

Análise de abordagens populacionais para meta-alinhamento de ontologias

Jairo de Souza¹, Sean Siqueira², Rubens Melo³, Carlos Lucena³

¹Departamento de Ciência da Computação – Instituto de Ciências Exatas
Universidade Federal de Juiz de Fora (UFJF) – Juiz de Fora, MG – Brasil

²Departamento de Informática Aplicada
Universidade Federal do Estado do Rio de Janeiro (UNIRIO)
Rio de Janeiro, RJ – Brasil

³Departamento de Informática
Pontifícia Universidade Católica do Rio de Janeiro (PUC-RJ)
Rio de Janeiro, RJ – Brasil

jairo.souza@ufjf.edu.br, sean@uniriotec.br, {rubens,lucena}@inf.puc-rio.br

Abstract. *Interoperability is an important requirement in information systems, mainly the semantic interoperability. Ontologies have been used for reaching semantic interoperability, but when systems use two or more ontologies it is necessary to consider techniques that allow aligning distinct ontologies. Ontology meta-matching (OMM) is a recent approach for ontology alignment. OMM addresses issues such as aligner aggregation and self-configuration. The choice of a good OMM approach is not trivial and it is necessary to analyze the characteristics of the solutions. In this work, we detail and compare the more recent OMM approaches and analyze their differences.*

Resumo. *A interoperabilidade é um requisito importante num ambiente de sistemas de informação, destacando-se a interoperabilidade semântica. Ontologias tem sido utilizadas para promover a interoperabilidade semântica, mas num ambiente onde coexistem mais de uma ontologia, é necessária a utilização de técnicas que permitam alinhar as diferentes ontologias para permitir a troca de informação entre os sistemas. Dentre as técnicas de alinhamento mais atuais, está o meta-alinhamento de ontologias, o qual permite a agregação e autoconfiguração de alinhadores. A escolha por uma abordagem de meta-alinhamento não é trivial e faz-se necessário analisar diferentes características das soluções. Esse trabalho apresenta e compara as abordagens atuais de meta-alinhamento de ontologias, apontando suas qualidades e deficiências.*

1. Introdução

Num ambiente de sistemas de informação, onde é necessário a troca de dados entre sistemas, a interoperabilidade é um requisito importante. Há vários níveis de heterogeneidade, dos quais se destaca a heterogeneidade semântica [Euzenat and Shvaiko 2007]. De modo a promover a interoperabilidade semântica, isto é, a capacidade de dois ou mais sistemas heterogêneos e distribuídos trabalharem em conjunto, compartilhando as informações com entendimento comum de seu significado [Buranarach 2001], ontologias costumam

ser utilizadas, pois permitem um compartilhamento e reuso de conhecimento e de dados de forma mais efetiva. Entretanto, atualmente existem diversas ontologias disponíveis na web e ontologias mais específicas são criadas para cada domínio de aplicação. Assim, para sistemas que fazem uso de ontologias distintas, a utilização de ontologias não reduz a heterogeneidade, mas leva os problemas de heterogeneidade para um nível mais alto [Euzenat and Shvaiko 2007].

Para alcançar a interoperabilidade semântica num ambiente de sistemas de informação que lidam com mais de uma ontologia, são utilizadas técnicas para compatibilizar ontologias. Para compatibilizar ontologias, pode-se utilizar técnicas de alinhamento. Alinhamento de ontologias é o processo de determinar correspondências entre entidades de dois ou mais modelos [Giunchiglia et al. 2007].

O processo de alinhamento é uma importante operação em aplicações tradicionais, como integração de modelos ou *data warehouses* [Do 2006]. Essas aplicações são caracterizadas por modelos com estruturas heterogêneas que são analisadas e alinhadas de forma manual ou semiautomática, geralmente durante o tempo de projeto. Em tais aplicações, o alinhamento é um pré-requisito para o funcionamento do sistema. Outros sistemas são caracterizados por sua dinâmica, como sistemas multiagentes [Santos et al. 2011], sistemas *peer-to-peer* [Staab and Stuckenschmidt 2006], serviços web [Fensel et al. 2011] e outras aplicações que consomem dados da web, como as baseadas em dados ligados [Souza et al. 2012]. Tais aplicações, ao contrário das aplicações tradicionais, requerem uma operação de alinhamento em tempo de execução.

Alinhadores alcançam melhores resultados em alguns cenários que em outros. Assim, fazer uso de diversos alinhadores e descobrir quais são melhores aplicados para cada caso é um desafio para o campo de alinhamento de ontologias [Shvaiko and Euzenat 2008]. Somado a esse desafio, encontra-se a necessidade de reavaliar os alinhadores utilizados em ambientes dinâmicos, como a web. É natural que, nesse tipo de ambiente, as características das aplicações estejam constantemente em mudança. Assim, as abordagens que tentam calibrar e adaptar automaticamente soluções de alinhamento para as configurações nas quais uma aplicação opera são de grande importância. Entre as abordagens atuais para resolver esses problemas, está o meta-alinhamento de ontologias, ou seja, um *framework* para combinar um conjunto de alinhadores de ontologias escolhidos.

