Os testes são gerados sistematicamente a partir da ontologia de referência e o descarte de uma série de informações, a fim de avaliar a forma como o algoritmo se comporta quando essas informações são inexistentes. As ontologias nos testes são descritas em OWL-DL e serializadas no formato RDF / XML.

Existem seis categorias de alteração nas ontologias (variações da #101):

- **Nome -** Nome de entidades que podem ser substituídos por sequências aleatórias, sinônimos com as convenções de nomes diferentes, sequências de caracteres em outro idioma que não o Inglês.
- **Comentários -** Os comentários podem ser suprimidos ou traduzidos em outros idiomas.
- **Hierarquia de Especialização -** Pode ser suprimida, expandida ou achatada.
- **Instâncias -** Podem ser suprimidas.
- **Propriedades -** Podem ser suprimidas ou com restrições sobre as classes descartadas.
- **Classes -** Podem ser expandidas, ou seja, ligadas a várias classes ou achatadas.

Um conjunto de dados de testes está disponível através de um conjunto de pastas (um por teste), cada diretório contendo uma ontologia (onto.rdf) em OWL. As pastas são nomeadas de acordo com os números do teste, por exemplo, a pasta 201 irá conter a ontologia correspondente ao teste 201. Cada lista contém também os alinhamentos de referência contra o qual os resultados do processo de *matching* serão avaliados.

A tarefa de *match* consistirá no alinhamento de cada ontologia teste com a do teste #101. O alinhamento resultante deve ser fornecido no formato RDF / XML. Ele será comparado com o alinhamento de referência para produzir as medições de conformidade

(principalmente de precisão e cobertura), para a ferramenta de *matching* para esse teste. Os únicos alinhamentos interessantes são aqueles que envolvem as classes e propriedades das ontologias dadas. Assim, os alinhamentos não devem alinhar os indivíduos, nem entidades de ontologias externas.

4.1.2. Base de Dados *Conference*

O domínio de Conferência consiste em 16 ontologias são disponibilizadas. O objetivo desta trilha é encontrar alinhamentos dentro de um conjunto de ontologias que descrevem o domínio da organização de conferências. Os alinhamentos de referência são uma coleção de 21 alinhamentos correspondentes no espaço de alinhamento completo entre 7 ontologias (*Cmt, confOf, Edas, Ekaw, Iasted, Sigkdd, Conference*) desses domínios. Estas 7 ontologias são um subconjunto de todas as ontologias (ver Tabela 3) dentro desta faixa. (Existem duas variantes de alinhamento de referência:) [19].

Tabela 3 - Subconjunto das ontologias dentro da faixa Conference da OAEI

Nome	Número de Classes	Número de propriedades de tipo de dados	Número de propriedades de objetos
Ekaw	74	0	33
Conference	60	18	46
Sigkdd	49	11	17
Iasted	140	3	38
ConfOf	38	23	13
Cmt	36	10	49
Edas	104	20	30

i) Alinhamento de referência ra1, que será considerado no nosso experimentos, e ii) O alinhamento de referência ra2, ele é derivado do alinhamento de referência ra1. A fim de obter conjunto de alinhamento de referência coerente, correspondências conflitantes são inspecionadas e resolvidas pelos avaliadores. Como resultado, o grau de exatidão e integridade do ra2 provavelmente é ligeiramente melhor do que para ra1. No

entanto, as diferenças são relativamente pequenas. Os resultados são muito semelhantes conforme demonstraram os resultados da campanha da OAEI-2011.5.

4.1.3. Base de Dados *MultiFarm*

Este domínio é composto por um subconjunto do domínio *Conference* Seção 4.1.2 Tabela 3, traduzido do inglês em oito idiomas (chinês (cn), tcheco (cz), holandês (nl), francês (fr), alemão (de), Português (pt), Russo (ru) e Espanhol (es)) e os alinhamentos correspondentes entre essas ontologias. Com base nestes casos de teste, é possível avaliar e comparar o desempenho das abordagens de alinhamento com um foco especial sobre o multilinguismo [19].

Dentro do conjunto de dados *MultiFarm*, podemos distinguir dois tipos de tarefas de alinhamento: (i) os casos de teste, onde duas ontologias diferentes foram traduzidas em diversas línguas (cmt - confOf, por exemplo), e (ii) os casos de teste onde a mesma ontologia foi traduzida em diferentes línguas (cmt - cmt, por exemplo). Para os processos de teste de tipo (ii), os bons resultados não estão diretamente relacionados com a utilização de técnicas específicas para lidar com ontologias em diferentes línguas naturais, mas com a possibilidade de explorar o fato de ambas as ontologias terem uma estrutura idêntica (e que o alinhamento de referência abrange todas as entidades descritas nas ontologias). Pode-se supor que estes casos de teste são dominadas por técnicas específicas concebidas para combinar diferentes versões da mesma ontologia. Abaixo segue a estrutura das ontologias desse domínio:

ont/

cn/
cmt-cn.owl
conference-cn.owl
[para cada ontologia cmt, conference, confOf, edas, ekaw, iasted, sigkdd]
cz/ (contém 7 arquivos)
cmt-cz.owl
conference-cz.owl
...
de/ (contém 7 arquivos)
cmt-de.owl
conference-de.owl
...
[um diretório para cada linguagem cn, cz, de, en, es, fr, nl, pt, ru]
ref/
cn-cz/
cmt-cmt-cn-cz.rdf
cmt-conference-cn-cz.rdf
cmt-conference-cz-cn.rdf
cmt-confOf-cn-cz.rdf
cmt-confOf-cz-cn.rdf
...
conference-conference-cn-cz.rdf
...
[total 21*2=42+7*1 arquivos]
[um diretório para cada par de linguagem cn-cz, cn-de, ...]

Para o experimento nesse domínio, serão consideradas apenas as tarefas de alinhamento envolvendo os idiomas Inglês e Português, que executa 50 tarefas de alinhamento por cenário experimentado. Como será realizado o experimento em 5 cenários diferentes, 250 tarefas de alinhamento serão geradas só nesse domínio.

Tabela 4 -Resumo da variação dos domínios

Domínio	Formalismo	Força	Relação	Modalidade	Idioma
Benchmark	OWL	[0 1]	(=)	cega+aberta	EN
Conference	OWL-DL	[0 1]	(=, ≤)	cega+aberta	EN
MultiFarm	OWL	[0 1]	(=)	aberta	CZ, CN, DE, EN, ES, DE, FR, RU, PT

Na Tabela 4 são apresentadas as características dos casos de teste: o formalismo da linguagem utilizada, a força, a relação, e a modalidade da avaliação: a aberta é feita com alinhamentos de referência já publicados e a modalidade de avaliação cega é feita pelos organizadores a partir de alinhamentos de referência desconhecidos. E por último, os idiomas utilizados nas ontologias do domino em questão.

4.2. Protótipo Athenas

Para a validação da contribuição da proposta foi construído um protótipo na linguagem Java, utilizando o banco MySQL para armazenar os dados com os resultados dos alinhamentos entres as ontologias utilizadas, que serviram de insumos para o aprendizado do classificador. É possível a seleção das métricas de similaridade a serem usadas na geração dos dados que serão utilizados na construção do classificador e também as entidades das ontologias (*ObjectProperty, DataProperty* e *Class*) que serão consideradas para a geração do produto cartesiano como mostra a Figura 14.

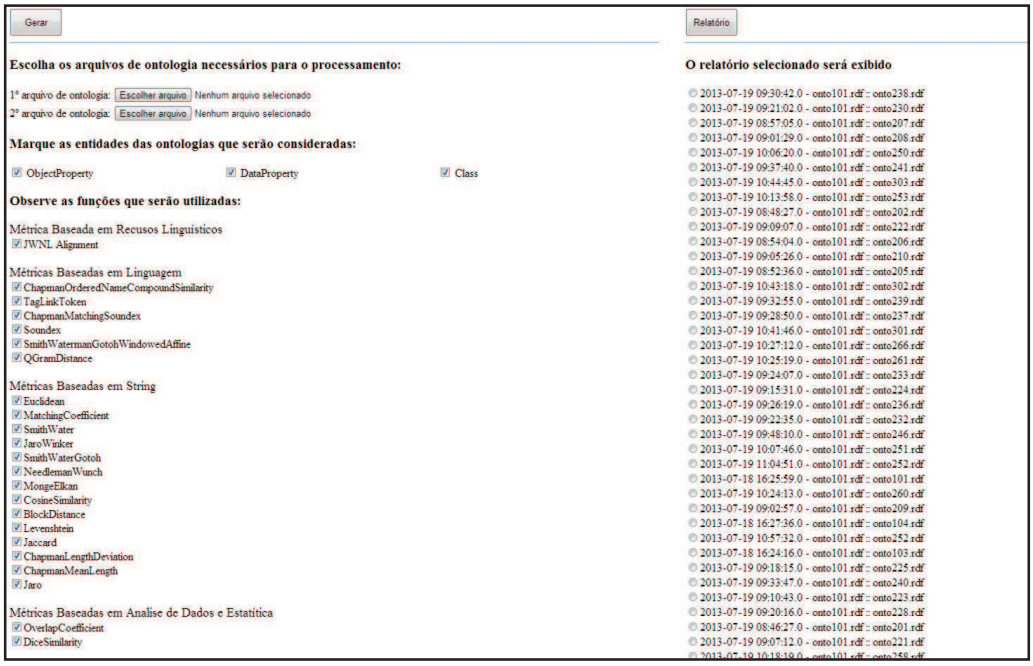
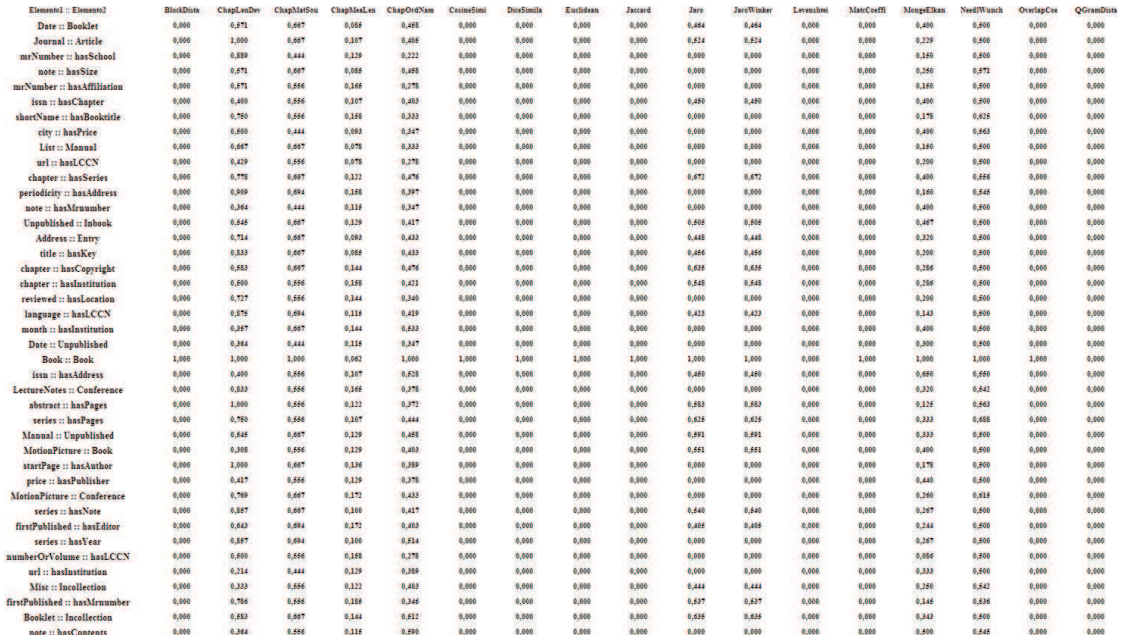


Figura 14 - Protótipo Athenas para Geração dos Dados Necessários para o Processo de Mineração de Dados

As métricas utilizadas na abordagem conforme descrito no 2.2.1, são as baseadas em *Strings*, Linguagem, Recurso Linguístico e Análise de Dados e Estatística. É possível no protótipo Athenas a seleção por grupo de métrica de similaridade. Há a possibilidade da combinação de quatro grupos de métricas (Baseados em *Strings*, Linguagem, Recurso Linguístico e Análise de Dados e Estatística) durante a tarefa de alinhamento de ontologia.



Elemento1 :: Elemento2	BlockDista	ChapLenDev	ChapMatSou	ChapMaxLen	ChapOrdNam	CosineSimi	DiceSimila	Euclidean	Jaccard	Jaro	JaroWinker	Levenshtei	MatchCoeffi	MongeElkan	NeedlWunch	OverlapCoe	QGramDista
Date :: Booklet	0,000	0,571	0,667	0,086	0,488	0,000	0,000	0,000	0,000	0,464	0,464	0,000	0,000	0,400	0,500	0,000	0,000
Journal :: Article	0,000	1,000	0,667	0,107	0,405	0,000	0,000	0,000	0,000	0,524	0,524	0,000	0,000	0,229	0,500	0,000	0,000
mrNumber :: hasSchool	0,000	0,889	0,444	0,129	0,222	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,180	0,500	0,000	0,000
note :: hasSize	0,000	0,571	0,667	0,086	0,468	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,250	0,571	0,000	0,000
mrNumber :: hasAffiliation	0,000	0,571	0,556	0,165	0,278	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,160	0,500	0,000	0,000
issn :: hasChapter	0,000	0,400	0,556	0,107	0,403	0,000	0,000	0,000	0,000	0,450	0,450	0,000	0,000	0,400	0,500	0,000	0,000
shortName :: hasBooktitle	0,000	0,750	0,556	0,188	0,333	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,175	0,625	0,000	0,000
city :: hasPrice	0,000	0,500	0,444	0,093	0,347	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,400	0,563	0,000	0,000
List :: Manual	0,000	0,667	0,667	0,078	0,333	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,160	0,500	0,000	0,000
url :: hasLCCN	0,000	0,429	0,556	0,078	0,278	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,200	0,500	0,000	0,000
chapter :: hasSeries	0,000	0,778	0,667	0,122	0,476	0,000	0,000	0,000	0,000	0,672	0,672	0,000	0,000	0,400	0,556	0,000	0,000
periodicity :: hasAddress	0,000	0,909	0,694	0,188	0,397	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,160	0,545	0,000	0,000
note :: hasMrnumber	0,000	0,364	0,444	0,116	0,347	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,400	0,500	0,000	0,000
Unpublished :: Inbook	0,000	0,545	0,667	0,129	0,417	0,000	0,000	0,000	0,000	0,505	0,505	0,000	0,000	0,467	0,500	0,000	0,000
Address :: Entry	0,000	0,714	0,667	0,093	0,433	0,000	0,000	0,000	0,000	0,448	0,448	0,000	0,000	0,320	0,500	0,000	0,000
title :: hasKey	0,000	0,333	0,667	0,086	0,433	0,000	0,000	0,000	0,000	0,466	0,466	0,000	0,000	0,200	0,500	0,000	0,000
chapter :: hasCopyright	0,000	0,583	0,667	0,144	0,476	0,000	0,000	0,000	0,000	0,635	0,635	0,000	0,000	0,286	0,500	0,000	0,000
chapter :: hasInstitution	0,000	0,500	0,556	0,188	0,421	0,000	0,000	0,000	0,000	0,545	0,545	0,000	0,000	0,286	0,500	0,000	0,000
reviewed :: hasLocation	0,000	0,727	0,556	0,144	0,340	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,200	0,500	0,000	0,000
language :: hasLCCN	0,000	0,875	0,694	0,116	0,419	0,000	0,000	0,000	0,000	0,423	0,423	0,000	0,000	0,143	0,500	0,000	0,000
month :: hasInstitution	0,000	0,267	0,667	0,144	0,533	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,400	0,500	0,000	0,000
Date :: Unpublished	0,000	0,364	0,444	0,116	0,347	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,300	0,500	0,000	0,000
Book :: Book	1,000	1,000	1,000	0,062	1,000	1,000	1,000	1,000	1,000	1,000	1,000	0,000	1,000	1,000	1,000	1,000	0,000
issn :: hasAddress	0,000	0,400	0,556	0,107	0,528	0,000	0,000	0,000	0,000	0,460	0,460	0,000	0,000	0,680	0,560	0,000	0,000
LectureNotes :: Conference	0,000	0,833	0,556	0,165	0,375	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,320	0,542	0,000	0,000
abstract :: hasPages	0,000	1,000	0,556	0,122	0,372	0,000	0,000	0,000	0,000	0,583	0,583	0,000	0,000	0,125	0,563	0,000	0,000
series :: hasPages	0,000	0,750	0,556	0,107	0,444	0,000	0,000	0,000	0,000	0,625	0,625	0,000	0,000	0,333	0,688	0,000	0,000
Manual :: Unpublished	0,000	0,545	0,667	0,129	0,468	0,000	0,000	0,000	0,000	0,691	0,691	0,000	0,000	0,333	0,500	0,000	0,000
MotionPicture :: Book	0,000	0,308	0,556	0,129	0,403	0,000	0,000	0,000	0,000	0,551	0,551	0,000	0,000	0,400	0,500	0,000	0,000
startPage :: hasAuthor	0,000	1,000	0,667	0,136	0,389	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,175	0,500	0,000	0,000
price :: hasPublisher	0,000	0,417	0,556	0,129	0,375	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,440	0,500	0,000	0,000
MotionPicture :: Conference	0,000	0,769	0,667	0,172	0,433	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,260	0,615	0,000	0,000
series :: hasNote	0,000	0,857	0,667	0,100	0,417	0,000	0,000	0,000	0,000	0,640	0,640	0,000	0,000	0,267	0,500	0,000	0,000
firstPublished :: hasEditor	0,000	0,643	0,694	0,172	0,403	0,000	0,000	0,000	0,000	0,466	0,466	0,000	0,000	0,244	0,500	0,000	0,000
series :: hasYear	0,000	0,857	0,694	0,100	0,514	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,267	0,500	0,000	0,000
numberOrVolume :: hasLCCN	0,000	0,500	0,556	0,188	0,278	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,086	0,500	0,000	0,000
url :: hasInstitution	0,000	0,214	0,444	0,129	0,389	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,333	0,500	0,000	0,000
Misc :: Incollection	0,000	0,333	0,556	0,122	0,403	0,000	0,000	0,000	0,000	0,444	0,444	0,000	0,000	0,250	0,542	0,000	0,000
firstPublished :: hasMrnumber	0,000	0,786	0,556	0,188	0,346	0,000	0,000	0,000	0,000	0,637	0,637	0,000	0,000	0,146	0,636	0,000	0,000
Booklet :: Incollection	0,000	0,583	0,667	0,144	0,512	0,000	0,000	0,000	0,000	0,635	0,635	0,000	0,000	0,343	0,500	0,000	0,000
note :: hasContents	0,000	0,364	0,556	0,116	0,590	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,500	0,545	0,000	0,000

Figura 15 - Resultado da Execução com o Retorno das Métricas de Similaridades

Ainda há a possibilidade da visualização do valor de força retornado por cada métrica de similaridade para cada par de entidades das ontologias, como é mostrado na Figura 15. O objetivo principal do protótipo é a geração dos dados com a combinação das diferentes métricas de similaridade, que servem de entrada para o processo de mineração de dados, necessário para a obtenção do classificador.

4.3. Planejamento dos Experimentos

Para validar a proposta serão definidos 5 cenários de experimentos que guiarão a escolha da melhor combinação dos grupos de métricas a serem utilizadas na criação do

classificador. Como se trabalhará com os grupos de métricas: 1 - Baseada em *Strings*; 2 - Baseada em Linguagem; 3 - Baseada em Recursos Linguísticos; e 4 - Baseado em Análise de Dados e Estatística, cinco cenários de experimentos serão elaborados combinando esses grupos de métricas para avaliar a melhor forma de combiná-los para a geração do classificador. A Tabela 7, que está no Apêndice I, traz todo o planejamento de todos os cenários, domínios dos experimentos e algoritmos de aprendizado utilizados para a avaliação da abordagem proposta. Os experimentos foram elaborados considerando a variação dos grupos de métricas, e em todos os cenários foram utilizados os algoritmos de aprendizado e os domínios da OAEI. A Figura 16 traz a arquitetura de como os experimentos foram elaborados. No cenário #1 foi experimentada a métrica baseada em recurso linguístico, com todos domínios: *Benchmark*, *Conference* e *MultiFarm,* e com todos os algoritmos de aprendizado: *RF, MLP* e *SVM.*

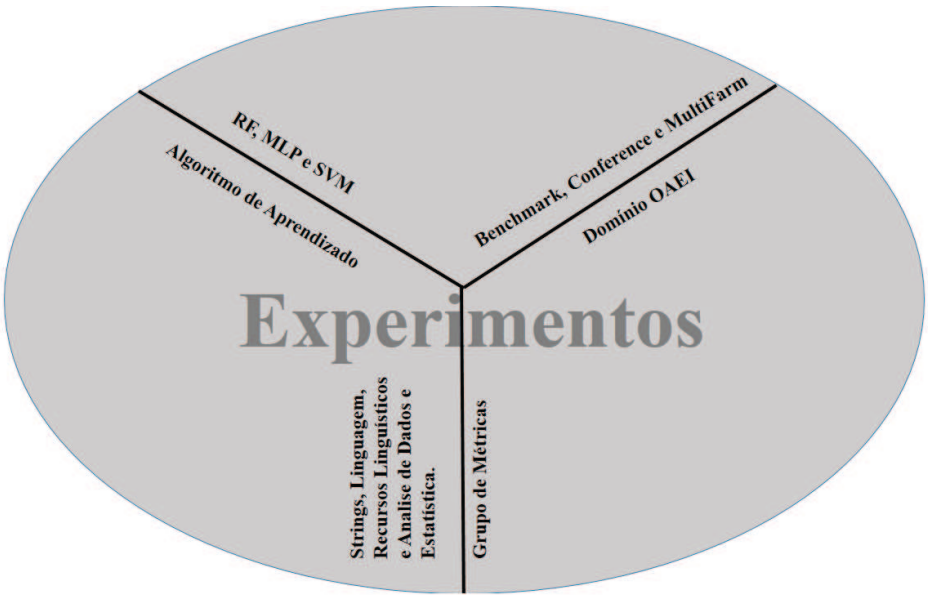


Figura 16 - Arquitetura dos Experimentos Planejados

Já no cenário #2 foi realizado o experimento com o grupo de métricas baseada em linguagem, com o todos domínios: *Benchmark*, *Conference* e *MultiFarm,* e com todos os algoritmos de aprendizado: *RF, MLP* e *SVM.* No cenário #3 foi realizado o experimento

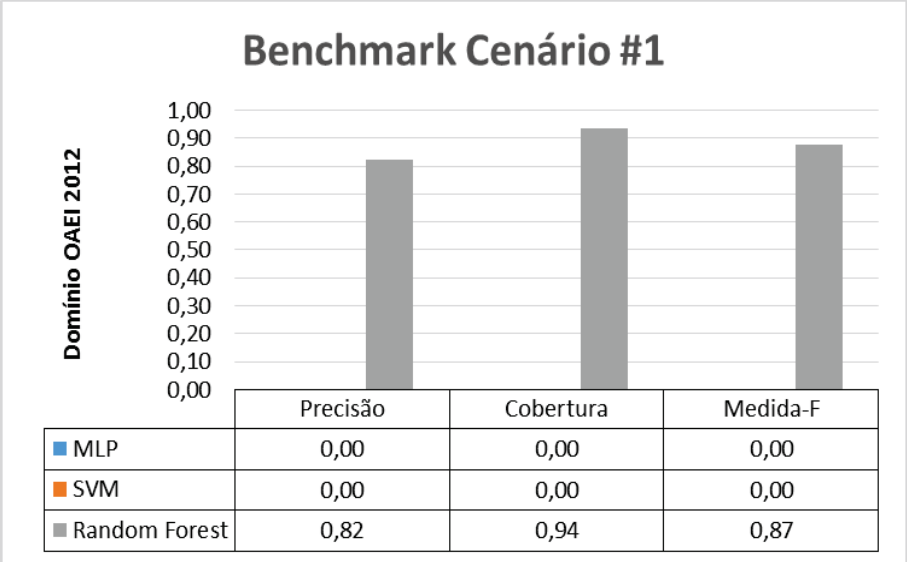
com o grupo de métricas baseada em *String*, com o todos domínios: *Benchmark*, *Conference* e *MultiFarm,* e com todos os algoritmos de aprendizado: *RF, MLP* e *SVM*. No cenário #4 foi realizado o experimento o grupo de métricas baseada em Análise de Dados e Estatística, com o todos domínios: *Benchmark*, *Conference* e *MultiFarm,* e com todos os algoritmos de aprendizado: *RF, MLP* e *SVM*. E no cenário #5 foi realizado o experimento com a combinação de todos os grupos de métricas: *Strings*, Linguagem, Recursos Linguísticos e Análise de Dados e Estatística, com o todos domínios: *Benchmark*, *Conference* e *MultiFarm,* e com todos os algoritmos de aprendizado: *RF, MLP* e *SVM*. O Objetivo é verificar como se comporta cada grupo de métrica isoladamente, e outro cenário onde os grupos de métricas de similaridades foram combinados para verificar se a combinação de diversos grupos de métricas tem um desempenho melhor comparado aos grupos de métricas de forma isolada.

A ferramenta de KDD, Weka[11], foi utilizada para apoiar os experimentos. A escolha da Ferramenta Weka se deve ao fato de estar disponível para uso (licença GPL) e ser amplamente utilizada em trabalhos científicos [76], [16], [72]. O software foi escrito na linguagem Java e contém uma GUI (*Graphical User Interface*) para interagir com arquivos de dados e produzir resultados visuais. E também dispõe de uma grande variedade de algoritmos de aprendizado implementados na ferramenta, como os que forma selecionados para o experimentos desse trabalho. A seguir serão descritos todos os parâmetros configuráveis utilizados em cada algoritmo utilizados. Os parâmetros dos algoritmos de aprendizado utilizados foram os padrões do Weka, exceto o parâmetro do *kernel* do *SVM*, que foi alterado para *RFBKernel*, pois segundo indicação do trabalho de [51] as demais são variações da *RFBKernel*.

[11]http://www.cs.waikato.ac.nz/ml/weka/.

4.4. Realização do Experimento - Cenário #1

Neste primeiro cenário foi utilizado a métrica baseada em Recurso Linguístico, com todos os domínios selecionados da OAEI 2012 e todos os algoritmos de aprendizados utilizado na abordagem. A métrica de similaridade utilizada foi o *JWNL Alignment*[12] disponível na API de Alinhamento [37]. Os resultados em relação a Medida-F, Cobertura e Precisão no domínio Benchmark são apresentado na Figura 17 de acordo com os algoritmo de aprendizados.

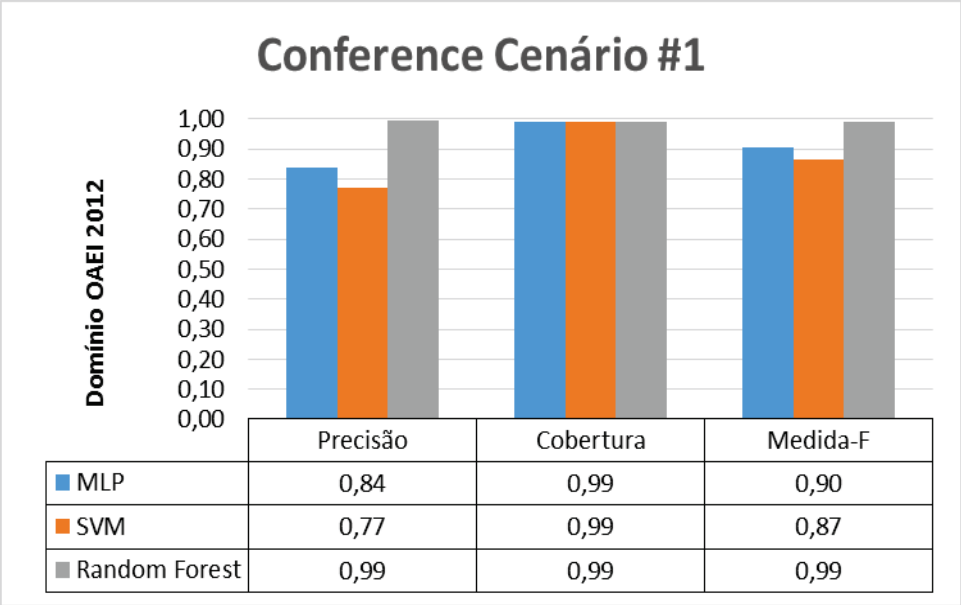


	Precisão	Cobertura	Medida-F
MLP	0,00	0,00	0,00
SVM	0,00	0,00	0,00
Random Forest	0,82	0,94	0,87

Figura 17 – Resultados do Domínio *Benchmark* no Cenário #1

Percebe-se que para o domínio *Benchmark* os resultados do algoritmo *RF* foram os melhores em todas as medidas de qualidade. E os algoritmos *MLP* e *SVM* obtiveram resultados ruins em todas as medidas avaliadas.

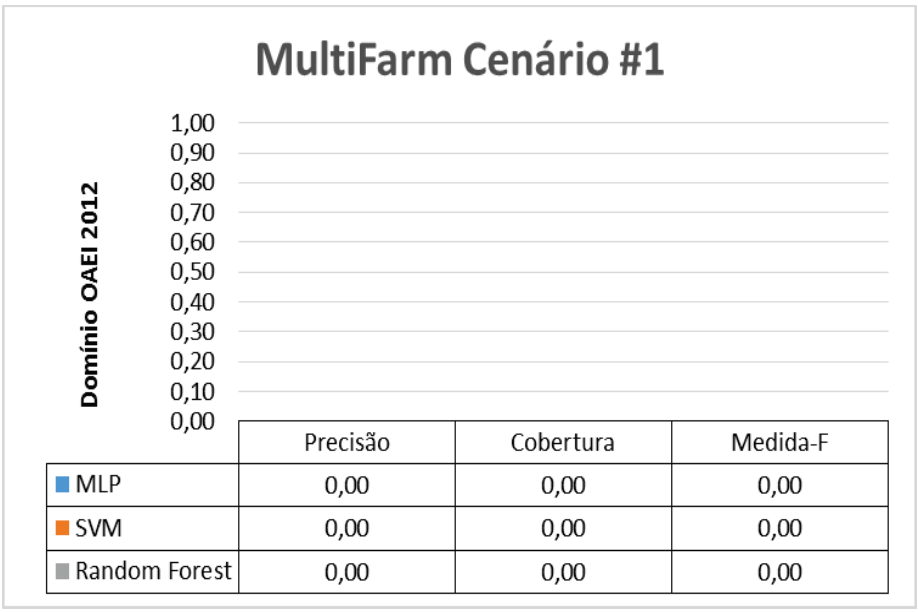
[12] JWNL é uma API para acessar o dicionário léxico WordNet em vários formatos, bem como a descoberta de relacionamento e processamento morfológico. É compatível com as versões WordNet 2.0 a 3.0.



	Precisão	Cobertura	Medida-F
MLP	0,84	0,99	0,90
SVM	0,77	0,99	0,87
Random Forest	0,99	0,99	0,99

Figura 18 - Resultados da Domínio *Conference* no Cenário #1

Analisando os resultados do domínio *Conference* na Figura 18, os resultados do algoritmo *RF* foram os melhores em todas as medidas de qualidade com valores próximo a 1. E os algoritmos *SVM* e *MLP* obtiveram valores muito próximos com uma pequena vantagem para o algoritmo *MLP* na medida de Precisão. E na Medida-F novamente houve uma pequena vantagem para o algoritmo *MLP*



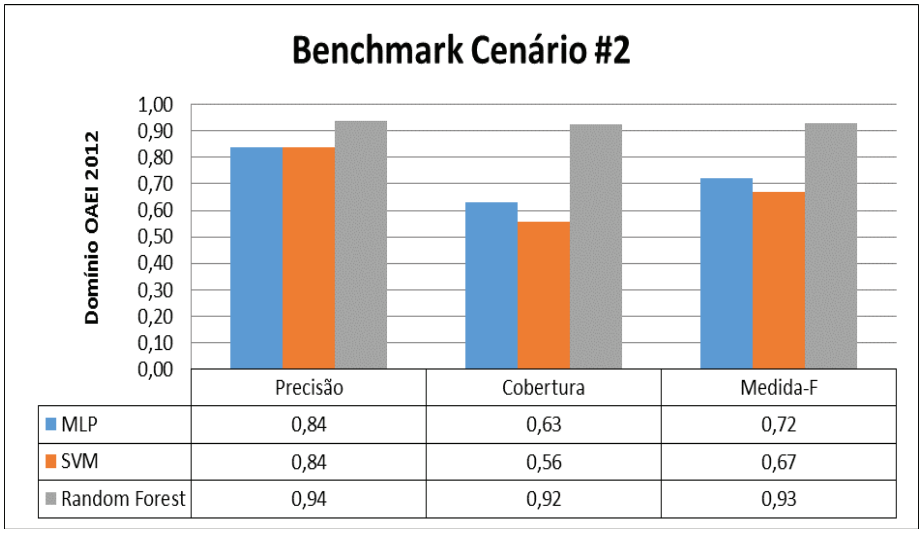
	Precisão	Cobertura	Medida-F
MLP	0,00	0,00	0,00
SVM	0,00	0,00	0,00
Random Forest	0,00	0,00	0,00

Figura 19 - Resultados do Domínio *MultiFarm* no Cenário #1

No domínio *MultiFarm* os resultados em todos os algoritmos foram muito parecidos como mostra a Figura 19. Todos os algoritmos tiveram uma performance ruim com valores de 0,00. Vale ressaltar que neste cenário só foi utilizada uma métrica, a baseada em Recurso Linguístico com o Wordnet, que é um grande dicionário léxico do Inglês, o que pode ter ocasionado a baixa performance nos resultados. Já que no domínio *MultiFarm* a tarefa de alinhamento é formulada com ontologias em diferentes idiomas. No presente trabalho foram utilizadas para a tarefa de alinhamento neste domínio e em todos os cenários as ontologias no idioma Inglês e Português.

4.5. Realização do Experimento - Cenário #2

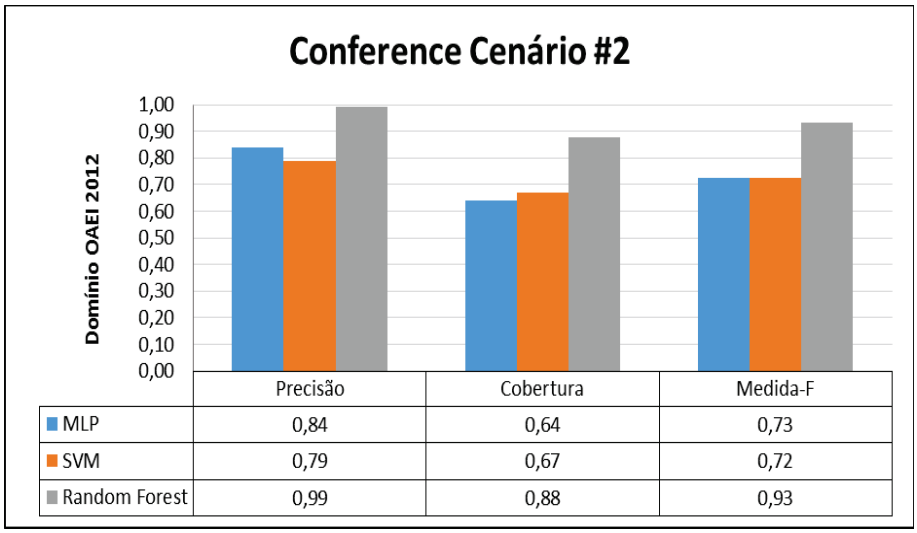
As métricas de similaridade utilizadas foram as baseadas em linguagem: *QGramDistance*, *ChapmanMatchingSoundex*, *SmithWatermanGotohWindowedAffine*, *Soundex, ChapmanOrderedNameCompoundSimilarity* e *TagLinkToken*. Os resultados do domínio *Benchmark* em relação a Medida-F, Precisão e Cobertura são apresentado por algoritmo de aprendizado na Figura 20.



	Precisão	Cobertura	Medida-F
MLP	0,84	0,63	0,72
SVM	0,84	0,56	0,67
Random Forest	0,94	0,92	0,93

Figura 20 - Resultados do Domínio *Benchmark* no Cenário #2

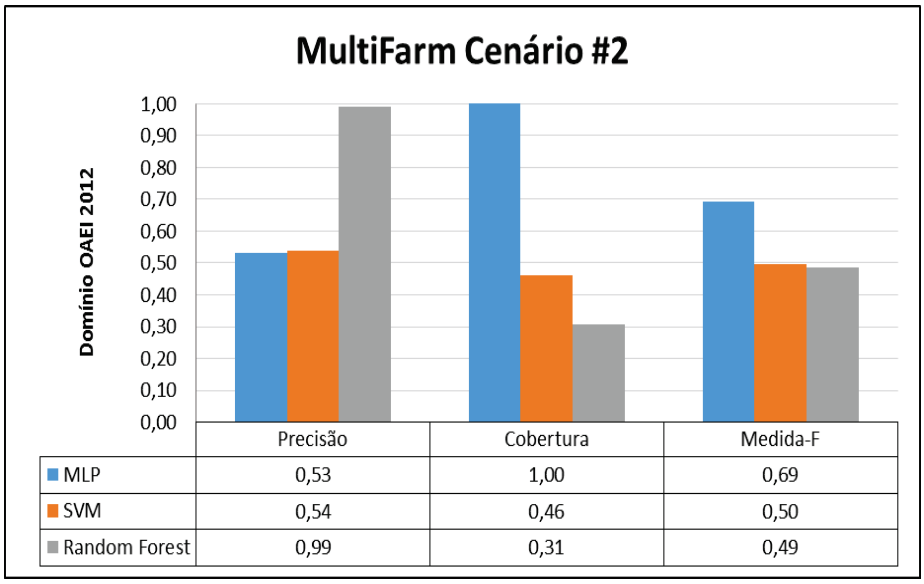
Os resultados em todas as medidas de qualidade foram melhores para o algoritmos de aprendizado *RF* em que os valores ficaram entre 0,92 e 0,94 em todas as medidas. Em seguida o algoritmo *MPL* obteve a segunda melhor colocação com algoritmo *SVM* tendo o pior desempenho em relação aos demais. Os resultados com as combinações da métricas baseadas em linguagem para o domínio *Benchmark* foram bem próximo ou igual ao valor ótimo, para o algoritmos de aprendizado *RF*.