Embora existam algumas propostas recentes para resolver o meta-alinhamento de ontologias, existem poucos trabalhos que detalham e analisam as diferenças entre as abordagens. Soluções para meta-alinhamento de ontologias tem sido criadas e estas geralmente fazem uso de meta-heurísticas ou de aprendizado de máquina. Contudo, entre as soluções mais promissoras para meta-alinhamento de grandes ontologias estão as soluções que fazem uso de meta-heurísticas baseadas em populações, como algoritmos genéticos, enxame de abelhas e outras. Estas são promissoras para grandes volumes de dados pois podem ser facilmente paralelizáveis, permitem que a execução possa ser pausada e restaurada posteriormente do mesmo ponto. Além disso, estas abordagens permitem que a execução seja interrompida e que o resultado calculado até aquele momento possa ser utilizado, uma vez que os valores são refinados a cada iteração.

Assim, esse artigo objetiva apresentar, analisar e comparar as abordagens popu-

lacionais recentes para meta-alinhamento de ontologias através de características destacadas na literatura. Para tal, na seção 2 é apresentada a definição do problema de meta-alinhamento de ontologias. Na seção 3 são apresentadas as abordagens heurísticas baseadas em populações para resolver o problema de meta-alinhamento de ontologias. Em seguida, as abordagens são comparadas na seção 4. Por fim, apresentamos nossas conclusões na seção 5.

2. Meta-alinhamento de ontologias

Embora várias técnicas para alinhar ontologias venham sendo propostas na literatura, não há uma técnica que seja completamente eficaz em todos os casos. Isso se dá porque, para encontrar boas técnicas, é necessário ter conhecimento do contexto em que as técnicas serão aplicadas, dos dados disponíveis e das diferenças presentes nos modelos a serem analisados. Assim, é necessário reconsiderar as técnicas escolhidas a cada caso. Contudo, a utilização conjunta e coordenada de técnicas distintas, preferencialmente complementares, pode ajudar na obtenção de melhores alinhamentos entre ontologias. Neste contexto, são utilizados meta-alinhadores de ontologias.

2.1. Meta-alinhamento de ontologias

A expressão Meta-Alinhamento de Ontologias foi introduzida por [Euzenat and Shvaiko 2007] para nomear sistemas que tentam configurar automaticamente funções de alinhamento de ontologias. Em seguida, diversos trabalhos foram publicados para tratar esse problema. Em geral, existem algumas características em comum para as estratégias de meta-alinhamento [Martinez-Gil and Aldana-Montes 2012]:

1. Não é necessário que o processo de meta-alinhamento seja realizado em tempo de execução. As funções de alinhamento podem ser computadas em *background* e serem aplicadas em tempo de execução;
2. Este deve ser um processo automático. Assim, deve ser possível que o processo seja implementado por alguma ferramenta de alinhamento.
3. O processo deve retornar a melhor função de alinhamento possível. Caso esta função não seja conhecida, o processo deve retornar uma função mais próxima possível da melhor função de alinhamento, se comportando como uma especialista que experimenta diversas combinações e pesos de funções de alinhamento.
4. Uma estratégia de meta-alinhamento é avaliada com a função de alinhamento retornada.

Na literatura, as expressões combinação de alinhadores (*matcher combination*), auto-adaptação de alinhadores (*matcher self-tuning*) e meta-alinhamento (*meta-matching*) podem causar alguma confusão. Combinação de alinhadores diz respeito à ordem de combinação de certos alinhadores pertencentes a uma biblioteca de alinhadores. Esta tarefa aumenta a complexidade do problema de alinhamento, uma vez que é necessário identificar quais alinhadores devem ser combinados e de que forma essa combinação deve ser realizada. Segundo [Martinez-Gil and Aldana-Montes 2012], este processo, atualmente, só pode ser realizado em tempo de projeto por ferramentas especializadas. A auto-adaptação de alinhadores é a tentativa de calibrar e adaptar automaticamente soluções de alinhamento para as configurações de estabilidade e desempenho que a aplicação opera. Esse processo geralmente ocorre em tempo de execução, onde, por exemplo, os sistemas podem escolher alinhadores mais rápidos ao receber ontologias muito grandes. O

meta-alinhamento de ontologia diz respeito à combinação de um conjunto de alinhadores heterogêneos. O principal objetivo desse processo é encontrar valores apropriados para seus pesos, pontos de cortes (*thresholds*) e quaisquer outros parâmetros que possam afetar os resultados do alinhamento. Ao contrário da auto-adaptação de alinhadores, o objetivo principal de um meta-alinhador não é manter um sistema funcionando de forma efetiva, mas gerar uma boa função de alinhamento de ontologias.

Meta-alinhadores podem realizar tarefas pré-alinhamento ou pós-alinhamento. As tarefas pré-alinhamento são realizadas automaticamente e podem não ser realizadas em tempo de execução. Estas tarefas consistem na seleção e treinamento dos alinhadores, além da configuração de parâmetros. Tarefas pós-alinhamento consistem na identificação de falsos positivos e falsos negativos. Uma vez que tenta-se evitar a intervenção do usuário, algumas técnicas utilizam estratégias de consulta a bases externas para melhorar os resultados gerados e checar consistências, por exemplo, como as medidas web propostas por [Gracia and Mena 2008].