	Precisão	Cobertura	Medida-F
MLP	0,84	0,64	0,73
SVM	0,79	0,67	0,72
Random Forest	0,99	0,88	0,93

Figura 21 - Resultados do Domínio *Conference* no Cenário #2

Para o domínio *Conference* a performance dos algoritmo de aprendizado foi parecida com a do domínio *Benchmark* como mostra a Figura 21. Onde os valores ficaram bem próximos ao domínio *Benchmark* na mesma proporção em relação aos algoritmos de aprendizado. O *RF* teve perto do ideal (= 1) e o *MLP* ficou um pouco melhor em relação ao *SVM*.



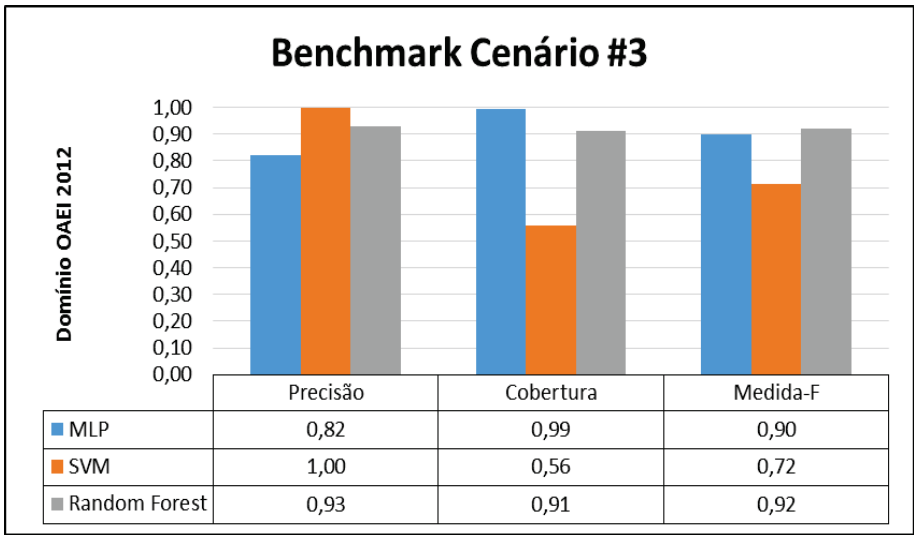
	Precisão	Cobertura	Medida-F
MLP	0,53	1,00	0,69
SVM	0,54	0,46	0,50
Random Forest	0,99	0,31	0,49

Figura 22 - Resultados do Domínio *MultiFarm* no Cenário #2

A Figura 22 traz os resultados em que o melhor algoritmo de aprendizado no domínio *MultiFarm* foi o *MLP,* diferente dos outros domínios onde o algoritmo *RF* obteve a melhor performance. Neste domínio, diferentemente dos demais os algoritmos *MLP* e *SVM* foram superiores ao algoritmo *RF* em relação a Medida-F.

4.6. Realização do Experimento - Cenário #3

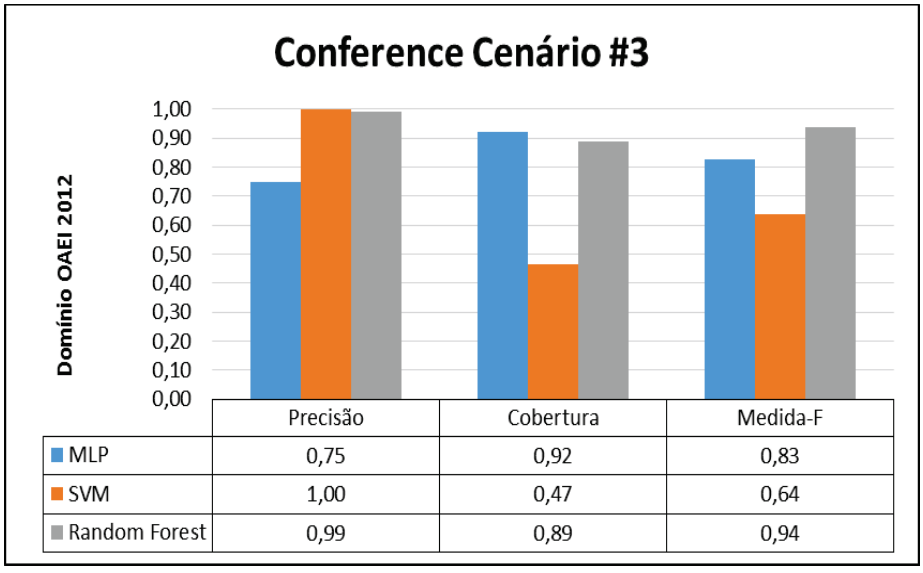
As métricas de similaridade utilizadas neste cenário foram as baseadas em *String*: *Levenshtein*, *SmithWaterman, SmithWatermanGotoh*, *JaroWinker, Euclidean, ChapmanLengthDeviation, BlockDistance, Jaro, ChapmanMeanLength, CosineSimilarity, NeedlemanWunch, MatchingCoefficient, MongeElkan* e *Jaccard*. Os resultados do domínio *Benchmark* em relação a Medida-F, Precisão e Cobertura são apresentado por algoritmo de aprendizado na Figura 23.



	Precisão	Cobertura	Medida-F
MLP	0,82	0,99	0,90
SVM	1,00	0,56	0,72
Random Forest	0,93	0,91	0,92

Figura 23 - Resultados do Domínio *Benchmark* no Cenário #3

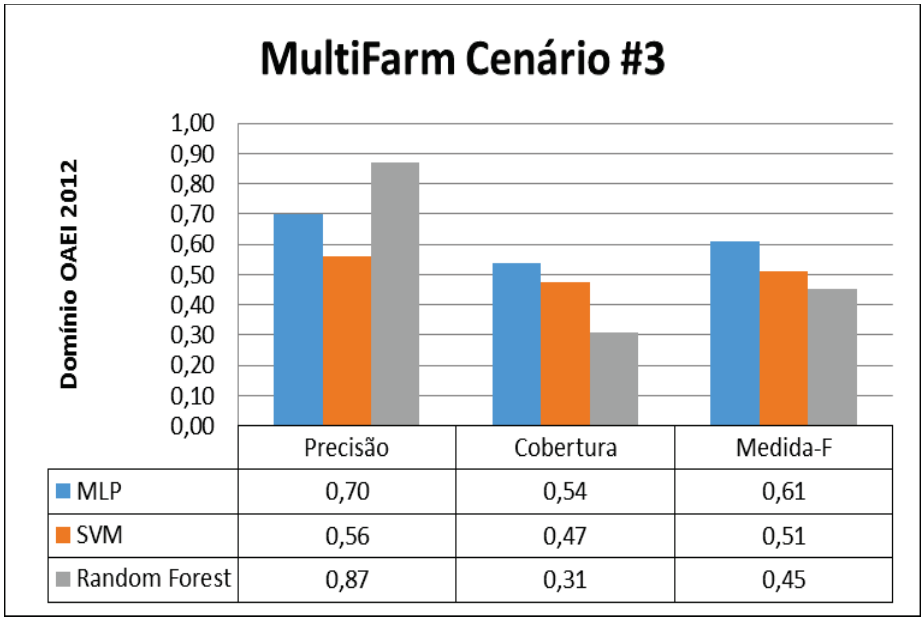
A performance do RF foi superior aos demais algoritmos de aprendizado na Medida-F para este domínio, com uma uniformidade em todas as medidas de qualidade. E novamente o *MLP* ficou com a segunda melhor colocação, e o *SVM* o pior resultado entre os algoritmos experimentados.



	Precisão	Cobertura	Medida-F
MLP	0,75	0,92	0,83
SVM	1,00	0,47	0,64
Random Forest	0,99	0,89	0,94

Figura 24 - Resultados do Domínio *Conference* no Cenário #3

Novamente o algoritmo *RF* obteve os melhores resultados em todas as medidas de qualidade com a mesma uniformidade já demonstrada. Comparando o *MLP* e *SVM*. O *MLP* ficou melhor em relação ao *SVM*.



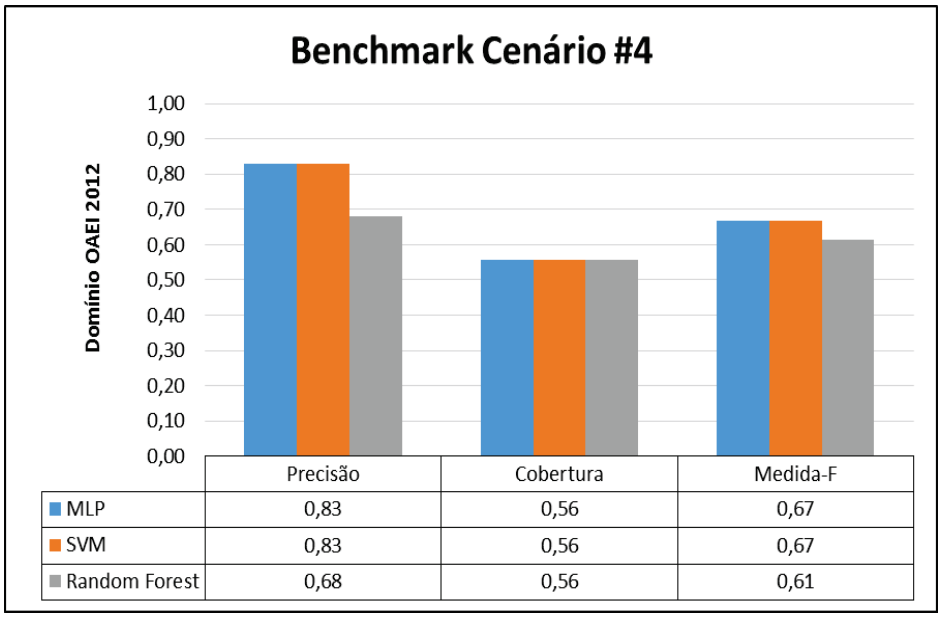
	Precisão	Cobertura	Medida-F
MLP	0,70	0,54	0,61
SVM	0,56	0,47	0,51
Random Forest	0,87	0,31	0,45

Figura 25 - Resultados do Domínio *MultiFarm* no Cenário #3

Neste domínio o *RF* foi inferior aos demais algoritmos, obtendo o pior resultado na Medida-F, e comparando o *MLP* e *SVM*. O *MLP* ficou novamente um pouco melhor em relação ao *SVM*.

4.7. Realização do Experimento - Cenário #4

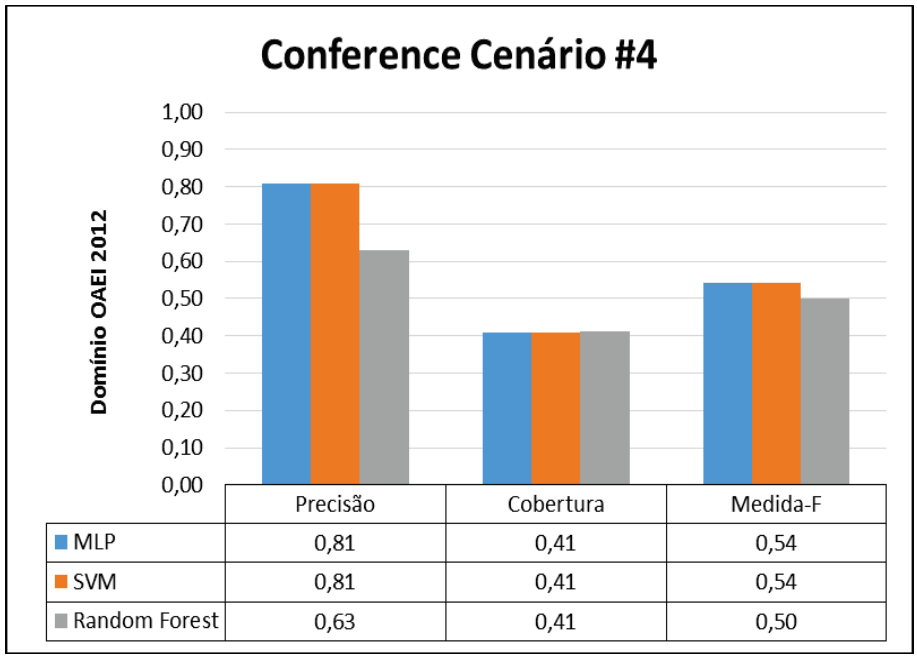
As métricas de similaridade utilizadas neste cenário foram as baseadas em análise da dados e estatística: *DiceSimilarity* e *OverlapCoefficient*. Os resultados do domínio *Benchmark* em relação a Medida-F, Precisão e Cobertura são apresentado por algoritmo de aprendizado na Figura 26.



	Precisão	Cobertura	Medida-F
MLP	0,83	0,56	0,67
SVM	0,83	0,56	0,67
Random Forest	0,68	0,56	0,61

Figura 26 - Resultados do Domínio *Benchmark* no Cenário #4

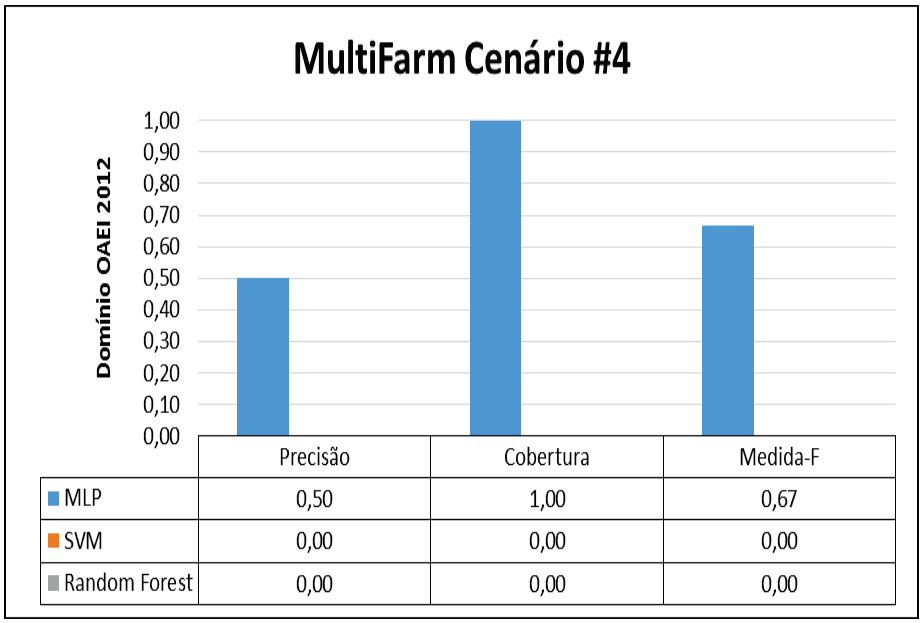
Neste cenário no domínio *Benchmark* os algoritmos de aprendizados obtiveram os resultados muitos parecidos. Os algoritmos *MLP* e *SVM* obtiveram os mesmos valores nas três medidas de qualidade, e o *RF* ficou 0,06 abaixo na Medida-F em relação aos dois primeiros.



	Precisão	Cobertura	Medida-F
MLP	0,81	0,41	0,54
SVM	0,81	0,41	0,54
Random Forest	0,63	0,41	0,50

Figura 27 - Resultados do Domínio *Conference* no Cenário #4

Como no domínio anterior os algoritmos *MLP* e *SVM* obtiveram os mesmos valores nas três medidas de qualidade neste cenário. O *RF* apesar de ter ficado com valor de precisão abaixo dos algoritmos *MLP* e *SVM*, porém na Medida-F obteve o valor 0,04 abaixo dos demais e o mesmo valor para a medida de cobertura.



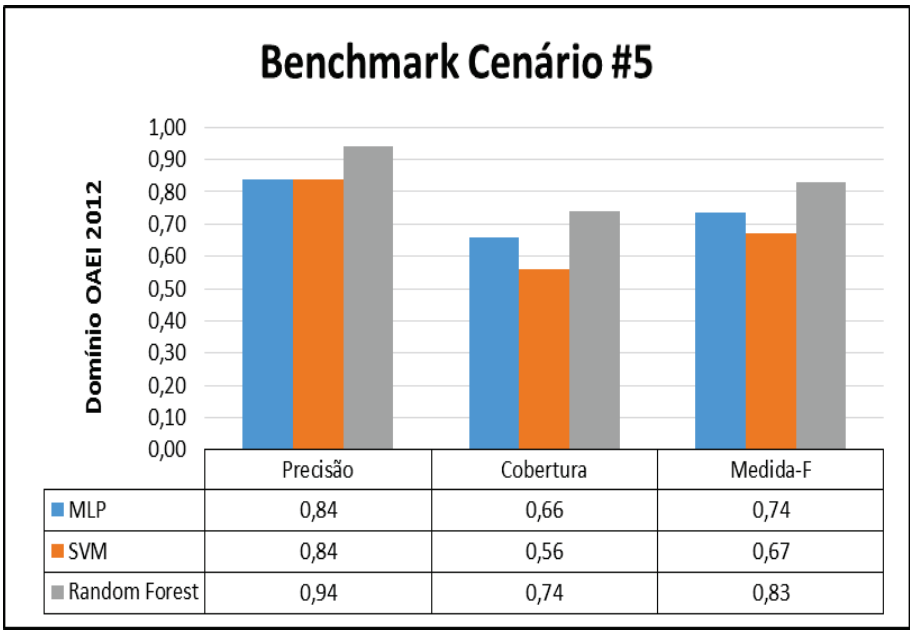
	Precisão	Cobertura	Medida-F
MLP	0,50	1,00	0,67
SVM	0,00	0,00	0,00
Random Forest	0,00	0,00	0,00

Figura 28 - Resultados do Domínio *MultiFarm* no Cenário #4

Mais uma vez no cenários *MultiFarm* o algoritmos *MLP* obteve o melhor resultado em todas as medidas de qualidade. Sendo que os algoritmos *SVM* e *RF* ficaram bem abaixo com 0,00 em todos os as medida de qualidade neste domínio.

4.8. Realização do Experimento - Cenário #5

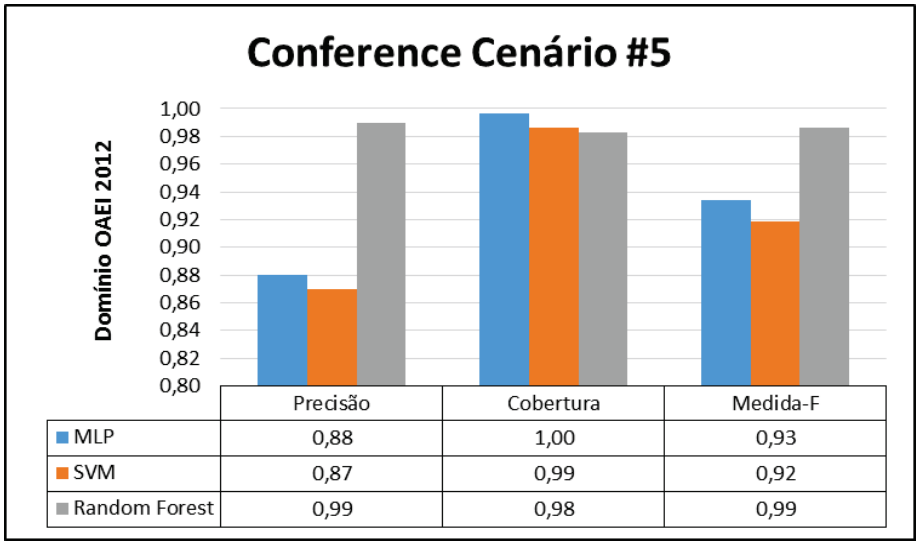
No cenário 5 foi utilizado a combinação de todos os grupos de métricas dos cenários #1, #2, #3 e #4, com todos os domínios selecionados da OAEI 2012 e todos os algoritmos de aprendizados utilizado na abordagem. Os resultados do domínio *Benchmark* em relação a Medida-F, Precisão e Cobertura são apresentado por algoritmo de aprendizado na Figura 29.