Segundo [Martinez-Gil and Aldana-Montes 2009], quanto às técnicas utilizadas para resolver o problema, pode-se classificar as abordagens de meta-alinhamento em dois grandes grupos: o meta-alinhamento baseado em aprendizado de máquina e o meta-alinhamento heurístico.

Abordagens baseadas em aprendizado de máquina utilizam: avaliação de relevância dos alinhamentos iniciais através da interação com o usuário [Ehrig et al. 2005]; aprendizado bayesiano para capturar interdependências entre os alinhadores e, assim, tentar melhorar o modo como esses alinhadores são combinados [Duchateau et al. 2009]; árvores de decisão para decidir quais alinhadores utilizar com base em critérios definidos pelo usuário, como a redução do custo de processamento [Duchateau et al. 2008]; e, por fim, treinamento de redes neurais utilizando diferentes bases de testes para, então, utilizar o conhecimento para predizer novas funções de similaridade [Huang et al. 2007, Mao et al. 2008, Spohr et al. 2011]. Conforme apontado por [Martinez-Gil and Aldana-Montes 2012], as abordagens baseadas em aprendizado de máquina geralmente são utilizadas em tempo de projeto, uma vez que demandam considerável processamento na fase de treinamento da abordagem.

Abordagens heurísticas para meta-alinhamento de ontologias utilizam diversas técnicas como algoritmos evolutivos, algoritmos baseados em regras e algoritmos gulosos. As abordagens heurísticas baseadas em população são facilmente paralelizáveis, permitindo alinhamento de ontologias em larga escala, e têm apresentado resultados promissores. Para permitir uma análise mais detalhada das abordagens de meta-alinhamento, este trabalho se restringirá às abordagens baseadas em população. As abordagens propostas na literatura serão apresentadas e discutidas com detalhes na seção 3. Antes, apresentamos as definições do problema de meta-alinhamento e uma breve explanação sobre meta-heurísticas baseadas em populações.

2.2. Definições

Linguagens para descrição de ontologias permitem a especificação de diferentes tipos de entidades¹, como classes, indivíduos, relações, tipos de dados e valores. Classes (ou con-

¹O termo entidade, neste contexto, possui um significado mais amplo do que quando utilizado na área de Banco de Dados em modelos como Entidade-Relacionamento.

ceitos) são as entidades principais de uma ontologia, as quais são interpretadas como um conjunto de indivíduos do domínio. Indivíduos (ou objetos, ou instâncias) são interpretados como objetos específicos de um domínio, instâncias de uma classe. Indivíduos, bem como classes, podem se relacionar com outros indivíduos ou classes através de diversos tipos de relações.

Tipos de dados especificam conjuntos de valores, ao contrário de indivíduos, contudo, valores não possuem identidades. Valores são instâncias de tipos primitivos. Por exemplo, “Pet Sounds” é um valor que pode ser o título de um *álbum musical*.

Classes não necessitam ser nomeadas e podem participar de construções lógicas como especialização, exclusão, instanciação e atribuição.

A especialização entre duas classes ou duas propriedades é interpretada como a inclusão das interpretações dessas entidades. Por exemplo, a classe *Carro* é uma especialização da classe *Veículo*, o que significa dizer que a noção de *Carro* está incluída na noção de *Veículo*, isto é, $Carro \subset Veículo$. Vale ressaltar que a relação de inclusão traduz, na linguagem da teoria dos conjuntos, a operação lógica de implicação de proposições. De fato, dizer que $A \subset B$ é o mesmo que dizer que a proposição “ $x \in A$ ” implica a proposição “ $x \in B$ ”. Tem-se então $A \subset B$ sse $\forall x \in \mathbb{U}, x \in A \Rightarrow x \in B$.

A disjunção entre duas classes ou propriedades é interpretada como a exclusão das interpretações dessas entidades, isto é, quando sua interseção é vazia. Por exemplo, a classe *Livro* pode ser declarada como exclusiva para a classe *Pessoa*. Assim, a relação de disjunção entre A e B implica que A é exclusivo para B sse $\forall x, y \in \mathbb{U}, (x \in A \Rightarrow x \notin B) \wedge (y \in B \Rightarrow y \notin A)$.

A instanciação ou tipagem entre indivíduos e classes, instâncias de propriedades e propriedades, valores e tipos de dados é interpretada como associação. Por exemplo, o indivíduo *Rio de Janeiro* é uma instância da classe *Cidade*.

Por fim, a atribuição é uma relação entre instâncias de propriedades, indivíduos e valores, a qual associa um valor a uma propriedade de um dado indivíduo. Por exemplo, é atribuído o valor 1565 à propriedade *Data de Fundação* do indivíduo *Rio de Janeiro*. Em resumo, podemos considerar uma ontologia como na definição 2.1 [Euzenat and Shvaiko 2007]:

Definição 2.1 (Ontologia) *Seja O o conjunto de ontologias, onde $o \in O$ é um tupla $o = \langle C, I, R, T, V, \leq, \perp, \in, = \rangle$, tal que:*

C é o conjunto de classes;

I é o conjunto de indivíduos;

R é o conjunto de relações;

T é o conjunto de tipos de dados;

V é o conjunto de valores;

\leq é uma relação em $(C \times C) \cup (R \times R) \cup (T \times T)$ chamada especialização;

\perp é uma relação em $(C \times C) \cup (R \times R) \cup (T \times T)$ chamada exclusão;

\in é uma relação sobre $(I \times C) \cup (V \times T)$ chamada instanciação;

= é uma relação sobre $I \times R \times (I \cup V)$ chamada atribuição.