	Precisão	Cobertura	Medida-F
MLP	0,84	0,66	0,74
SVM	0,84	0,56	0,67
Random Forest	0,94	0,74	0,83

Figura 29 - Resultados do Domínio *Benchmark* no Cenário #5

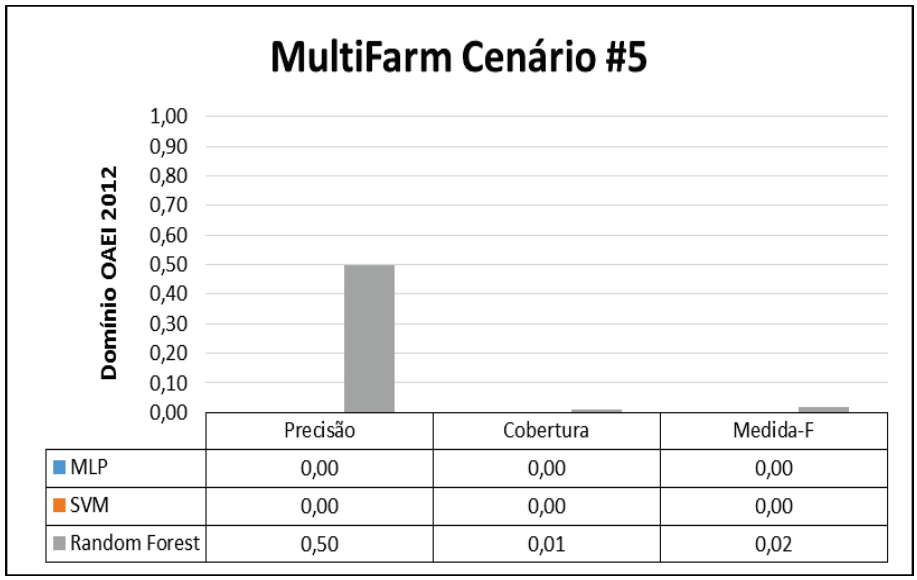
Diferente de outros domínios em comparação com os algoritmo *MLP* e *SVM*, o *RF* obteve a melhor performance em todas as medidas de qualidade. O *MLP* ficou em segundo com valor de 0,74 para a Medida-F com o *SVM* ficando um pouco abaixo com 0,67 para a mesma medida.



	Precisão	Cobertura	Medida-F
MLP	0,88	1,00	0,93
SVM	0,87	0,99	0,92
Random Forest	0,99	0,98	0,99

Figura 30 - Resultados do Domínio *Conference* no Cenário #5

No domínio *Conference* os resultados foram parecidos com os do *Benchmark* em comparação com os algoritmos de aprendizado para este cenário. O *RF* novamente foi o melhor em todas as medidas de qualidade com valores próximo a 1 para todas. E em seguida o *MLP* que ficou um pouco acima que o *SVM* com um diferença de 0,01 para a Medida-F.



	Precisão	Cobertura	Medida-F
MLP	0,00	0,00	0,00
SVM	0,00	0,00	0,00
Random Forest	0,50	0,01	0,02

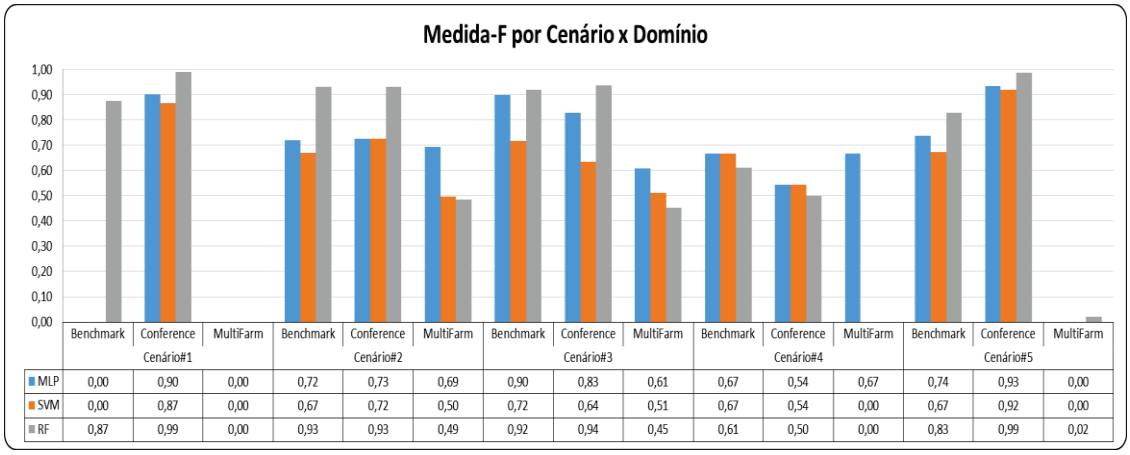
Figura 31 - Resultados do Domínio *MultiFarm* no Cenário #5

Já no domínio MultiFarm para este cenário os resultados foram ruins para todos os algoritmos, o *MLP* e *SVM* ficaram com 0,00 em todas as medidas de qualidade. *RF* foi o melhor entre os algoritmos de aprendizado com valor 0,50 na Precisão e 0,01 para Cobertura, porém a Medida-F ficou muito baixa com um valor 0,02.

4.9. Considerações Finais

Um aspecto importante verificado durante os experimentos é que a combinação dentro do próprio grupo de métricas como o grupo de métrica baseado em Linguagem e o baseado em *String*, já retornam resultados satisfatórios acima de 0,9 para o algoritmo *RF* nos domínios *Benchmark* e *Conference*, e os melhores resultados para o domínio *MultiFarm*

com o algoritmo *MLP*, como mostra a Figura 32. O que faz com que os resultados da combinação dentro desses grupos de métricas de similaridade tenham desempenho melhores que combinação de todos os grupos de métricas, que foi a que obteve o melhor resultado para o domínio *Conference*. Inclusive, o *RF* obteve resultados melhores que as ferramentas participantes da OAEI 2012 nos domínios experimentados (*Benchmark*, *Conference*) e o *MLP* obteve resultados melhores que as ferramentas participantes para o domínio *MultiFarm*.



	Benchmark	Conference	MultiFarm	Benchmark	Conference	MultiFarm	Benchmark	Conference	MultiFarm	Benchmark	Conference	MultiFarm	Benchmark	Conference	MultiFarm
	Cenário#1			Cenário#2			Cenário#3			Cenário#4			Cenário#5		
MLP	0,00	0,90	0,00	0,72	0,73	0,69	0,90	0,83	0,61	0,67	0,54	0,67	0,74	0,93	0,00
SVM	0,00	0,87	0,00	0,67	0,72	0,50	0,72	0,64	0,51	0,67	0,54	0,00	0,67	0,92	0,00
RF	0,87	0,99	0,00	0,93	0,93	0,49	0,92	0,94	0,45	0,61	0,50	0,00	0,83	0,99	0,02

Figura 32 - Resultados dos Experimentos

Comparando Athenas com as médias dos domínios da OAEI 2012, o desempenho é superior com quase todos os grupo de métricas utilizado, como mostra a Figura 33. O que ratifica a abordagem como promissora para o propósito de tentar resolver o problema da heterogeneidade semântica.

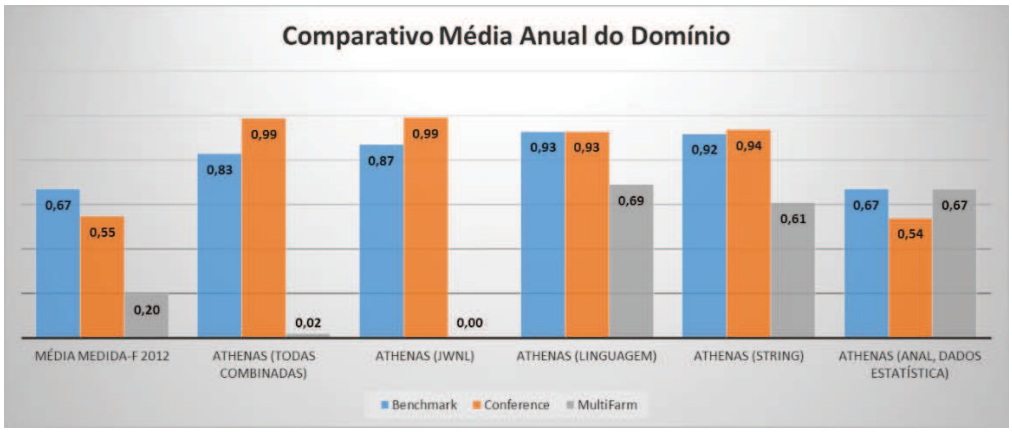


Figura 33 - Comparativo Athenas x Média Medida-F 2012 dos Domínios

Diante de todos os dados coletados durantes os experimentos, verificou-se que a hipótese foi confirmada, pois a combinação de métricas de similaridades na maioria dos domínios experimentados obteve os melhores resultados nas mediadas de qualidade. Apenas na combinação de todas as métricas no domínio *MultiFarm* ficou abaixo da média anual, a métrica baseada em recursos linguístico (*JWNL*) também ficou abaixo, mas ai apenas uma métrica é utilizada, então não é feita a combinação.

Capítulo 5 - Trabalhos Relacionados

Esta dissertação relaciona-se com trabalhos que, independente de técnica, usam algumas características similares à abordagem proposta, como a combinação de métricas de similaridade, o emprego de Aprendizado de Máquina na tarefa de alinhamento de ontologia, e também a utilização das ontologias pertencentes aos domínios da OAIE 2012 para realizar o alinhamento.

5.1. Descrição dos Trabalhos Relacionados

Na tese de DuyHoa Ngo [16] foi proposta uma nova abordagem de alinhamento que combina diferentes técnicas de AM, com a métrica de alinhamento baseado em grafo e recuperação de informação, com o objetivo de melhorar o alinhamento de ontologia. É feito uso de técnicas de recuperação de informação para projetar uma nova medida de similaridade efetiva para comparar os *labels* e os perfis de contexto de entidades em nível de elemento. Também é aplicado um método de alinhamento baseado em grafo chamado propagação de semelhança no nível da estrutura que efetivamente descobre mapeamentos, explorando informações estruturais das entidades nas ontologias de entrada. Em termos de combinação de medidas de similaridade no nível de elemento, é transformada a tarefa de alinhamento de ontologia em uma tarefa de classificação em AM. Além disso, propõe-se um método de soma ponderada dinâmica para combinar automaticamente os resultados dos alinhamentos obtidos a partir das entidades e dos alinhamentos em nível de estrutura. A fim de remover mapeamentos inconsistentes, foi implementado um novo método de filtragem semântica. Finalmente, para lidar com a tarefa de alinhamento em ontologias de larga escala, são propostos dois métodos de seleção de candidatos para reduzir o espaço

computacional. Todas as contribuições foram implementadas em um protótipo chamado YAM ++. Para avaliar a abordagem, adotaram várias trilhas dos domínios *Benchmark, Conference, Multifarm, Anatomy, Library and Large Biomedical Ontologies* da campanha OAEI. Os resultados dos experimentos mostraram que a proposta de alinhamento é promissora. Em comparação a outros participantes na campanha da OAEI, YAM ++ mostrou se altamente competitivo e ganhou uma alta posição no *ranking*.

No trabalho de NEZHADI *et al.* [53] é apresentado um método para combinar medidas de similaridade de diferentes categorias, são utilizados 15 métricas de similaridades baseados em *String*, lingüística e estrutura, sem as instâncias das ontologias. Para alinhar diferentes ontologias de forma eficiente, *K NearestNeighbor* (*KNN*), *Support Vector Machine* (*SVM*), *Árvore de Decisão* (*DT*) e classificadores *AdaBoost* são investigados. Cada classificador é otimizado com base no menor custo e melhor taxa de classificação. O modelo proposto determina o processo de alinhamento sem a necessidade prévia de instâncias de ontologias, através de um processo de otimização completa de parâmetros operacionais. O que segundo o autor, facilita a tarefa de alinhamento.

Optima [55] exibe as ontologias separadamente em componentes de visualização de grafos e, consequentemente, foca no problema de alinhamento em grafo. Optima modela o problema de alinhamento em grafo como uma busca local no espaço de correspondências candidatas e utiliza o GEM (*Generalized Expectation-Maximization*) [15] iterativamente para orientar na busca. Dentro da abordagem GEM, é usado tanto, a semelhança estrutural entre os grafos das ontologias quanto a similaridade léxica entre os *labels* dos conceito e as instâncias. Semelhante a algumas das abordagens recentes, os resultados gerados entre as ontologias são inexatos, o que podem resultar em vários nós de uma ontologia sendo mapeados para um único nó da outra. O alinhamento inexato é o processo de encontrar uma melhor correspondência possível entre os dois grafos quando

a correspondência exata não é possível ou é computacionalmente difícil. Os alinhamentos dos nós são destacados e exibidos em azul e o usuário pode selecionar um nó para identificar o nó correspondente na outra ontologia.

Em CODI (Otimização Combinatória para Integração de Dados) [35] são aproveitadas as estruturas terminológica para alinhamento de ontologias. A implementação atual produz mapeamentos entre conceitos, relacionamentos e indivíduos. O sistema combina medidas de similaridade léxicas com informações sobre o esquema para evitar completamente a incoerência e inconsistência durante o processo de alinhamento. CODI é baseado em sintaxe e semântica da lógica de Markov [14] e transforma o problema de alinhamento para um problema de otimização máxima-a-posteriori. Este problema precisa de valores a priori de confiança para cada hipótese de correspondências como entrada, conhecido também como: "*anchor-alignments"*. Foi implementado um método de agregação de diferentes medidas de similaridade. Outro recurso do CODI é o reconhecimento dos pares de ontologias pertencentes a diferentes versões da mesma ontologia. Em alinhamento de instâncias CODI não computa semelhanças lexicais para todos os pares existentes de casos, mas utiliza afirmações da propriedade de objeto para reduzir as comparações necessárias, com esta técnica é possível a redução do tempo para alinhamento de ontologias de larga escala como a *Anatomy e Library*.

AROMA (*Association Rule ontology Matching Approach*) [20] [60] é uma abordagem híbrida de alinhamento extensional e assimétricos concebidos para descobrir relações de equivalência e subsunção entre entidades, isto é, classes e propriedades, a partir de duas taxonomias textuais (diretórios web ou ontologias OWL). A abordagem faz uso do paradigma da regra de associação [58], e de uma medida de estatística. AROMA baseia-se na seguinte suposição: Uma entidade A será mais específica, ou equivalente a

uma entidade B, se o vocabulário (isto é, condições e também dados) utilizado para descrever um dos seus descendentes e as suas instâncias tende a ser incluída na entidade B. AROMA está dividido em três fases principais sucessivas: (1) A fase de pré processamento representa cada entidade, ou seja, classes e propriedades, por um conjunto de termos, (2) a segunda fase consiste na descoberta de regras de associação entre as entidades, e, finalmente, (3) a fase de pós-processamento destina-se a limpeza e a melhora do alinhamento resultante. A primeira fase constrói um conjunto de termos relevantes e / ou valores de dados para cada classe e propriedade. São extraídos os vocabulários de classes e de propriedades de suas anotações e valores individuais com o auxílio de um extrator de termos aplicado ao texto resultante. A segunda etapa AROMA descobre as relações de subsunção usando o modelo de regras de associação e a medida de implicação da intensidade [59]. No contexto de AROMA, uma regra de associação a → b representa uma quase-implicação (ou seja, uma implicação permitindo alguns contraexemplos) do vocabulário da entidade *a* no vocabulário da entidade *b*. Essa regra pode ser interpretada como uma relação de subordinação a entidade antecedente para o consequente. Por exemplo, carro → veículo significa: "O conceito carro é mais específico do que o conceito veículo". A regra do algoritmo de extração tira vantagem da estrutura de ordem parcial fornecida pela relação de subsunção, e uma propriedade da intensidade de incidência de poda no espaço de busca. A última etapa refere-se ao pós-processamento das regras da associação definidas.

Hertuda [69] é a primeira idéia de um *matcher* baseado em elemento com comparação de *String*s. Ele gera apenas alinhamentos homogêneos, que são compatíveis com OWL Lite / DL. Isso significa que classes, propriedades de dados e propriedades do objeto são tratados separadamente. Como resultado, existem três limites que podem ser definidos de forma independente. Um para classe em classe, objeto para propriedade do

objeto e dados para propriedade de dados. Um limite global simples define todos os sublimites para o mesmo valor. Sobre todos os conceitos um produto cruzado é computado. Para cada conceito todos os rótulos, comentários e fragmentos URI são extraídos. Então estes termos formam um conjunto. Para comparar dois conceitos, respectivamente, os conjuntos de termos, cada um dos elementos do primeiro conjunto é comparado com cada um dos elementos do segundo conjunto. O melhor valor é a medida de similaridade para esses conceitos. Uma etapa de pré-processamento para comparação dos termos *Camel Case*[13] ou termos sublinhados ou com hifens, são divididas em tokens e simples convertidos em letras minúsculas. Portanto *writePaper*, *write-paper* e *write* tudo irá resultar em dois tokens, ou seja, *write* e *paper*. Depois uma matriz de similaridade é calculada com a distância *Damerau Levenshtein*. A média dos mapeamentos melhores são, então, devolvidos como a semelhança entre dois conjuntos de tokens. O sistema de alinhamento final contém uma abordagem alinhamento baseado em *Strings* e um filtro para a remoção de alinhamentos que não são considerados no alinhamento de referência. O filtro remove todos os alinhamentos que são verdadeiros, mas não estão no alinhamento de referência. Os mapeamentos são retirados principalmente de ontologias de Topo como *dublin core* ou *friend of a friend (FoF)*.

LogMap [39], [38] é um sistema de alinhamento de ontologia altamente escalável com o raciocínio embutido e capacidade de reparação de inconsistência. LogMap também suporta (em tempo real) interação com o usuário durante o processo de alinhamento, que é essencial para os casos em que requerem mapeamentos muito precisos. LogMap é o único sistema de alinhamento que (1) pode eficientemente alinhar ontologias semanticamente ricas contendo dezenas (e até centenas) de milhares de classes, (2)

[13] CamelCase é a denominação em inglês para a prática de escrever palavras compostas ou frases, onde cada palavra é iniciada com Maiúsculas e unidas sem espaços.

incorpora raciocínio sofisticado e técnicas de reparo para minimizar o número de inconsistências lógica, e (3) fornece suporte para a intervenção do usuário durante o processo de alinhamento. LogMap também está disponível como uma variante "leve" chamado LogMapLt, que essencialmente ignora todas as etapas de indexação do raciocínio, reparação e semântica. Devido à sua simplicidade, escalabilidade e razoável qualidade de sua produção, LogMapLt foi adotado como referência em alguns domínios OAEI.

GOMMA [41] é uma infraestrutura genérica para gerenciamento e análise de ontologias de ciências da vida e sua evolução. GOMMA utiliza um repositório genérico para gerenciar de maneira uniforme e eficiente versões de ontologia e diferentes tipos de mapeamentos. Além disso, fornece componentes para alinhamento de ontologia, e controle de mudanças evolutivas da ontologia. As principais funções incluem um dispositivo de armazenamento de versões genéricas da ontologia e mapeamentos, o apoio a alinhamento de ontologias e controle de mudanças nas ontologia. As funcionalidades suportadas para analisar mudanças da ontologia são úteis para avaliar o seu impacto sobre as aplicações dependentes da ontologia, e para o enriquecimento dos termos. GOMMA complementa a *Ontology Evolution Explorer* (OnEX), fornecendo funcionalidades para gerenciar várias versões de mapeamentos entre duas ontologias e permite combinar diferentes abordagens de alinhamento de ontologia.