O objetivo das técnicas de alinhamento de ontologias é encontrar correspondências entre entidades definidas em diferentes ontologias. Quase sempre, estas correspondências são relações de equivalência, que são descobertas através de medidas de similaridade entre as entidades das ontologias. O modo mais comum de avaliar a similaridade entre duas entidades é definindo uma medida de similaridade, conforme a definição 2.2.

Definição 2.2 (Similaridade) *Seja κ o conjunto de todos os pares (o_i, o_j) definidos em $O \times O$, uma similaridade $\sigma : \kappa \rightarrow \mathbb{R}$ é uma função que avalia um par de entidades $x \in o_i$ e $y \in o_j$ e mapeia para um número real que expressa o grau de similaridade entre dois objetos tal que:*

$$\begin{aligned} \forall x, y \sigma(x, y) &\geq 0 && \text{(positividade)} \\ \forall z \in o_j, \sigma(x, x) &\geq \sigma(y, z) && \text{(a)} \\ \forall x, y \sigma(x, y) &= \sigma(y, x) && \text{(simetria)} \end{aligned}$$

Uma vez que uma função de similaridade é aplicada sobre entidades de ontologias, podem-se definir funções de similaridade específicas para combinação de tipos de entidades, por exemplo, classes e classes, relações e relações, classes e relações etc. Como cada função de similaridade é aplicada sobre algumas entidades e, ainda, cada função de similaridade utiliza técnicas diferentes de análise, a escolha da função de similaridade mais adequada para avaliar duas ontologias torna-se dependente da estrutura das ontologias dadas. Contudo, a análise de duas ontologias pode ser feita, de forma composta, por várias funções de similaridade, gerando uma função composta, conforme a definição 2.3.

Definição 2.3 (Similaridade composta) *Uma similaridade composta $\delta : o \times o \rightarrow \mathbb{R}$ é uma função de similaridade dada pela soma de similaridades $\sigma_1 + \sigma_2 + \dots + \sigma_n$, onde σ_i indica uma função de similaridade sobre entidades de o e $i \in [1..n]$.*

Definição 2.4 (Similaridade normalizada) *Uma similaridade composta é dita normalizada se seu valor varia no intervalo de números reais $[0,1]$, onde o valor 1 representa uma similaridade perfeita, o valor 0 representa nenhuma similaridade e valores intermediários representam a probabilidade das entidades serem similares. Uma versão normalizada de uma similaridade composta δ é denotada como $\bar{\delta}$.*

Novas funções de similaridade compostas podem ser definidas utilizando outras funções de similaridade, compostas ou não. Assim, podemos definir uma função de similaridade composta como, por exemplo, $\delta''(a, b) = \sigma(a, b) + \delta'(a, b)$.

Para normalizar funções de similaridade, pode-se utilizar pesos associados a cada função membro da composição. Cada peso representa a relevância da função de similaridade para o cálculo da similaridade composta.

Definição 2.5 (Composição com pesos) *Sejam o_1 e o_2 duas ontologias e $F = g_1 \circ g_2 \circ \dots \circ g_n$ uma função de similaridade composta por n funções membros. Dadas duas entidades a e b onde $a \in o_1, b \in o_2$, seja $g_i(a, b)$ a i -ésima função de similaridade membro de $F(a, b)$, k um valor inteiro não negativo e os pesos p_i , então a função normalizada $\bar{F}(a, b)$ pode ser definida como*

$$\bar{F}(a, b) = \sum_{i=1}^n g_i(a, b)p_i, \text{ onde } \sum_{i=1}^n p_i \leq k, \quad (1)$$

sendo $p_i \in \mathbb{R}$.

A partir do retorno de uma ou mais funções membros de similaridade, alinhadores podem decidir qual a devida correspondência entre pares de entidades e gerar o alinhamento entre as ontologias. Pode-se definir alinhamentos conforme a definição 2.6, adaptada de [Shvaiko and Euzenat 2005].