A Tabela 5 tem o comparativo das características dos trabalhos relacionados que são relevantes para a análise da nossa contribuição. Em relação a combinação de métricas de similaridade apenas Aroma e Optima não utilizaram para a realização do alinhamento. Outra característica analisada foi a quantidade de métricas utilizada pelas abordagem que utilizaram a combinação de métricas de similaridade. Neste quesito apenas YAM++ com 16 e [53] com 15 se aproximaram da nossa abordagem que pode combinar até 23 métricas

de similaridade. Na análise dos domínios é importante salientar que alguns utilizados pelos trabalhos relacionados não eram interessantes para a nossa abordagem, como os domínios de *Anatomy, Library*, *Large Biology* e *Instance Matching.* Os três primeiros são específicos para ontologias de larga escala, onde as abordagem são muito especificas para este domínio, e o último para alinhamento de instâncias, que não foram consideradas no presente trabalho, porque nos domínios que foram utilizados as instâncias não são alinhadas. Podem ser utilizadas para ajudar no alinhamento das classes e relacionamentos.

Tabela 5 - Comparativo dos Trabalhos Relacionados

Ferramentas de Alinhamento	Combina métricas de similaridade?	Grupos de métricas de similaridade utilizados	Domínios utilizados	Necessita intervenção do usuário na tarefa de alinhamento?	Definição de um valor de *Threshold (limiar)* para o valor de força retornado
Abordagem Proposta - ATHENAS	Sim	String, Linguagem, Recurso Linguístico, Análise da Dados e Estatística	Benchmark, Comference e MultiFarm.	Não	Automático
YAM ++ [16]	Sim	String, Linguagem, Recurso Linguístico, Grafo e Modelo	Benchmark, Comference, Anatomy, MultiFarm, Library e Large Biology	Não	Manual
GOMMA [41]	Sim	String, Linguagem, Recurso Linguístico, Análise da Dados e Estatística e Repositorio de Estruturas	Benchmark, Comference, Anatomy, MultiFarm, Library e Large Biology	Não	Manual
LogMap [13, 62]	Sim	Linguagem, Recurso Linguístico, Reuso de Alinhamento e Modelo	Benchmark, Comference, Anatomy, MultiFarm, Library, Large Biology e Instance Macthing	Sim	Manual
[32]	Sim	String, Linguagem, Análise da Dados e Estatística	Benchmark (parcial)	Não	Automático
AROMA [20, 60]	Não	String e Taxomonia	Benchmark, Comference, Anatomy, MultiFarm, Library e Large Biology	Não	Manual
CODI [35]	Sim	String, Análise de Dados e Estatística e Modelo	Benchmark, Comference, Anatomy e Instance Macthing	Não	Manual
Optima [55]	Não	String, Grafo e Modelo	Benchmark, Comference, Anatomy, MultiFarm e Library	Sim	Automático
Hertuda [69]	Não	String, Linguagem e Ontologia de Topo	Benchmark, Comference, Anatomy, MultiFarm, Library e Large Biology	Sim	Manual

A necessidade da utilização de um alinhamento de referência foi analisada e apenas três trabalhos não necessitam para realização do alinhamento Optima, CODI e Aroma.

As outras características analisadas foram a necessidade de intervenção do usuário para a realização do alinhamento e se o sistema possuía uma GUI. Os trabalhos LogMap, Optima e Hertuda sofrem a intervenção do usuário durante o processo de alinhamento. Já os trabalhos de CODI, Aroma e [53] não possuem nos seus sistemas uma interface gráfica. Existem várias técnicas para calcular a agregação ideal para diferentes tipos de medidas de similaridade, tais como *Fuzzy*, soma ponderada, etc. [16], [53]. No entanto a escolha dos parâmetros ótimos destas técnicas, tais como limites (*threshold*) e outras restrições é uma tarefa de difícil definição para a imputação desse valor. Os trabalhos de Aroma, CODI, GOMMA, Hertuda, YAM ++ necessitam da definição de uma valor de *threshold* que nesses trabalhos variam entre 0.7 a 0.88. AM oferece a possibilidade de combinar diferentes medidas de similaridade automaticamente.

No presente trabalho, os algoritmos de AM supervisionados são utilizados para extrair o modelo ideal para combinação de métricas. Assim, o problema de alinhamento é transformado em uma tarefa de AM supervisionado. Esse é mais um diferencial da nossa abordagem em relação aos trabalhos correlatos. A definição do valor de *threshold* é definido automaticamente pelos algoritmos de AM supervisionados. Geralmente as ferramentas e métricas de similaridade retornam somente os pares onde existe alguma semelhança de acordo com as métricas de similaridades utilizadas. Para que o processo de mineração de dados, que irá gerar o classificador, seja mais eficiente, propõe-se que seja gerado o produto cartesiano dos entidades das duas ontologias. Esse é um dos diferencias da nossa proposta em relação às demais, já que do trabalhos relacionados somente o trabalho de YAM ++ [16] leva em consideração todos os pares de entidades das ontologias, o que é fundamental para o aprendizado do classificador para alinhamento de ontologia, pois além dos casos onde existe o alinhamento o classificador tem também os casos de não alinhamento para saber quando um par de entidades não é alinhável. Outra

análise que foi feita diz respeito às métricas de similaridades utilizadas. A Tabela 6 traz um quadro comparativo dos grupos de métricas que os trabalhos correlatos utilizaram nas suas abordagens.

Tabela 6 - Métricas utilizadas pelas ferramentas de alinhamento de ontologia

Ferramentas	Técnicas de Alinhamento [21]										
	Baseada em String	Baseada em Linguagem	Baseada em Recursos Linguísticos	Baseada em Restrições	Baseada em Reuso de Alinhamento	Baseada em Ontologia de Topo e Dominio Especifico	Baseada em Analise de Daddos e Estatística	Baseada em Grafos	Baseada em Taxonomia	Baseada em Repositório de Estrutura.	Baseada em Modelo
ATHENAS	√	√	√				√				
YAM ++ [16]	√	√	√					√			√
GOMMA [41]	√	√	√				√			√	
LogMap [13] e [62]		√	√		√						√
[32]	√	√	√							√	
Optima [55]	√							√		√	√
Aroma [20] e [60]	√								√		
Hertuda [69]	√	√				√					
CODI [35]	√						√				√

A metade dos trabalhos correlatos utilizaram as métricas baseadas em *String* e Linguagem, foram elas: YAM ++, GOMMA, [53] e Hertuda. Outra métrica em que mais quatro trabalhos (YAM ++, GOMMA, [53] e LogMap) utilizaram foi a baseada em recursos linguísticos. As baseadas em modelos também tiveram a utilização por quatro ferramentas: YAM ++, Optima, CODI e LogMap. Uma métrica que teve o segundo maior grupo de ferramentas utilizando foi a baseada em repositório de estrutura, foram as ferramentas: GOMMA, [53] e Optima. Dessas quatro grupos de métricas mais utilizadas a nossa abordagem considerou três: Baseadas em *String*, Linguagem e Recursos Linguísticos além da baseada em Análise de Dados e Estatística. Um fato interessante é que nenhuma ferramenta utilizou a métrica baseada em restrição, e somente uma das ferramentas utilizaram as métricas baseadas em reuso de alinhamento, ontologia de topo e taxonomia. E nenhuma das abordagens combinou os mesmos grupos de métricas. E de

todos os trabalhos somente a ferramenta LogMap não utilizou a métrica baseada em *String*.

5.2. Avaliação Experimental

Nesta seção é apresentada a avaliação dos resultados que foram realizados nos experimentos planejados para avaliar a abordagem. Foi realizada uma análise quantitativa dos resultados da abordagem com os trabalhos correlatos.

A experimentação oferece um modo sistemático, disciplinado, computável e controlado para que as teorias possam ser formuladas e corrigidas. O controle das variáveis é um fator crítico para se obter respostas em um experimento [74]. Tipicamente, experimentos são utilizados para verificar ou falsear uma hipótese previamente formulada. Hipótese é usualmente formulada como uma relação de causa. Nesse trabalho a hipótese foi verificar se os valores de precisão, cobertura e Medida-F melhoram em relação aos trabalhos correlatos (causa) com a combinação das métricas de similaridade (efeito). Os resultados de Athenas que serão comparados com os trabalhos relacionados são relativos ao algoritmo de aprendizado *RF* que obteve os melhores resultados em todos os cenários e domínios utilizados, exceto no domínio *MultiFarm* em que o *MLP* obteve os melhores valores para a Medida-F em três cenários e no cenário #4 em todos os domínios.

5.2.1. Domínio Benchmark

A avaliação da abordagem para este domínio pode ser verificada na Figura 34, que traz os resultados da Medida-F dos trabalhos relacionados participantes do OAEI 2012 e os resultados dos 5 grupos de métricas que foram experimentados com a abordagem Athenas. Neste domínio, dos grupos de métricas que foram utilizadas por Athenas, 3 ficaram com valores acima da ferramenta YAM ++ que obteve o melhor resultado 0,83

na Medida-F entre as ferramentas participantes. Das outras duas métricas de similaridade em Athenas Análise de Dados e Estatística obteve 0,67, um valor próximo a YAM ++, e todas as métricas combinadas obteve 0,83, o que a deixa na primeira colocação comparando com as ferramentas participantes do OAEI 2012.

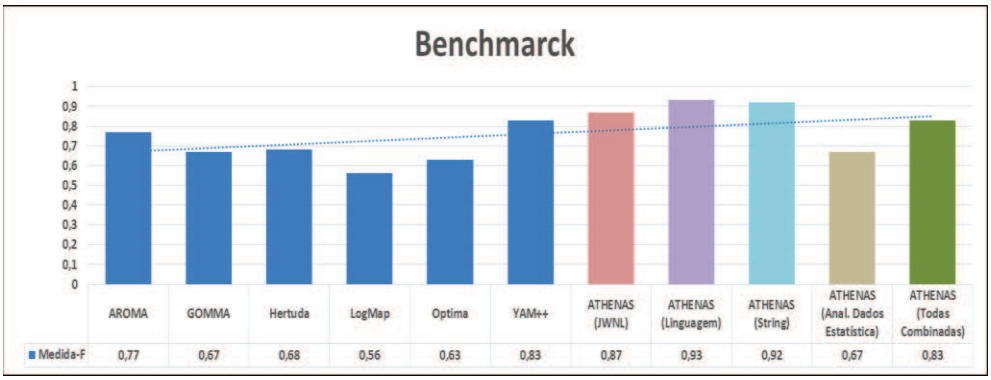


Figura 34 - Análise Quantitativa das Ferramentas para o Domínio Benchmark

Dos três grupo de métricas que foram utilizadas por Athenas que ficaram com os melhores valores, o grupo de métricas baseadas em recursos linguísticos ficou em terceiro lugar com o resultado de 0,87 para Medida-F, em segundo ficou o grupo de métrica baseado em *String* com o valor de 0,92 para Medida-F. E em primeiro lugar ficou grupo de métrica baseado em linguagem com o valor de 0,93 para Medida-F, o que vai de encontro com a hipótese levantada neste trabalho. Os três grupos de métricas obtiveram um bom resultado com valores próximos a 1, o que é um valor considerado como ótimo. Neste domínio Athenas se saiu muito bem comparado com os resultados da melhor ferramenta e da média anual referente a este domínio que é de 0,83 e 0,67 para a Medida-F respectivamente.

5.2.2. Domínio Conference

Um pouco melhor que no cenário anterior a abordagem de Athenas foi superior em relação a quase todas as ferramentas participantes do OAEI 2012, o pior desempenho foi

da métrica baseada em análise de dados e estatística que ficou com 0,54. Athenas foi superior a melhor ferramenta que foi YAM ++ com 0,71 na Medida-F, como é possível verificar na Figura 35.

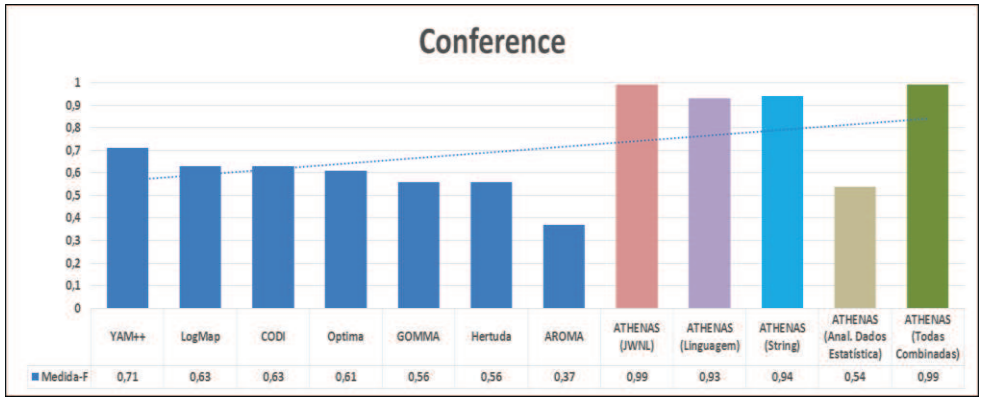


Figura 35 - Análise Quantitativa das Ferramentas para o Domínio Conference

Nos outros 4 grupos de métricas utilizados por Athenas para este domínio, os resultados para a Medida-F foram muito próximos com pequenas diferenças entres os grupos de métricas. O grupo baseado em Linguagem ficou a apenas 0,01 do grupo de métrica baseada em baseado em *String*, que por sua vez ficou 0,05 da combinação de todas as métricas com 0,99, ficando empatado com métrica baseada em Recurso Linguístico, sendo o melhor resultado neste domínio. Para o presente domínio Athenas ficou acima da média anual relativa a Medida-F que é de 0,55 comparado com todas as combinação de métrica de similaridade utilizada exceto na métrica baseada em análise de dados e estatística que ficou 0,01 abaixo da média.

5.2.3. Domínio MultiFarm

O domínio MultiFarm que é um domínio que participa da competição da OAEI há apenas dois anos e tem o valor mais baixo entre todos os domínios em relação a média para a Medida-F. Neste domínio as ontologias são disponibilizadas em 8 idiomas para a

tarefa de alinhamento como explicado na Seção 4.1.3. Neste trabalho apenas foram utilizadas as ontologias nos idiomas inglês e português.

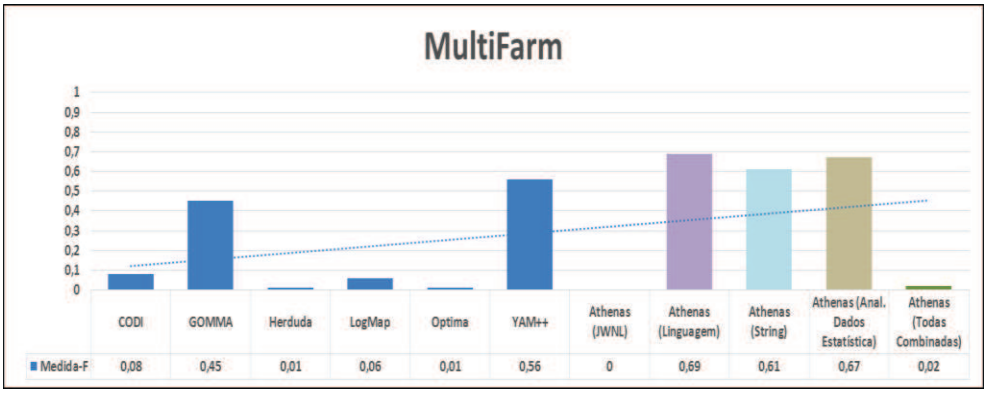


Figura 36 - Análise Quantitativa das Ferramentas para o Domínio MultiFarm

A Figura 36 traz os resultados das ferramentas participantes da do OAEI 2012 para este domínio nos idiomas Inglês / Português. E é notado que Athenas foi superior com três combinações de métricas de similaridade utilizadas, fincando melhor colocada que a melhor ferramenta participante neste domínio, que foi a YAM ++ com 0,56. E Athenas ficou bem acima da média desse domínio que é de 0,20 na Medida-F comparado com estas três combinações de métricas de similaridade utilizadas. A melhor combinação de métrica para este domínio em Athenas foi da métrica baseada em linguagem com 0,67, o que é muito intuitivo, já que neste domínio a tarefa de alinhamento é baseada no multilinguismo, onde as mesmas ontologias são disponibilizadas em oito idiomas diferentes, e as métricas que utilizam as técnicas baseadas em linguagem tendem a ter um melhor desempenho.

5.2.4. Considerações Gerais

Os resultados apresentados pelos experimentos demonstram que o objetivo da proposta foi alcançado com sucesso. Pois a combinação dos diferentes métricas de similaridade

obtiveram os melhores resultados em todos os domínios, quando comparadas com abordagem que são o estado da arte em alinhamento de ontologia. Em todos os cenários experimentados Athenas consegui a melhor colocação quando comparado com as ferramentas participantes da OAEI 2012. Um outro fato verificado durante os experimentos, é que a combinações dentro dos grupos métrica de similaridade baseado em linguagem e baseado em *String*, conseguiram resultados melhores que abordagens que são o estado da arte em alinhamento de ontologias.

Capítulo 6 – Conclusão

Este capítulo apresenta uma visão conclusiva sobre esta dissertação, e argumenta a importância do alinhamento de ontologias. São feitas as conclusões finais, bem como são discutidas as vantagens e limitações da abordagem apresentada, e alguns dos possíveis trabalhos futuros.

6.1. Discussão sobre a proposta

O objetivo desta dissertação foi melhorar a qualidade dos resultados no alinhamento de ontologia, com a combinação de diferentes grupos de similaridade. Neste trabalho foi feito um estudo acerca do estado-da-arte das soluções em alinhamento de ontologias, bem como os cenários onde o uso de tais soluções se faz necessária. Além de um estudo das métricas de similaridades, que serviram como embasamento na proposta de combinação de grupos de métricas de similaridade. Após a elaboração da solução proposta, uma ferramenta foi desenvolvida utilizando tecnologias citadas na literatura que são usadas na implementações de ferramentas de alinhamento de ontologias.

Para verificar se a proposta teria alguma melhora na relevância dos resultados, foram planejados experimentos em cinco diferentes cenários. E cada cenário foi utilizada a mesma configuração de algoritmo de aprendizado e domínios da ontologias, diferenciando somente o grupo de métricas de similaridade utilizadas. Foi possível verificar durante os experimentos que a combinação dentro do grupo de métricas baseado em linguagem, baseado em *Strings* e a combinação de todas as métrica, obtiveram ganho sobre as abordagem que são o estado-da-arte em alinhamento de ontologia. Como essas combinações de métricas de similaridades em todos os domínios experimentados

obtiveram os melhores resultados nas mediadas de qualidade diante das ferramentas que são o estado da arte, então, a hipótese levantada foi confirmada.

Quanto aos algoritmos que foram empregados na construção do classificador, vale ressaltar que a performance do algoritmos de aprendizado *RF* foi a melhor de todos os algoritmos utilizados. O algoritmo de aprendizado *SVM* obteve a pior performance para o aprendizado do classificador, levando dias para executar a tarefa de aprendizado em alguns cenários, enquanto o *MLP* levava duas horas no mesmo cenário e o *RF* alguns minutos. Um cenário onde isso ocorreu foi o cenário #3 no domínio *MultiFarm*.