Definição 2.6 (Alinhamento) *Dadas duas ontologias o e o' , alinhamento é um conjunto de correspondências entre pares de entidades $\langle e, e' \rangle$ pertencentes a o e o' , respectivamente. Uma correspondência é descrita como uma quádrupla $\langle e, e', r, n \rangle$ onde:*

- *e e e' são as entidades (por exemplo, termos, classes, indivíduos) sobre as quais uma relação é afirmada pela correspondência.*
- *r é a relação, entre e e e' , afirmada pela correspondência, podendo ser uma relação da teoria dos conjuntos, uma relação fuzzy, uma medida de similaridade, etc.*
- *n é o grau de confiança na correspondência, onde $n \in [0, 1]$. Deve-se observar que este grau não se refere à relação r , mas à medida de confiança no fato de que a correspondência é verdadeira.*

O processo de alinhamento de ontologias realizado por um alinhador pode ser descrito como uma função F_a onde, dado um par de ontologias o e o' , um alinhamento pré-existente A , um conjunto de parâmetros p e um conjunto de recursos r , um alinhamento A' é retornado [Bouquet et al. 2004], conforme a equação 2.

$$A' = F_a(o, o', A, p, r) \quad (2)$$

Ressalta-se que os conjuntos A , p e r podem ser vazios. Dessa forma, um alinhador pode utilizar um processo que gera alinhamentos sem a necessidade de alinhamentos pré-existentes, ou sem a possibilidade de configuração do alinhador ou, ainda, sem a utilização de outros recursos, como base de dados externas, outras ontologias, documentos e *feedback* do usuário [Souza et al. 2010].

Um alinhador pode ser avaliado quanto à qualidade dos alinhamentos gerados. Verificar a qualidade de alinhamentos significa verificar se o alinhamento possui correspondências corretas. Esse processo de avaliação pode ser realizado por um usuário especialista no domínio. É possível definir medidas de avaliação de alinhadores, caso exista um alinhamento de referência conhecido. Essas medidas podem avaliar alinhamentos através da comparação entre as correspondências obtidas pelo alinhador e as correspondências esperadas.

Definição 2.7 (Avaliação de alinhamentos) *Uma avaliação de alinhamentos ε é uma função $\varepsilon : A \rightarrow [0, 1]$ que associa um alinhamento A a um número real que representa a qualidade do alinhamento A em relação ao alinhamento correto entre o e o' .*

Segundo [Euzenat and Shvaiko 2007], meta-alinhadores são sistemas nos quais a originalidade está no modo em que estes sistemas utilizam e combinam outros alinhadores. Meta-alinhadores utilizam variadas técnicas para encontrar um bom alinhamento entre ontologias a partir do resultado de outros alinhadores. Para o contexto desse trabalho, define-se uma função de meta-alinhamento conforme a definição 2.8 e um meta-alinhador conforme a definição 2.9.

Definição 2.8 (Função de meta-alinhamento) *Seja $S \subset \mathbb{R}$ um conjunto de valores que indicam todos os graus de similaridade calculados obtidos a partir da aplicação de uma função de similaridade. Uma função de meta-alinhamento h é uma função $h : S \mapsto \mathbb{R}$ que define a relevância de cada grau de similaridade previamente calculado $s \in S$. O resultado obtido é um grau de similaridade otimizado $s_o \in \mathbb{R}$. Chamamos de grau de similaridade otimizado o melhor valor possível encontrado, ou seja, o grau de similaridade mais próximo de 1.*

Definição 2.9 (Meta-alinhador) *Um meta-alinhador é uma função $\Psi : o \times o' \xrightarrow{h} A$ que associa duas ontologias de entrada o e o' a um alinhamento A fazendo uso de uma função de meta-alinhamento h , de tal forma que $\varepsilon(\Psi(o, o'))$ seja o mais próximo possível de 1.*

Na prática, meta-alinhamento é a técnica de selecionar os algoritmos, correspondências e pesos mais apropriados em diferentes cenários de alinhamento com o objetivo de obter um alinhamento satisfatório entre ontologias [Martinez-Gil and Aldana-Montes 2012].

2.3. Meta-heurísticas baseadas em populações

Algoritmos exatos de aproximação não são capazes de fornecer uma solução apropriada na resolução de problemas de aproximação que possuem um espaço de busca com muitas dimensões [Beheshti and Shamsuddin 2013]. Em problemas como o meta-alinhamento de ontologias, o espaço de busca cresce conforme novas características são adicionadas ao problema: quantidade de alinhadores, possibilidade de mapeamentos complexos, autoconfiguração de parâmetros específicos de cada alinhador, entre outros. Uma exploração exaustiva do espaço de soluções não é uma estratégia prática. Algoritmos meta-heurísticos mais adaptáveis e que podem explorar espaços muito vasto de soluções candidatas têm sido desenvolvidos para resolver problemas de otimização [Haupt et al. 2004] e vêm sendo adotados para resolver o problema de meta-alinhamento de esquemas. Entre esses algoritmos, os algoritmos meta-heurísticos baseados em populações são adequados para realizar buscas globais devido à habilidade de aprimoramento local e exploração global.

São chamadas de meta-heurísticas baseadas em populações aquelas que realizam a busca partindo de múltiplos pontos iniciais, mantendo e melhorando as múltiplas soluções candidatas, quase sempre usando características da população para guiar a busca. Estes algoritmos possuem características que os fazem adequados para o problema de alinhamento de ontologias:

- Métodos baseados em populações podem ser facilmente paralelizáveis [Whitley 2001], uma vez que os pontos podem ser avaliados de forma independente em cada nó paralelo. Assim, alinhamentos em larga escala podem ser realizados fazendo uso de uma infra-estrutura paralela, o qual permite um resultado mais rápido.