6.2. Contribuições

Também foi feito um estudo acerca dos métodos e métricas existentes na literatura para avaliação da qualidade do resultado das soluções de alinhamento de ontologias. Além disso, foram elaborados 5 cenários de experimentos onde foram testados três algoritmos de aprendizado (*SVM, RF* e *MLP*), em três domínios diferentes, com 595 tarefas de alinhamento utilizando a ferramenta implementada. Dessa forma, como contribuições deste trabalho, citam-se:

- Resumo da literatura acerca do estado-da-arte das soluções de alinhamento de ontologias.
- Estudo de métodos e métricas para avaliação destas soluções;
- Estudos das métricas de similaridades utilizadas na literatura;
- Criação de uma abordagem automática para combinação de até 4 diferentes grupos de métricas de similaridade, com 23 métricas disponíveis para geração de correspondências entre os elementos das ontologias; responsável pela geração dos dados para o aprendizado do classificador.

- Combinação de diferentes grupos de métricas de similaridade, dos que foram feitos pelos trabalhos relacionados.

6.3. Limitações da abordagem e Trabalhos Futuros

As limitações da abordagem proposta e os trabalhos futuros serão discutidas nesta Seção. Primeiramente vamos expor as limitações percebidas durante o desenvolvimento da proposta e posteriormente sugerir alguns trabalhos futuros que podem melhorar a proposta.

- O processo pode levar muito tempo dependendo do tamanho das ontologias. Esse foi um dos fatores por que não foram utilizadas ontologias de larga escala e baseadas em instâncias.
- Outra limitação é quanto a geração de um alinhamento final no formato padrão RDF. Na proposta só foi aprendido o melhor conjunto de métricas pelo classificador, mas não foi implementado o classificador aprendido para gerar um alinhamento com todas as correspondências encontradas entre as ontologias.
- Um trabalho futuro seria a adaptação da proposta para trabalhar com ontologias de larga escala e instâncias, otimizando o tempo de todo o processo.
- Usar outros grupos de métricas de similaridade que não foram utilizados nesse trabalho, se possível todos os grupos existentes, para verificar se a combinação de outros grupos melhoraria as medidas de qualidade do alinhamento.
- Utilizar novos domínios.

Referências

[1] ABOLHASSANI, H., HARIRI, B., HAERI, S., “On Ontology Alignment Experiments”, In: *Webology*, Volume 3, Number 3, September, 2006.

[2] Apresentação “Boosting and Random Forest for Visual Recognition” Disponível em http://www.iis.ee.ic.ac.uk/~tkkim/iccv09_tutorial. Acessado em 06/09/2013.

[3] BAADER, F., *The description logic handbook: theory, implementation, and applications*. Cambridge university press, 2003.

[4] BERNERS-LEE, T., HENDLER, J., LASSILA, O., “The Semantic Web”. (A. Gómez-Pérez, Y. Yu, & Y. Ding, Eds.) *Scientific American*, 284(5), pp. 34-43. Citeseer, 2001.

[5] BISPO JR, E. L., e WASSERMANN, R. “Uma Nova Abordagem de Avaliação de Alinhamentos de Ontologias baseada em Consultas”. In: *ENIA - VIII Encontro Nacional de Inteligência Artificia*. Natal, RN. 2011.

[6] BREIMAN L.: *Random Forests*. Machine Learning, v. 45, n. 1, pp. 5-32, 2001.

[7] CHAPMAN, S. SimMetrics. http://sourceforge.net/projects/simmetrics/

[8] CHAWLA, N. V., BOWYER, K. W., HALL, L. O., e KEGELMEYER, W. P. “SMOTE: synthetic minority over-sampling technique”. arXiv preprint arXiv:1106.1813. 2011.

[9] CHANG , C. e CHIH-JEN LIN, “LIBSVM: a library for support vector machines”. *ACM Transactions on Intelligent Systems and Technology*, v.2, n.3 pp. 27, 2011.

[10] CHOWDHURY, G. G. "Natural language processing." *Annual review of information science and technology* v.37 n.1. pp. 51-89, 2003.

[11] COELHO, L. D. S.; SANTOS, A. A. P.; COSTA JR., NEWTON C. A. “Podemos prever a taxa de cambio brasileira? Evidência empírica utilizando inteligência computacional e modelos econométricos”. In: *Gestão e. Produção*, São Carlos, v. 15, n. 3, Dezembro. 2008.

[12] DAVID, J. “Association rule ontology matching approach”. In: *International Journal on Semantic Web and Information Systems (IJSWIS)*, v. 3, n. 2, pp. 27-49, 2007.

[13] DO, H. H., & RAHM, E. "Matching large schemas: Approaches and evaluation", In: *Information Systems,* v.32, n.6, pp. 857–885. 2007.

[14] DOMINGOS, P., LOWD, D., KOK, S., POON, H., RICHARDSON, M. and SINGLA, P. “Just add weights: Markov logic for the semantic web”. In: *Proceedings of the Workshop on Uncertain Reasoning for the Semantic Web*, pp. 1–25, 2008.

[15] DOSHI, P., KOLLI, R., THOMAS, C. “Inexact matching of ontology graphs using expectation-maximization”.In: *Web Semantics: Science, Services and Agents on the World Wide Web*, v. 7, n. 2, pp. 90-106, 2009.

[16] DUYHOA N., ZOHRA B., E REMI C. “A Flexible system for ontology matching”. Proceedings InCaise 2011 Forum.

[17] EHRIG, M., *Ontology Alignment: Bridging the Semantic Gap*, Springer 2007.

[18] EUZENAT, J. e VALTCHEV, P. “An Integrative Proximity Measure for Ontology Alignment”. In: *Semantic Integration Workshop, Second International Semantic Web Conference (ISWC-03)*. 2003.

[19] EUZENAT, J., MEILICKE, C., STUCKENSCHMIDT, H., SHVAIKO, P. & TROJAHN, C., “Ontology Alignment Evaluation Initiative: six years of experience”, in: *Journal on Data Semantics XV*, LNCS 6720, pp. 158–192. 2011.

[20] EUZENAT, J., FERRARA, A., VAN H. W. R., HOLLINK, L., MEILICKE, C., NIKOLOV, A., & dos SANTOS, C. T. “Final results of the ontology alignment evaluation initiative 2011”. In: Proc. *6th ISWC workshop on ontology matching (OM)*. pp. 85-110. 2011.

[21] EUZENAT, J. and SHVAIKO, P. *Ontology Matching*. Springer-Verlag, Berlin Heidelberg. 2007, X, 334 pp. 67 illus. ISBN 978-3-540-49611-3.

[22] FAYYAD, U., PIATETSKY-SHAPIRO, G., SMYTH, P., *From Data Mining to Knowledge Discovery in Databases*, AI Magazine, American Association for Artificial Intelligence, pp. 37-54. 1996.

[23] FERLIN, C. “*Imputação em cascata: uma abordagem para imputação multivariada de dados*”. Tese de D.Sc., COPPE/UFRJ, Rio de Janeiro, 2008.

[24] GOLDSCHMIDT, R. & PASSOS, E. *Data Mining: um guia prático*. Editora Campus, Rio de Janeiro: Elsevier, 2005.

[25] GOTOH, O. "An improved algorithm for matching biological sequences". *Journal of molecular biology,* v.162, n.3 pp. 705-708. 1982.

[26] GUARINO, N. “Formal Ontology and Information Systems*”*. In: *Formal Ontologies in Information Systems*, N. Guarino (Ed.), IOS Press, pp. 3 -15, 1998.

[27] GUIZZARDI, G. *Ontological Foundations for Structural Conceptual Models*. Ph.D. Thesis, University of Twente, The Netherlands. 2005.

[28] GRUBER, T., “Collective knowledge systems: Where the Social Web meets the Semantic Web”. *Journal of Web Semantics* 6(1), 4–13, 2008.

[29] GRUBER, T. R. “A Translation Approach to Portable Ontology Specifications”. In: Knowledge Acquisition, v.5, n.2, pp. 199-220, 1993.

[30] HAYKIN, S. S., HAYKIN, S. S., HAYKIN, S. S., & HAYKIN, S. S. *Neural networks and learning machines*. New York: Prentice Hall. Vol. 3, 2009.

[31] HAN, J., KAMBER, M. and JIAN, P. *Data mining: concepts and techniques*. Morgan kaufmann, 2006.

[32] HARIRI, B. B., SAYYADI, H., ABOLHASSANI, H. and ESMAILI, K. S. “Combining Ontology Alignment Metrics Using the Data Mining Techniques”. *In Proceeding of 2006 International Workshop on Context and Ontologies* (C&O '2006), Trento, Italy, August 2006.

[33] HASSOUN, M. H. *Fundamentals of artificial neural networks*. MIT press, 1995.

[34] HAZEWINKEL, M., ed. "Law of large numbers", In: *Encyclopedia of Mathematics*, Springer, 2001. ISBN 978-1-55608-010-4. 2005.

[35] HUBER, J., SZTYLER, T., NOESSNER, J., & MEILICKE, C. “CODI: Combinatorial optimization for data integration” In*: results for OAEI 2011.Ontology Matching*, pp. 134, 2011.

[36] JARO, M. A. “Advances in record-linkage methodology as applied to matching the 1985 census of Tampa”, Florida. In: *Journal of the American Statistical Association.* V.84, n.406, pp. 414–420. 1989.

[37] JÉRÔME D., J. EUZENAT, F. SCHARFFE, dos SANTOS, C. “The Alignment API 4.0”, Semantic web journal, v.2, n.1, pp. 3-10, 2011.

[38] JIMENEZ-RUIZ, E., CUENCA G., B., ZHOU, Y., HORROCKS, I.: Large-scale interactive ontology matching: Algorithms and implementation. In: *Eur. Conf. on Artif. Intell*. (ECAI) (2012).

[39] JIMÉNEZ-RUIZ, E., GRAU, B. C. “LogMap: Logic-based and Scalable Ontology Matching”. In: *the 10th International Semantic Web Conference (ISWC*). 2011.

[40] KARINA G., SÀNCHEZ-MARRÈA, M., CODINAA, V. “Choosing the Right Data Mining Technique: Classification of Methods and Intelligent Recommendation”. In *International Environmental Modelling and Software Society (iEMSs)*. For Environment’s Sake, Fifth Biennial Meeting, Ottawa, Canada. 2010.

[41] KIRSTEN, T., GROSS, A., HARTUNG, M., RAHM, E.: “Gomma: a component-based infrastructure for managing and analyzing life science ontologies and their evolution”. In: *Journal of Biomedical Semantics,* v.2, pp. 6, 2011.

[42] KONDRAK, G. “N-gram similarity and distance”. In: *Proceedings of the Twelfth International Conference on String Processing and Information Retrieval (SPIRE 2005)*, pp. 115-126, Buenos Aires, Argentina. 2005.

[43] KOHAVI, R. “A study of cross-validation and bootstrap for accuracy estimation and model selection”. In: *International joint Conference on artificial intelligence*. [S.l.: s.n.], v. 14, pp. 1137–1145. 1995.

[44] LEVENSTEIN, I. “Binary codes capable of correcting deletions, insertions and reversals”. In: *Cybernetics and Control Theory*. 1966.

[45] LIMA, E.S., POZZER, C.T., D'ORNELLAS, M., CIARLINI, A.E.M., FEIJO, B., FURTADO, A.L., 2009. “Support Vector Machines for Cinematography Real-Time Camera Control in Storytelling Environments”. *In: VIII Brazilian Symposium on Games and Digital Entertainment*, Rio de Janeiro, Brazil, pp. 44-51. [DOI: http://doi.ieeecomputersociety.org/10.1109/SBGAMES.2009.14].

[46] LIMA, E.S. “Apresentação de aula de Inteligência Artificial. Support Vector Machine (SVM)”. Disponível em http://edirlei.3dgb.com.br/aulas/ia/IA_Aula_21_SVM.pdf. Acessado em 07/03/2013.

[47] MAEDCHE, A. “Ontology Learning for the Sematic Web”. In: *Kluwer Academic Publishers*. pp. 173-190. 2002.

[48] MASSMANN, S.; RAUNICH, S., AUMUELLER, D.; ARNOLD, P.; RAHM and ERHARD. “Evolution of the COMA Match System”, *OM-2011 The Sixth International Workshop on Ontology Matching*, October 24th, Bonn, Germany. pp. 49, 2011.

[49] MITCHELL, T. M. *Machine Learning*. Burr Ridge, IL:McGraw-Hill. v. 45, 1997.

[50] MONGE, A., ELKAN, C. “The field-matching problem: algorithm and applications”. *In: Proceedings of the second international Conference on Knowledge Discovery and Data Mining*. pp. 267-270, 1996.

[51] NASCIMENTO, R. F. F., ALCÂNTARA, E. H. D., KAMPEL, M., STECH, J. L., NOVO, E. M. L. D. M., e FONSECA, L. M. G. "O algoritmo Support Vector Machines (SVM): avaliação da separação ótima de classes em imagens CCD-CBERS-2." In: *Simpósio Brasileiro de Sensoriamento Remoto*. v. 14, pp. 2079-2086, 2009.

[52] NEEDLEMAN, S. B., WUNSCH, C. D. “A general method applicable to the search for similarities in the amino acid sequence of two proteins”. In: *Journal of Molecular Biology*. 48, 1970, S. 443–453.

[53] NEZHADI, A., B. SHADGARA, and ALIREZA O. "Ontology alignment using machine learning techniques”. *International Journal of Computer Science & Information Technology* v.3. 2011.

[54] ODELL, K. M. e RUSSELL, R. C. “Soundex phonetic comparison system” [cf. U.S. Patents 1261167 (1918), 1435663 (1922).

[55] PANG-NING T., M. STEINBACH, V. KUMAR. *Introdução ao “Data Mining” Mineração de Dados*. Rio de Janeiro: Editora Ciência Moderna Ltda. 2009.

[56] PRINCETON UNIVERSITY "About WordNet." WordNet. Princeton University. 2010. http://wordnet.princeton.edu

[57] RAFAEL D. C. S. "Introdução a Mineração de Dados com Aplicações em Ciências Espaciais", editor Reinaldo Rosa and Nilson Sant'anna, Minicursos ELAC 2012.

[58] RAKESH, A., TOMASZ, I., and ARUN, S. “Mining association rules between sets of items in large databases”. In: *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, pp. 207–216. ACM Press, 1993.

[59] RÉGIS, G., EINOSHIN, S., FABRICE, G., and FILIPPO, S. editors. “Statistical Implicative Analysis, Theory and Applications”, of Studies In*: Computational Intelligence*. volume 127, Springer, 2008.

[60] ROBERT E. L., HUI, L. ADAM, C. and GEORGE, G. Laboratory of Computational Proteomics. *University of Illinois at Chicago*. Acessado em 15/06/2013. Disponível em http://proteomics.bioengr.uic.edu/malibu.

[61] RUMELHART, D. E., MCCLELLAND, J. L., & the PDP research group. Parallel distributed processing: In: *Explorations in the microstructure of cognition*. Volume I. Cambridge, MA: MIT Press. 1986.

[62] SARULADHA, K.; AGHILA, G.; SATHIYA, B. “A Comparative Analysis of Ontology and Schema Matching Systems”. In: *International Journal of Computer Applications*, v. 34, n. 8, p. 14-21, 2011.

[63] SHVAIKO, P., and EUZENAT, J., “A Survey of Schema-based Matching Approaches”, LNCS, In: *Journal on Data Semantics*, Volume 4, pp. 146-171, Germany, 2005.

[64] SILVA A. A., PADILHA, N.F.; SIQUEIRA, S.W.M.; BAIÃO, F., REVOREDO, K. “Using Concept Maps and Ontology Alignment for Learning Assessment*”. IEEE Multidisciplinary Engineering Education Magazine*, v. 7, pp. 33-40, 2012.

[65] SILVA, A. A.; GUEDES, A. V. S.; REVOREDO, K.; BAIÃO, F. “Classificador de Alinhamento de Ontologias Utilizando Técnicas de Aprendizado de Máquina”. In: *Simpósio Brasileiro de Sistemas de Informação*, 2013, João Pessoa. Anais do Simpósio Brasileiro de Sistemas de Informação, v. 1. pp. 1. 2013.

[66] SILVA, A., A.; FERREIRA, R. F.; FERLIN, C.; GOLDSCHMIDT, R. R.; DINIZ CORRÊA, F. A.; CASTANEDA, R. “A New Approach Based on Concept of Quality Group for Imputation”. In: *7th CONTECSI International Conference on Information*

Systems and Technology Management (7º CONTECSI), São Paulo. Anais do Contecsi. São Paulo: Centrográfica Gráfica & Editora Ltda, v. 1. pp. 3574-3596. 2010.

[67] SILVA, V. S.; CAMPOS, M. L. M.; SILVA, J. C. P.; CAVALCANTI, M. C. "An Approach for the Alignment of Biomedical Ontologies based on Foundational Ontologies". In: *Journal of Information and Data Management*, v. 2, pp. 557-572, 2011.

[68] SMITH, T. F.; and WATERMAN, M. S. "Identification of Common Molecular Subsequences". Journal of Molecular Biology 147: 195–197. Doi: 10.1016/0022-2836(81)90087-5. PMID 7265238. 1981.

[69] SVEN H., "Hertuda. Results for OEAI 2012", In: *Seventh International Workshop on Ontology Matching (OM)*, 2012.

[70] STUDER, R. BENJAMINS, V. R. and FENSEL, D. "Knowledge engineering: principles and methods". In: *Data Knowledge Engineering*, Elsevier Ltd, Volume 25, Issues 1-2, pp.161-197. 1998.

[71] TEKNOMO, K. "Similarity Measurement". Disponível em http://people.revoledu.com\kardi\tutorial\Similarity\, acessado em 22/06/2013.

[72] THORNTON, C., HUTTER, F., HOOS. H. H., and LEYTON-BROWN, K. "Auto-WEKA: Automated Selection and Hyper-Parameter Optimization of Classification Algorithms". In: *Proceedings Computing Research Repository*, abs/1208.3719, 2012.

[73] VAPNIK, V. *The Nature of Statistical Learning Theory*, Springer Verlag, New York, 1995.

[74] WAINER, J. "Métodos de pesquisa quantitativa e qualitativa para a Ciência da Computação". *Atualização em Informática. Org: Tomasz Kowaltowski; Karin Breitman*. Rio de Janeiro: Ed. PUC-Rio, 2007.

[75] WINKLER, W. E. "String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage". *In Proceedings of the Section on Survey Research Methods (American Statistical Association)*: pp. 354–359. 1990.

[76] WITTEN, I. H., FRANK, E., e HALL, M. A. *Data Mining: Practical Machine Learning - Tools and Techniques*. Morgan Kaufmann, 3nd Edition, 2011.

[77] ZHANG, S., ZHANG, C., YANG, Q. "Data Preparation for Data Mining", In: *Applied Artificial Intelligence*, v. 17, n. 5-6 (May-Jun), pp. 375-381. 2003.

Apêndice I – Panejamento dos Experimentos

Tabela 1- Planejamento de todos os cenários dos experimentos

Tarefa de Alinhamento	Domínio OAEI		Ontologias		Métrica de Similaridade																						Técnicas de Classificação			Status
					Baseada em Recursos Linguísticos	Baseada em Linguagem						Baseada em *Strings*														Baseada em Análise de Dados e Estatística				
	Domínio	Cenário	O1	O2	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12	M13	M14	M15	M16	M17	M18	M19	M20	M21	M22	M23	RF	MLP	SVM	Feito
1	Benchmark	#1	101	101	X																							X	X	X	√
2		#1	101	103	X																							X	X	X	√
3		#1	101	104	X																							X	X	X	√
4		#1	101	201	X																							X	X	X	√
5		#1	101	202	X																							X	X	X	√
6		#1	101	204	X																							X	X	X	√
7		#1	101	205	X																							X	X	X	√
8		#1	101	206	X																							X	X	X	√

9		**#1**	**101**	**207**	X																							X	X	X	√
10		**#1**	**101**	**208**	X																							X	X	X	√
11		**#1**	**101**	**209**	X																							X	X	X	√
12		**#1**	**101**	**210**	X																							X	X	X	√
13		**#1**	**101**	**221**	X																							X	X	X	√
14		**#1**	**101**	**222**	X																							X	X	X	√
15		**#1**	**101**	**223**	X																							X	X	X	√
16		**#1**	**101**	**224**	X																							X	X	X	√
17		**#1**	**101**	**225**	X																							X	X	X	√
18		**#1**	**101**	**228**	X																							X	X	X	√
19		**#1**	**101**	**230**	X																							X	X	X	√
20		**#1**	**101**	**232**	X																							X	X	X	√
21		**#1**	**101**	**233**	X																							X	X	X	√
22		**#1**	**101**	**236**	X																							X	X	X	√
23		**#1**	**101**	**237**	X																							X	X	X	√
24		**#1**	**101**	**238**	X																							X	X	X	√
25		**#1**	**101**	**239**	X																							X	X	X	√
26		**#1**	**101**	**240**	X																							X	X	X	√
27		**#1**	**101**	**241**	X																							X	X	X	√
28		**#1**	**101**	**246**	X																							X	X	X	√
29		**#1**	**101**	**247**	X																							X	X	X	√
30		**#1**	**101**	**248**	X																							X	X	X	√
31		**#1**	**101**	**249**	X																							X	X	X	√
32		**#1**	**101**	**250**	X																							X	X	X	√

33		**#1**	**101**	**251**	X																							X	X	X	√
34		**#1**	**101**	**252**	X																							X	X	X	√
35		**#1**	**101**	**253**	X																							X	X	X	√
36		**#1**	**101**	**254**	X																							X	X	X	√
37		**#1**	**101**	**257**	X																							X	X	X	√
38		**#1**	**101**	**258**	X																							X	X	X	√
39		**#1**	**101**	**259**	X																							X	X	X	√
40		**#1**	**101**	**260**	X																							X	X	X	√
41		**#1**	**101**	**261**	X																							X	X	X	√
42		**#1**	**101**	**262**	X																							X	X	X	√
43		**#1**	**101**	**265**	X																							X	X	X	√
44		**#1**	**101**	**266**	X																							X	X	X	√
45		**#1**	**101**	**301**	X																							X	X	X	√
46		**#1**	**101**	**302**	X																							X	X	X	√
47		**#1**	**101**	**303**	X																							X	X	X	√
48		**#1**	**101**	**304**	X																							X	X	X	√
49	**Conference**	**#1**	**iasted**	**sigkdd**	X																							X	X	X	√
50		**#1**	**ekaw**	**sigkdd**	X																							X	X	X	√
51		**#1**	**ekaw**	**iasted**	X																							X	X	X	√
52		**#1**	**edas**	**sigkdd**	X																							X	X	X	√
53		**#1**	**edas**	**iasted**	X																							X	X	X	√
54		**#1**	**edas**	**ekaw**	X																							X	X	X	√
55		**#1**	**confOf**	**sigkdd**	X																							X	X	X	√
56		**#1**	**confOf**	**iasted**	X																							X	X	X	√

57		**#1**	**confOf**	**ekaw**	X																							X	X	X	√
58		**#1**	**confOf**	**edas**	X																							X	X	X	√
59		**#1**	**Conference**	**sigkdd**	X																							X	X	X	√
60		**#1**	**Conference**	**iasted**	X																							X	X	X	√
61		**#1**	**Conference**	**ekaw**	X																							X	X	X	√
62		**#1**	**Conference**	**edas**	X																							X	X	X	√
63		**#1**	**Conference**	**confOf**	X																							X	X	X	√
64		**#1**	**cmt**	**sigkdd**	X																							X	X	X	√
65		**#1**	**cmt**	**iasted**	X																							X	X	X	√
66		**#1**	**cmt**	**ekaw**	X																							X	X	X	√
67		**#1**	**cmt**	**edas**	X																							X	X	X	√
68		**#1**	**cmt**	**confOf**	X																							X	X	X	√
69		**#1**	**cmt**	**Conference**	X																							X	X	X	√
70	**MultiFarm en-pt**	**#1**	**cmt-en**	**cmt-pt**	X																							X	X	X	√
71		**#1**	**cmt-en**	**conference-pt**	X																							X	X	X	√
72		**#1**	**cmt-pt**	**conference-en**	X																							X	X	X	√
73		**#1**	**cmt-en**	**confOf-pt**	X																							X	X	X	√
74		**#1**	**cmt-pt**	**confOf-en**	X																							X	X	X	√
75		**#1**	**cmt-en**	**edas-pt**	X																							X	X	X	√
76		**#1**	**cmt-pt**	**edas-en**	X																							X	X	X	√
77		**#1**	**cmt-en**	**ekaw-pt**	X																							X	X	X	√
78		**#1**	**cmt-pt**	**ekaw-en**	X																							X	X	X	√
79		**#1**	**cmt-en**	**iasted-pt**	X																							X	X	X	√
80		**#1**	**cmt-pt**	**iasted-en**	X																							X	X	X	√

81		**#1**	**cmt-en**	**sigkdd-pt**	X																							X	X	X	√
82		**#1**	**cmt-pt**	**sigkdd-en**	X																							X	X	X	√
83		**#1**	**Conference-en**	**Conference-pt**	X																							X	X	X	√
84		**#1**	**Conference-en**	**confOf-pt**	X																							X	X	X	√
85		**#1**	**Conference-pt**	**confOf-en**	X																							X	X	X	√
86		**#1**	**Conference-en**	**edas-pt**	X																							X	X	X	√
87		**#1**	**Conference-pt**	**edas-en**	X																							X	X	X	√
88		**#1**	**Conference-en**	**ekaw-pt**	X																							X	X	X	√
89		**#1**	**Conference-pt**	**ekaw-en**	X																							X	X	X	√
90		**#1**	**Conference-en**	**iasted-pt**	X																							X	X	X	√
91		**#1**	**Conference-pt**	**iasted-en**	X																							X	X	X	√
92		**#1**	**Conference-en**	**sigkdd-pt**	X																							X	X	X	√
93		**#1**	**Conference-pt**	**sigkdd-en**	X																							X	X	X	√
94		**#1**	**confOf-en**	**confOf-pt**	X																							X	X	X	√
95		**#1**	**confOf-en**	**edas-pt**	X																							X	X	X	√
96		**#1**	**confOf-pt**	**edas-en**	X																							X	X	X	√
97		**#1**	**confOf-en**	**ekaw-pt**	X																							X	X	X	√
98		**#1**	**confOf-pt**	**ekaw-en**	X																							X	X	X	√
99		**#1**	**confOf-en**	**iasted-pt**	X																							X	X	X	√
100		**#1**	**confOf-pt**	**iasted-en**	X																							X	X	X	√
101		**#1**	**confOf-en**	**sigkdd-pt**	X																							X	X	X	√
102		**#1**	**confOf-pt**	**sigkdd-en**	X																							X	X	X	√
103		**#1**	**edas-en**	**edas-pt**	X																							X	X	X	√
104		**#1**	**edas-en**	**ekaw-pt**	X																							X	X	X	√

105		**#1**	**edas-pt**	**ekaw-en**	X																							X	X	X	√
106		**#1**	**edas-en**	**iasted-pt**	X																							X	X	X	√
107		**#1**	**edas-pt**	**iasted-en**	X																							X	X	X	√
108		**#1**	**edas-en**	**sigkdd-pt**	X																							X	X	X	√
109		**#1**	**edas-pt**	**sigkdd-en**	X																							X	X	X	√
110		**#1**	**edas-en**	**confOf-pt**	X																							X	X	X	√
111		**#1**	**ekaw-en**	**ekaw-pt**	X																							X	X	X	√
112		**#1**	**ekaw-en**	**iasted-pt**	X																							X	X	X	√
113		**#1**	**ekaw-pt**	**iasted-en**	X																							X	X	X	√
114		**#1**	**ekaw-en**	**sigkdd-pt**	X																							X	X	X	√
115		**#1**	**ekaw-pt**	**sigkdd-en**	X																							X	X	X	√
116		**#1**	**iasted-en**	**iasted-pt**	X																							X	X	X	√
117		**#1**	**iasted-en**	**sigkdd-pt**	X																							X	X	X	√
118		**#1**	**iasted-pt**	**sigkdd-en**	X																							X	X	X	√
119		**#1**	**sigkdd-en**	**sigkdd-pt**	X																							X	X	X	√
120	**Benchmark**	**#2**	**101**	**101**		X	X	X	X	X	X																	X	X	X	√
121		**#2**	**101**	**103**		X	X	X	X	X	X																	X	X	X	√
122		**#2**	**101**	**104**		X	X	X	X	X	X																	X	X	X	√
123		**#2**	**101**	**201**		X	X	X	X	X	X																	X	X	X	√
124		**#2**	**101**	**202**		X	X	X	X	X	X																	X	X	X	√
125		**#2**	**101**	**204**		X	X	X	X	X	X																	X	X	X	√
126		**#2**	**101**	**205**		X	X	X	X	X	X																	X	X	X	√
127		**#2**	**101**	**206**		X	X	X	X	X	X																	X	X	X	√
128		**#2**	**101**	**207**		X	X	X	X	X	X																	X	X	X	√

129		**#2**	**101**	**208**		X	X	X	X	X	X																	X	X	X	√
130		**#2**	**101**	**209**		X	X	X	X	X	X																	X	X	X	√
131		**#2**	**101**	**210**		X	X	X	X	X	X																	X	X	X	√
132		**#2**	**101**	**221**		X	X	X	X	X	X																	X	X	X	√
133		**#2**	**101**	**222**		X	X	X	X	X	X																	X	X	X	√
134		**#2**	**101**	**223**		X	X	X	X	X	X																	X	X	X	√
135		**#2**	**101**	**224**		X	X	X	X	X	X																	X	X	X	√
136		**#2**	**101**	**225**		X	X	X	X	X	X																	X	X	X	√
137		**#2**	**101**	**228**		X	X	X	X	X	X																	X	X	X	√
138		**#2**	**101**	**230**		X	X	X	X	X	X																	X	X	X	√
139		**#2**	**101**	**232**		X	X	X	X	X	X																	X	X	X	√
140		**#2**	**101**	**233**		X	X	X	X	X	X																	X	X	X	√
141		**#2**	**101**	**236**		X	X	X	X	X	X																	X	X	X	√
142		**#2**	**101**	**237**		X	X	X	X	X	X																	X	X	X	√
143		**#2**	**101**	**238**		X	X	X	X	X	X																	X	X	X	√
144		**#2**	**101**	**239**		X	X	X	X	X	X																	X	X	X	√
145		**#2**	**101**	**240**		X	X	X	X	X	X																	X	X	X	√
146		**#2**	**101**	**241**		X	X	X	X	X	X																	X	X	X	√
147		**#2**	**101**	**246**		X	X	X	X	X	X																	X	X	X	√
148		**#2**	**101**	**247**		X	X	X	X	X	X																	X	X	X	√
149		**#2**	**101**	**248**		X	X	X	X	X	X																	X	X	X	√
150		**#2**	**101**	**249**		X	X	X	X	X	X																	X	X	X	√
151		**#2**	**101**	**250**		X	X	X	X	X	X																	X	X	X	√
152		**#2**	**101**	**251**		X	X	X	X	X	X																	X	X	X	√

153		#2	101	252		X	X	X	X	X	X																	X	X	X	√
154		#2	101	253		X	X	X	X	X	X																	X	X	X	√
155		#2	101	254		X	X	X	X	X	X																	X	X	X	√
156		#2	101	257		X	X	X	X	X	X																	X	X	X	√
157		#2	101	258		X	X	X	X	X	X																	X	X	X	√
158		#2	101	259		X	X	X	X	X	X																	X	X	X	√
159		#2	101	260		X	X	X	X	X	X																	X	X	X	√
160		#2	101	261		X	X	X	X	X	X																	X	X	X	√
161		#2	101	262		X	X	X	X	X	X																	X	X	X	√
162		#2	101	265		X	X	X	X	X	X																	X	X	X	√
163		#2	101	266		X	X	X	X	X	X																	X	X	X	√
164		#2	101	301		X	X	X	X	X	X																	X	X	X	√
165		#2	101	302		X	X	X	X	X	X																	X	X	X	√
166		#2	101	303		X	X	X	X	X	X																	X	X	X	√
167		#2	101	304		X	X	X	X	X	X																	X	X	X	√
168	**Conference**	#2	iasted	sigkdd		X	X	X	X	X	X																	X	X	X	√
169		#2	ekaw	sigkdd		X	X	X	X	X	X																	X	X	X	√
170		#2	ekaw	iasted		X	X	X	X	X	X																	X	X	X	√
171		#2	edas	sigkdd		X	X	X	X	X	X																	X	X	X	√
172		#2	edas	iasted		X	X	X	X	X	X																	X	X	X	√
173		#2	edas	ekaw		X	X	X	X	X	X																	X	X	X	√
174		#2	confOf	sigkdd		X	X	X	X	X	X																	X	X	X	√
175		#2	confOf	iasted		X	X	X	X	X	X																	X	X	X	√
176		#2	confOf	ekaw		X	X	X	X	X	X																	X	X	X	√

177		#2	confOf	edas		X	X	X	X	X	X																	X	X	X	√
178		#2	Conference	sigkdd		X	X	X	X	X	X																	X	X	X	√
179		#2	Conference	iasted		X	X	X	X	X	X																	X	X	X	√
180		#2	Conference	ekaw		X	X	X	X	X	X																	X	X	X	√
181		#2	Conference	edas		X	X	X	X	X	X																	X	X	X	√
182		#2	Conference	confOf		X	X	X	X	X	X																	X	X	X	√
183		#2	cmt	sigkdd		X	X	X	X	X	X																	X	X	X	√
184		#2	cmt	iasted		X	X	X	X	X	X																	X	X	X	√
185		#2	cmt	ekaw		X	X	X	X	X	X																	X	X	X	√
186		#2	cmt	edas		X	X	X	X	X	X																	X	X	X	√
187		#2	cmt	confOf		X	X	X	X	X	X																	X	X	X	√
188		#2	cmt	Conference		X	X	X	X	X	X																	X	X	X	√
189	**MultiFarm en-pt**	#2	cmt-en	cmt-pt		X	X	X	X	X	X																	X	X	X	√
190		#2	cmt-en	conference-pt		X	X	X	X	X	X																	X	X	X	√
191		#2	cmt-pt	conference-en		X	X	X	X	X	X																	X	X	X	√
192		#2	cmt-en	confOf-pt		X	X	X	X	X	X																	X	X	X	√
193		#2	cmt-pt	confOf-en		X	X	X	X	X	X																	X	X	X	√
194		#2	cmt-en	edas-pt		X	X	X	X	X	X																	X	X	X	√
195		#2	cmt-pt	edas-en		X	X	X	X	X	X																	X	X	X	√
196		#2	cmt-en	ekaw-pt		X	X	X	X	X	X																	X	X	X	√
197		#2	cmt-pt	ekaw-en		X	X	X	X	X	X																	X	X	X	√
198		#2	cmt-en	iasted-pt		X	X	X	X	X	X																	X	X	X	√
199		#2	cmt-pt	iasted-en		X	X	X	X	X	X																	X	X	X	√
200		#2	cmt-en	sigkdd-pt		X	X	X	X	X	X																	X	X	X	√

201		#2	cmt-pt	sigkdd-en		X	X	X	X	X	X																	X	X	X	√
202		#2	Conference-en	Conference-pt		X	X	X	X	X	X																	X	X	X	√
203		#2	Conference-en	confOf-pt		X	X	X	X	X	X																	X	X	X	√
204		#2	Conference-pt	confOf-en		X	X	X	X	X	X																	X	X	X	√
205		#2	Conference-en	edas-pt		X	X	X	X	X	X																	X	X	X	√
206		#2	Conference-pt	edas-en		X	X	X	X	X	X																	X	X	X	√
207		#2	Conference-en	ekaw-pt		X	X	X	X	X	X																	X	X	X	√
208		#2	Conference-pt	ekaw-en		X	X	X	X	X	X																	X	X	X	√
209		#2	Conference-en	iasted-pt		X	X	X	X	X	X																	X	X	X	√
210		#2	Conference-pt	iasted-en		X	X	X	X	X	X																	X	X	X	√
211		#2	Conference-en	sigkdd-pt		X	X	X	X	X	X																	X	X	X	√
212		#2	Conference-pt	sigkdd-en		X	X	X	X	X	X																	X	X	X	√
213		#2	confOf-en	confOf-pt		X	X	X	X	X	X																	X	X	X	√
214		#2	confOf-en	edas-pt		X	X	X	X	X	X																	X	X	X	√
215		#2	confOf-pt	edas-en		X	X	X	X	X	X																	X	X	X	√
216		#2	confOf-en	ekaw-pt		X	X	X	X	X	X																	X	X	X	√
217		#2	confOf-pt	ekaw-en		X	X	X	X	X	X																	X	X	X	√
218		#2	confOf-en	iasted-pt		X	X	X	X	X	X																	X	X	X	√
219		#2	confOf-pt	iasted-en		X	X	X	X	X	X																	X	X	X	√
220		#2	confOf-en	sigkdd-pt		X	X	X	X	X	X																	X	X	X	√
221		#2	confOf-pt	sigkdd-en		X	X	X	X	X	X																	X	X	X	√
222		#2	edas-en	edas-pt		X	X	X	X	X	X																	X	X	X	√
223		#2	edas-en	ekaw-pt		X	X	X	X	X	X																	X	X	X	√
224		#2	edas-pt	ekaw-en		X	X	X	X	X	X																	X	X	X	√

225		**#2**	**edas-en**	**iasted-pt**		X	X	X	X	X	X																	X	X	X	√
226		**#2**	**edas-pt**	**iasted-en**		X	X	X	X	X	X																	X	X	X	√
227		**#2**	**edas-en**	**sigkdd-pt**		X	X	X	X	X	X																	X	X	X	√
228		**#2**	**edas-pt**	**sigkdd-en**		X	X	X	X	X	X																	X	X	X	√
229		**#2**	**edas-en**	**confOf**		X	X	X	X	X	X																	X	X	X	√
230		**#2**	**ekaw-en**	**ekaw-pt**		X	X	X	X	X	X																	X	X	X	√
231		**#2**	**ekaw-en**	**iasted-pt**		X	X	X	X	X	X																	X	X	X	√
232		**#2**	**ekaw-pt**	**iasted-en**		X	X	X	X	X	X																	X	X	X	√
233		**#2**	**ekaw-en**	**sigkdd-pt**		X	X	X	X	X	X																	X	X	X	√
234		**#2**	**ekaw-pt**	**sigkdd-en**		X	X	X	X	X	X																	X	X	X	√
235		**#2**	**iasted-en**	**iasted-pt**		X	X	X	X	X	X																	X	X	X	√
236		**#2**	**iasted-en**	**sigkdd-pt**		X	X	X	X	X	X																	X	X	X	√
237		**#2**	**iasted-pt**	**sigkdd-en**		X	X	X	X	X	X																	X	X	X	√
238		**#2**	**sigkdd-en**	**sigkdd-pt**		X	X	X	X	X	X																	X	X	X	√
239	**Benchmark**	**#3**	**101**	**101**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
240		**#3**	**101**	**103**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
241		**#3**	**101**	**104**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
242		**#3**	**101**	**201**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
243		**#3**	**101**	**202**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
244		**#3**	**101**	**204**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
245		**#3**	**101**	**205**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
246		**#3**	**101**	**206**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
247		**#3**	**101**	**207**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
248		**#3**	**101**	**208**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√