- Seu comportamento permite que a execução do algoritmo seja interrompida a qualquer momento para retornar a melhor solução encontrada até aquele instante. Esta pode ser uma propriedade desejável em cenários de alinhamento em tempo de execução [Bock et al. 2012].
- Estes métodos funcionam de forma incremental, o que permite que o algoritmo se inicie com uma configuração inicial padrão ou uma configuração parcial pré-calculada que vai sendo refinada ao longo da execução do algoritmo.
- Métodos baseados em população consideram a qualidade da solução completa. Assim, no caso de alinhamento de ontologias é verificado a qualidade de todo o alinhamento e não somente de uma única correspondência [Bock et al. 2011].
- Estes métodos funcionam como meta-heurísticas, ou seja, são independentes da função objetivo que será otimizada. Em relação ao alinhamento de ontologias, isto significa que a função objetivo pode ser ajustada de acordo com a especificidade do cenário em que se está trabalhando.

São exemplos de meta-heurísticas baseadas em populações os algoritmos genéticos, programação genética, programação evolucionária, enxame de partículas, colônia de formigas, colônia de abelhas, entre outras [Beheshti and Shamsuddin 2013].

3. Meta-alinhamento de ontologias baseado em populações

Para identificar as abordagens a serem analisadas, o seguinte processo foi realizado. Primeiramente, a maioria das abordagens baseadas em populações para meta-alinhamento de ontologias presentes na literatura foram identificadas. Em seguida, foram selecionadas as abordagens que apresentavam uma descrição formal da sua função objetivo e que utilizaram um *benchmark* para avaliação da abordagem. As ferramentas que atenderam a esses requisitos foram o GAOM [Wang et al. 2006], GOAL [Martinez-Gil and Aldana-Montes 2009], GNoSIS+ [Souza 2012], MapPSO [Bock and Hettenhausen 2012], MapEVO [Bock et al. 2012] e as abordagens de [Gínsca and Iftene 2010] e [Acampora et al. 2012]. Estas abordagens são descritas nesta seção.

3.1. Ferramenta GAOM

O GAOM (*Genetic Algorithm based Ontology Matching*) foi o primeiro meta-alinhador de ontologias baseado em algoritmo genético. O meta-alinhamento é modelado como uma otimização de um mapeamento entre duas ontologias. O problema é tratado como um processo de combinação de características. Cada ontologia é caracterizada por um conjunto de características extensionais e intensionais e, em seguida, é realizada uma busca para encontrar a melhor combinação. Características intensionais de um conceito são definidas como uma tupla (n, p, I) que descreve a essência do conceito, onde n é o nome do conceito, p é o conjunto de propriedades relacionadas com o conceito e I é o conjunto de instâncias associadas ao conceito. Características extensionais de um conceito é definido como o conjunto R de relações do conceito com outros conceitos da ontologia.

Na sua abordagem genética, o GAOM representa cada indivíduo como um vetor numérico de tamanho n_1 para armazenar valores entre 1 e n_2 , onde n_1 é a quantidade de entidades de uma ontologia o_1 e n_2 é a quantidade de entidades de uma ontologia o_2 .

Assim, cada indivíduo M é denotado como uma sequência de valores N_i , onde $N_i \in \{1, 2, \dots, n_2\}$, $i \in \{1, 2, \dots, n_1\}$ e a representação significa que o i -ésimo conceito em o_1 é mapeado para N_i -ésimo conceito em o_2 .

Uma vez que uma ontologia é representada como uma coleção de características, o GAOM calcula a similaridade entre duas ontologias como um processo de combinação de características. Assim, a função objetivo utilizada mede a similaridade global entre duas ontologias o_1 e o_2 conforme a equação 3.

$$S_{o_1, o_2}(M) = \frac{f(C)}{f(C) + \alpha f((F_{o_1} - F_{o_2})|M) + \beta f((F_{o_2} - F_{o_1})|M)} \quad (3)$$

Na equação 3, M é um indivíduo da população que corresponde a um mapeamento; F_{o_1} e F_{o_2} são conjuntos de características de uma ontologia o_1 e o_2 , respectivamente (pode-se considerar F_i como o conjunto de conceitos da ontologia o_i caso esteja-se mapeamento somente conceitos); $C = (F_{o_1} \cap F_{o_2})|M$, o qual denota o conjunto de características mapeadas de F_{o_1} e F_{o_2} em relação ao mapeamento M ; $(F_{o_1} - F_{o_2})|M$ e $(F_{o_2} - F_{o_1})|M$ são dois conjuntos de características não mapeadas em relação ao mapeamento M ; α e β são dois parâmetros entre 0 e 1 que determinam a relevância dos dois conjuntos de características não mapeadas; f é uma função definida como a cardinalidade do conjunto.

Dado duas ontologias e um mapeamento M , ao computar a aptidão do indivíduo com a equação 3, o GAOM verifica o quanto o mapeamento M contribui para o mapeamento de todas as entidades das duas ontologias. A melhor solução para o GAOM é aquela que representa o mapeamento de todas as entidades das duas ontologias.

3.2. Ferramenta GOAL

O GOAL (*Genetics for Ontology Alignments*) é um algoritmo genético para calibrar alinhadores automaticamente utilizando alinhamentos fornecidos por especialistas. Cada indivíduo é representado como um vetor numérico onde cada gene determina o peso associado a um alinhador. O GOAL trata o problema de calibragem como uma otimização de uma medida de qualidade utilizada na área de Recuperação de Informação.

O GOAL permite a utilização de quatro funções objetivos, embora somente uma possa ser utilizada por vez. Cada função objetivo avalia o indivíduo em relação ao alinhamento esperado para verificar a precisão, cobertura, medida-F ou *fall-out*. Uma vez que estas medidas são comumente utilizadas por meta-alinhadores, iremos defini-las abaixo:

Definição 3.1 (Precisão) *Dado um alinhamento de referência A e um alinhamento obtido B, a precisão é uma medida da habilidade de fornecer alinhamentos corretos dentre os alinhamentos obtidos, calculada como:*

$$\frac{|A \cap B|}{|B|} \quad (4)$$

Definição 3.2 (Cobertura) *Dado um alinhamento de referência A e um alinhamento obtido B, a cobertura é uma medida da habilidade de fornecer alinhamentos corretos dentre o total de alinhamentos de referência, calculada como:*

$$\frac{|A \cap B|}{|A|} \quad (5)$$

Definição 3.3 (Medida-F) Dada uma precisão P e uma cobertura R , a medida-F é a média harmônica entre P e R dada por

$$\frac{2PR}{P + R} \quad (6)$$

Definição 3.4 (Fall-out) Dado um alinhamento de referência A e um alinhamento obtido B , o fall-out é a razão entre fornecer alinhamentos incorretos dentre o total de alinhamentos obtidos, calculada como:

$$\frac{|B| - |A \cap B|}{|B|} \quad (7)$$

Como o GOAL necessita conhecer o alinhamento esperado para que possa funcionar corretamente, é necessário que os especialistas de domínio forneçam casos de teste suficientes para que possa ser realizada, por exemplo, a maximização da precisão ou a minimização dos falsos positivos no alinhamento gerado.

Nesta abordagem, a avaliação de um indivíduo I da população consiste na geração de um alinhamento A' utilizando um conjunto de alinhadores pré-determinados e aplicando a estes alinhadores os pesos representados em I . Com o alinhamento gerado, a função objetivo escolhida avalia cada correspondência em A' em relação às correspondências ao alinhamento de referência A . Diferente de algumas abordagens que utilizam algoritmos genéticos, o GOAL é capaz de encontrar soluções muito boas para o problema de calibragem, uma vez que o algoritmo tem conhecimento dos alinhamentos de referência. Por outro lado, o custo computacional para avaliar cada indivíduo da população é alto, uma vez que cada avaliação pressupõe a execução de todos os alinhadores para gerar um novo alinhamento e uma comparação entre as ocorrências contidas no alinhamento gerado e no alinhamento de referência.

3.3. Abordagem de [Gînsca and Iftene 2010]

Em [Gînsca and Iftene 2010], é apresentada uma abordagem genética para o problema de meta-alinhamento que, segundo os autores, guarda muitas similaridades com a abordagem desenvolvida pelo sistema GOAL (seção 3.2). A principal diferença entre as abordagens está no fato de que, enquanto o GOAL tenta otimizar uma agregação de alinhadores, esta abordagem tenta otimizar outros parâmetros do sistema, como o ponto de corte (*threshold*).

Pode-se, contudo, apontar outras características importantes desta abordagem. Em primeiro lugar, a abordagem tenta diminuir o tempo de processamento do meta-alinhamento ao inserir uma fase de pré-processamento. Nesta fase de pré-processamento, os alinhadores que serão utilizados para alinhar as ontologias são executados e os resultados gerados por cada alinhador para o par de ontologias de entrada é armazenado em um arquivo XML. Durante a execução do algoritmo genético, sendo necessário recuperar o valor de alguma similaridade, o valor é recuperado do arquivo XML ao invés de ser recalculado.

Um cromossomo é definido como uma sequência de bits que representa uma coleção de correspondências entre os conceitos das duas ontologias e um valor de ponto de corte. Assim, cada cromossomo representa um alinhamento e seu ponto de corte. Assim como no sistema GOAL, os autores permitem a utilização de uma das quatro medidas de qualidade do alinhamento utilizadas pelo GOAL para avaliar a aptidão do cromossomo. Assim, é necessário que o algoritmo genético conheça o alinhamento de referência para que a função objetivo possa guiar o comportamento do algoritmo.

3.4. Abordagem de [Acampora et al. 2012]

Uma abordagem heurística para resolver o problema de meta-alinhamento de ontologias como um problema de otimização é apresentada em [Acampora et al. 2012]. Os autores utilizam algoritmos meméticos [Moscato and Cotta 2003]. Algoritmos meméticos são métodos evolucionários híbridos baseados em metaheurísticas populacionais e na aplicação de métodos de melhoramentos locais da solução. Neste método, cada indivíduo passa por um refinamento local dentro do espaço de busca, de modo que o indivíduo pode ter seu nível de aptidão aumentado após passar pela etapa de refinamento.