249		#3	101	209								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
250		#3	101	210								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
251		#3	101	221								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
252		#3	101	222								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
253		#3	101	223								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
254		#3	101	224								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
255		#3	101	225								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
256		#3	101	228								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
257		#3	101	230								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
258		#3	101	232								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
259		#3	101	233								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
260		#3	101	236								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
261		#3	101	237								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
262		#3	101	238								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
263		#3	101	239								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
264		#3	101	240								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
265		#3	101	241								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
266		#3	101	246								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
267		#3	101	247								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
268		#3	101	248								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
269		#3	101	249								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
270		#3	101	250								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
271		#3	101	251								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
272		#3	101	252								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√

273		**#3**	**101**	**253**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
274		**#3**	**101**	**254**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
275		**#3**	**101**	**257**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
276		**#3**	**101**	**258**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
277		**#3**	**101**	**259**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
278		**#3**	**101**	**260**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
279		**#3**	**101**	**261**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
280		**#3**	**101**	**262**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
281		**#3**	**101**	**265**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
282		**#3**	**101**	**266**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
283		**#3**	**101**	**301**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
284		**#3**	**101**	**302**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
285		**#3**	**101**	**303**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
286		**#3**	**101**	**304**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
287	**Conference**	**#3**	**iasted**	**sigkdd**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
288		**#3**	**ekaw**	**sigkdd**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
289		**#3**	**ekaw**	**iasted**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
290		**#3**	**edas**	**sigkdd**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
291		**#3**	**edas**	**iasted**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
292		**#3**	**edas**	**ekaw**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
293		**#3**	**confOf**	**sigkdd**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
294		**#3**	**confOf**	**iasted**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
295		**#3**	**confOf**	**ekaw**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
296		**#3**	**confOf**	**edas**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√

297		#3	Conference	sigkdd								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
298		#3	Conference	iasted								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
299		#3	Conference	ekaw								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
300		#3	Conference	edas								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
301		#3	Conference	confOf								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
302		#3	cmt	sigkdd								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
303		#3	cmt	iasted								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
304		#3	cmt	ekaw								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
305		#3	cmt	edas								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
306		#3	cmt	confOf								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
307		#3	cmt	Conference								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
308	**MultiFarm en-pt**	#3	cmt-en	cmt-pt								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
309		#3	cmt-en	conference-pt								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
310		#3	cmt-pt	conference-en								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
311		#3	cmt-en	confOf-pt								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
312		#3	cmt-pt	confOf-en								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
313		#3	cmt-en	edas-pt								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
314		#3	cmt-pt	edas-en								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
315		#3	cmt-en	ekaw-pt								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
316		#3	cmt-pt	ekaw-en								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
317		#3	cmt-en	iasted-pt								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
318		#3	cmt-pt	iasted-en								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
319		#3	cmt-en	sigkdd-pt								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
320		#3	cmt-pt	sigkdd-en								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√

321		#3	Conference-en	Conference-pt								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
322		#3	Conference-en	confOf-pt								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
323		#3	Conference-pt	confOf-en								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
324		#3	Conference-en	edas-pt								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
325		#3	Conference-pt	edas-en								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
326		#3	Conference-en	ekaw-pt								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
327		#3	Conference-pt	ekaw-en								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
328		#3	Conference-en	iasted-pt								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
329		#3	Conference-pt	iasted-en								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
330		#3	Conference-en	sigkdd-pt								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
331		#3	Conference-pt	sigkdd-en								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
332		#3	confOf-en	confOf-pt								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
333		#3	confOf-en	edas-pt								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
334		#3	confOf-pt	edas-en								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
335		#3	confOf-en	ekaw-pt								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
336		#3	confOf-pt	ekaw-en								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
337		#3	confOf-en	iasted-pt								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
338		#3	confOf-pt	iasted-en								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
339		#3	confOf-en	sigkdd-pt								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
340		#3	confOf-pt	sigkdd-en								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
341		#3	edas-en	edas-pt								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
342		#3	edas-en	ekaw-pt								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
343		#3	edas-pt	ekaw-en								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
344		#3	edas-en	iasted-pt								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√

345		**#3**	**edas-pt**	**iasted-en**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
346		**#3**	**edas-en**	**sigkdd-pt**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
347		**#3**	**edas-pt**	**sigkdd-en**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
348		**#3**	**edas-en**	**confOf**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
349		**#3**	**ekaw-en**	**ekaw-pt**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
350		**#3**	**ekaw-en**	**iasted-pt**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
351		**#3**	**ekaw-pt**	**iasted-en**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
352		**#3**	**ekaw-en**	**sigkdd-pt**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
353		**#3**	**ekaw-pt**	**sigkdd-en**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
354		**#3**	**iasted-en**	**iasted-pt**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
355		**#3**	**iasted-en**	**sigkdd-pt**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
356		**#3**	**iasted-pt**	**sigkdd-en**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
357		**#3**	**sigkdd-en**	**sigkdd-pt**								X	X	X	X	X	X	X	X	X	X	X	X	X	X			X	X	X	√
358	**Benchmark**	**#4**	**101**	**101**																							X	X	X	X	X	√
359		**#4**	**101**	**103**																							X	X	X	X	X	√
360		**#4**	**101**	**104**																							X	X	X	X	X	√
361		**#4**	**101**	**201**																							X	X	X	X	X	√
362		**#4**	**101**	**202**																							X	X	X	X	X	√
363		**#4**	**101**	**204**																							X	X	X	X	X	√
364		**#4**	**101**	**205**																							X	X	X	X	X	√
365		**#4**	**101**	**206**																							X	X	X	X	X	√
366		**#4**	**101**	**207**																							X	X	X	X	X	√
367		**#4**	**101**	**208**																							X	X	X	X	X	√
368		**#4**	**101**	**209**																							X	X	X	X	X	√

369		#4	101	210																					X	X	X	X	X	√
370		#4	101	221																					X	X	X	X	X	√
371		#4	101	222																					X	X	X	X	X	√
372		#4	101	223																					X	X	X	X	X	√
373		#4	101	224																					X	X	X	X	X	√
374		#4	101	225																					X	X	X	X	X	√
375		#4	101	228																					X	X	X	X	X	√
376		#4	101	230																					X	X	X	X	X	√
377		#4	101	232																					X	X	X	X	X	√
378		#4	101	233																					X	X	X	X	X	√
379		#4	101	236																					X	X	X	X	X	√
380		#4	101	237																					X	X	X	X	X	√
381		#4	101	238																					X	X	X	X	X	√
382		#4	101	239																					X	X	X	X	X	√
383		#4	101	240																					X	X	X	X	X	√
384		#4	101	241																					X	X	X	X	X	√
385		#4	101	246																					X	X	X	X	X	√
386		#4	101	247																					X	X	X	X	X	√
387		#4	101	248																					X	X	X	X	X	√
388		#4	101	249																					X	X	X	X	X	√
389		#4	101	250																					X	X	X	X	X	√
390		#4	101	251																					X	X	X	X	X	√
391		#4	101	252																					X	X	X	X	X	√
392		#4	101	253																					X	X	X	X	X	√

393		#4	101	254																						X	X	X	X	X	√
394		#4	101	257																						X	X	X	X	X	√
395		#4	101	258																						X	X	X	X	X	√
396		#4	101	259																						X	X	X	X	X	√
397		#4	101	260																						X	X	X	X	X	√
398		#4	101	261																						X	X	X	X	X	√
399		#4	101	262																						X	X	X	X	X	√
400		#4	101	265																						X	X	X	X	X	√
401		#4	101	266																						X	X	X	X	X	√
402		#4	101	301																						X	X	X	X	X	√
403		#4	101	302																						X	X	X	X	X	√
404		#4	101	303																						X	X	X	X	X	√
405		#4	101	304																						X	X	X	X	X	√
406	**Conference**	#4	iasted	sigkdd																						X	X	X	X	X	√
407		#4	ekaw	sigkdd																						X	X	X	X	X	√
408		#4	ekaw	iasted																						X	X	X	X	X	√
409		#4	edas	sigkdd																						X	X	X	X	X	√
410		#4	edas	iasted																						X	X	X	X	X	√
411		#4	edas	ekaw																						X	X	X	X	X	√
412		#4	confOf	sigkdd																						X	X	X	X	X	√
413		#4	confOf	iasted																						X	X	X	X	X	√
414		#4	confOf	ekaw																						X	X	X	X	X	√
415		#4	confOf	edas																						X	X	X	X	X	√
416		#4	Conference	sigkdd																						X	X	X	X	X	√

417		**#4**	**Conference**	**iasted**																						X	X	X	X	X	√
418		**#4**	**Conference**	**ekaw**																						X	X	X	X	X	√
419		**#4**	**Conference**	**edas**																						X	X	X	X	X	√
420		**#4**	**Conference**	**confOf**																						X	X	X	X	X	√
421		**#4**	**cmt**	**sigkdd**																						X	X	X	X	X	√
422		**#4**	**cmt**	**iasted**																						X	X	X	X	X	√
423		**#4**	**cmt**	**ekaw**																						X	X	X	X	X	√
424		**#4**	**cmt**	**edas**																						X	X	X	X	X	√
425		**#4**	**cmt**	**confOf**																						X	X	X	X	X	√
426		**#4**	**cmt**	**Conference**																						X	X	X	X	X	√
427	**MultiFarm cn-pt**	**#4**	**cmt-en**	**cmt-pt**																						X	X	X	X	X	√
428		**#4**	**cmt-en**	**conference-pt**																						X	X	X	X	X	√
429		**#4**	**cmt-pt**	**conference-en**																						X	X	X	X	X	√
430		**#4**	**cmt-en**	**confOf-pt**																						X	X	X	X	X	√
431		**#4**	**cmt-pt**	**confOf-en**																						X	X	X	X	X	√
432		**#4**	**cmt-en**	**edas-pt**																						X	X	X	X	X	√
433		**#4**	**cmt-pt**	**edas-en**																						X	X	X	X	X	√
434		**#4**	**cmt-en**	**ekaw-pt**																						X	X	X	X	X	√
435		**#4**	**cmt-pt**	**ekaw-en**																						X	X	X	X	X	√
436		**#4**	**cmt-en**	**iasted-pt**																						X	X	X	X	X	√
437		**#4**	**cmt-pt**	**iasted-en**																						X	X	X	X	X	√
438		**#4**	**cmt-en**	**sigkdd-pt**																						X	X	X	X	X	√
439		**#4**	**cmt-pt**	**sigkdd-en**																						X	X	X	X	X	√
440		**#4**	**Conference-en**	**Conference-pt**																						X	X	X	X	X	√

441		#4	Conference-en	confOf-pt																						X	X	X	X	X	√
442		#4	Conference-pt	confOf-en																						X	X	X	X	X	√
443		#4	Conference-en	edas-pt																						X	X	X	X	X	√
444		#4	Conference-pt	edas-en																						X	X	X	X	X	√
445		#4	Conference-en	ekaw-pt																						X	X	X	X	X	√
446		#4	Conference-pt	ekaw-en																						X	X	X	X	X	√
447		#4	Conference-en	iasted-pt																						X	X	X	X	X	√
448		#4	Conference-pt	iasted-en																						X	X	X	X	X	√
449		#4	Conference-en	sigkdd-pt																						X	X	X	X	X	√
450		#4	Conference-pt	sigkdd-en																						X	X	X	X	X	√
451		#4	confOf-en	confOf-pt																						X	X	X	X	X	√
452		#4	confOf-en	edas-pt																						X	X	X	X	X	√
453		#4	confOf-pt	edas-en																						X	X	X	X	X	√
454		#4	confOf-en	ekaw-pt																						X	X	X	X	X	√
455		#4	confOf-pt	ekaw-en																						X	X	X	X	X	√
456		#4	confOf-en	iasted-pt																						X	X	X	X	X	√
457		#4	confOf-pt	iasted-en																						X	X	X	X	X	√
458		#4	confOf-en	sigkdd-pt																						X	X	X	X	X	√
459		#4	confOf-pt	sigkdd-en																						X	X	X	X	X	√
460		#4	edas-en	edas-pt																						X	X	X	X	X	√
461		#4	edas-en	ekaw-pt																						X	X	X	X	X	√
462		#4	edas-pt	ekaw-en																						X	X	X	X	X	√
463		#4	edas-en	iasted-pt																						X	X	X	X	X	√
464		#4	edas-pt	iasted-en																						X	X	X	X	X	√

465		**#4**	**edas-en**	**sigkdd-pt**																					X	X	X	X	X	√
466		**#4**	**edas-pt**	**sigkdd-en**																					X	X	X	X	X	√
467		**#4**	**edas-en**	**confOf**																					X	X	X	X	X	√
468		**#4**	**ekaw-en**	**ekaw-pt**																					X	X	X	X	X	√
469		**#4**	**ekaw-en**	**iasted-pt**																					X	X	X	X	X	√
470		**#4**	**ekaw-pt**	**iasted-en**																					X	X	X	X	X	√
471		**#4**	**ekaw-en**	**sigkdd-pt**																					X	X	X	X	X	√
472		**#4**	**ekaw-pt**	**sigkdd-en**																					X	X	X	X	X	√
473		**#4**	**iasted-en**	**iasted-pt**																					X	X	X	X	X	√
474		**#4**	**iasted-en**	**sigkdd-pt**																					X	X	X	X	X	√
475		**#4**	**iasted-pt**	**sigkdd-en**																					X	X	X	X	X	√
476		**#4**	**sigkdd-en**	**sigkdd-pt**																					X	X	X	X	X	√
477	**Benchmark**	**#5**	**101**	**101**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
478		**#5**	**101**	**103**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
479		**#5**	**101**	**104**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
480		**#5**	**101**	**201**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
481		**#5**	**101**	**202**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
482		**#5**	**101**	**204**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
483		**#5**	**101**	**205**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
484		**#5**	**101**	**206**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
485		**#5**	**101**	**207**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
486		**#5**	**101**	**208**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
487		**#5**	**101**	**209**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
488		**#5**	**101**	**210**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√

489		#5	101	221	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
490		#5	101	222	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
491		#5	101	223	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
492		#5	101	224	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
493		#5	101	225	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
494		#5	101	228	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
495		#5	101	230	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
496		#5	101	232	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
497		#5	101	233	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
498		#5	101	236	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
499		#5	101	237	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
500		#5	101	238	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
501		#5	101	239	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
502		#5	101	240	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
503		#5	101	241	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
504		#5	101	246	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
505		#5	101	247	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
506		#5	101	248	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
507		#5	101	249	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
508		#5	101	250	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
509		#5	101	251	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
510		#5	101	252	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
511		#5	101	253	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
512		#5	101	254	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√

513		**#5**	**101**	**257**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
514		**#5**	**101**	**258**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
515		**#5**	**101**	**259**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
516		**#5**	**101**	**260**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
517		**#5**	**101**	**261**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
518		**#5**	**101**	**262**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
519		**#5**	**101**	**265**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
520		**#5**	**101**	**266**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
521		**#5**	**101**	**301**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
522		**#5**	**101**	**302**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
523		**#5**	**101**	**303**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
524		**#5**	**101**	**304**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
525	**Conference**	**#5**	**iasted**	**sigkdd**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
526		**#5**	**ekaw**	**sigkdd**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
527		**#5**	**ekaw**	**iasted**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
528		**#5**	**edas**	**sigkdd**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
529		**#5**	**edas**	**iasted**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
530		**#5**	**edas**	**ekaw**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
531		**#5**	**confOf**	**sigkdd**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
532		**#5**	**confOf**	**iasted**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
533		**#5**	**confOf**	**ekaw**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
534		**#5**	**confOf**	**edas**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
535		**#5**	**Conference**	**sigkdd**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
536		**#5**	**Conference**	**iasted**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√

537		**#5**	**Conference**	**ekaw**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
538		**#5**	**Conference**	**edas**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
539		**#5**	**Conference**	**confOf**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
540		**#5**	**cmt**	**sigkdd**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
541		**#5**	**cmt**	**iasted**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
542		**#5**	**cmt**	**ekaw**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
543		**#5**	**cmt**	**edas**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
544		**#5**	**cmt**	**confOf**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
545		**#5**	**cmt**	**Conference**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
546	**MultiFarm en-pt**	**#5**	**cmt-en**	**cmt-pt**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
547		**#5**	**cmt-en**	**conference-pt**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
548		**#5**	**cmt-pt**	**conference-en**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
549		**#5**	**cmt-en**	**confOf-pt**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
550		**#5**	**cmt-pt**	**confOf-en**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
551		**#5**	**cmt-en**	**edas-pt**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
552		**#5**	**cmt-pt**	**edas-en**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
553		**#5**	**cmt-en**	**ekaw-pt**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
554		**#5**	**cmt-pt**	**ekaw-en**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
555		**#5**	**cmt-en**	**iasted-pt**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
556		**#5**	**cmt-pt**	**iasted-en**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
557		**#5**	**cmt-en**	**sigkdd-pt**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
558		**#5**	**cmt-pt**	**sigkdd-en**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
559		**#5**	**Conference-en**	**Conference-pt**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
560		**#5**	**Conference-en**	**confOf-pt**	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√

561		#5	Conference-pt	confOf-en	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
562		#5	Conference-en	edas-pt	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
563		#5	Conference-pt	edas-en	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
564		#5	Conference-en	ekaw-pt	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
565		#5	Conference-pt	ekaw-en	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
566		#5	Conference-en	iasted-pt	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
567		#5	Conference-pt	iasted-en	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
568		#5	Conference-en	sigkdd-pt	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
569		#5	Conference-pt	sigkdd-en	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
570		#5	confOf-en	confOf-pt	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
571		#5	confOf-en	edas-pt	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
572		#5	confOf-pt	edas-en	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
573		#5	confOf-en	ekaw-pt	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
574		#5	confOf-pt	ekaw-en	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
575		#5	confOf-en	iasted-pt	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
576		#5	confOf-pt	iasted-en	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
577		#5	confOf-en	sigkdd-pt	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
578		#5	confOf-pt	sigkdd-en	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
579		#5	edas-en	edas-pt	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
580		#5	edas-en	ekaw-pt	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
581		#5	edas-pt	ekaw-en	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
582		#5	edas-en	iasted-pt	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
583		#5	edas-pt	iasted-en	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
584		#5	edas-en	sigkdd-pt	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√

585		#5	edas-pt	sigkdd-en	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
586		#5	edas-en	confOf	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
587		#5	ekaw-en	ekaw-pt	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
588		#5	ekaw-en	iasted-pt	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
589		#5	ekaw-pt	iasted-en	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
590		#5	ekaw-en	sigkdd-pt	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
591		#5	ekaw-pt	sigkdd-en	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
592		#5	iasted-en	iasted-pt	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
593		#5	iasted-en	sigkdd-pt	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
594		#5	iasted-pt	sigkdd-en	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√
595		#5	sigkdd-en	sigkdd-pt	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	√