Os autores utilizam uma função objetivo $F(A) = \sum_{i=1}^{|A|} f(c_i)$, onde A é o conjunto de correspondências c_i entre as duas ontologias alinhadas e $c_i \in A$. A função f é denominada função de adequação, a qual associa um valor no intervalo $[0, 1]$ para cada ocorrência c_i no alinhamento A . A função f é utilizada para avaliar a qualidade de uma correspondência para alcançar o alinhamento ótimo. Esta função é calculada como uma agregação utilizando pesos e uma coleção de medidas de similaridade, como mostra a equação 8.

$$f(c_i) = \phi(\vec{s}(c_i), \vec{w}) \quad (8)$$

onde a função ϕ representa uma estratégia de agregação, a qual combina o vetor de medidas de similaridades \vec{s} considerando o vetor de pesos \vec{w} .

Como restrição do trabalho, os autores somente consideram alinhadores que implementam medidas de distância. Dessa forma, o problema de meta-alinhamento é modelado como uma minimização da função objetivo, onde, conseqüentemente, um valor de $F(A)$ próximo de zero corresponde a um alinhamento A próximo do alinhamento ótimo.

Para representação dos indivíduos, cada cromossomo S representa um alinhamento A entre duas ontologias o_1 e o_2 . Assim $S = \{(e_0, e_{j_0}), (e_1, e_{j_1}), \dots, (e_h, e_{j_h})\}$, onde $h = |o_1| - 1$ e $j_l \in \{0, 1, 2, \dots, |o_2| - 1\}$ com $l = 0, 1, 2, \dots, h$. Com isso, para avaliar a aptidão de cada indivíduo, é necessário recalculer a função de adequação f (equação 8) para cada indivíduo, o que, por consequência, exige o cálculo de cada medida de similaridade para cada gene do indivíduo a cada avaliação.

Como critério de parada, os autores adotam as mesmas medidas de qualidade adotadas pelo sistema GOAL (seção 3.2), sendo necessário que o algoritmo memético tenha conhecimento do alinhamento de referência para retornar bons alinhamentos.

3.5. GNoSIS+

O GNoSIS+ é um meta-alinhador de ontologias criado para ser utilizado em alinhamentos em tempo de execução [Souza 2012]. Assim, foi definida uma abordagem para calibra-

gem de pesos que aproxima a solução com um custo menor que os existentes. Para solucionar o problema de calibragem, o GNoSIS+ implementa um algoritmo genético onde os indivíduos são representados como sequências de valores reais. Cada valor real representa um peso associado a um alinhador. Diferente de outros sistemas de meta-alinhamento, o GNoSIS+ não precisa conhecer os alinhamentos de referência, mas somente um pequeno conjunto de correspondências fornecidas pelo engenheiro de ontologias. Desta forma, a abordagem se torna mais aplicável para cenários em que não são conhecidos os alinhamentos, mas onde um engenheiro de ontologias possa apontar correspondências entre algumas entidades sem muito esforço. Por utilizar um conjunto pequeno para treinamento, a função objetivo consegue avaliar as soluções intermediárias mais rapidamente.

Ainda, o GNoSIS+ permite criar alinhadores mais complexos através da definição de composição de alinhadores atômicos, isto é, alinhadores pré-implementados. A figura 1 mostra uma composição de alinhadores definida no GNoSIS+.

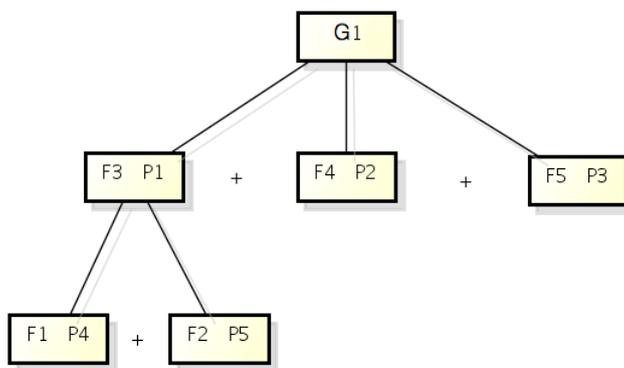


Figura 1. Definição do alinhador $G_1 = (F_1 P_4 + F_2 P_5) P_1 + F_4 P_2 + F_5 P_3$, onde F_1, F_2, F_4 e F_5 são alinhadores atômicos e P_i são pesos.

O percurso em pós-ordem na árvore de composição apresentada na figura 1 gera uma sequência $\Xi = \{F_1, F_2, F_3, F_4, F_5\}$. Para encontrar uma solução que representa a melhor calibragem para essas funções, são criados cromossomos contendo 5 genes ($|\Xi| = 5$) na forma $C = \{g_1, g_2, g_3, g_4, g_5\}$, onde g_i é um número real pertencente a $[0, 1]$ que representa o peso a ser aplicado à função $F_i, 1 \leq i \leq 5$. Uma sequência $\Xi^{e,e'}$ representa a sequência de valores v_1, v_2, \dots, v_n de similaridades entre os elementos e e e' de ontologias distintas, onde v_i é igual a $F_i(e, e')$ e $F_i \in \Xi$. Considerando uma correspondência conhecida entre dois elementos e e e' quaisquer e um cromossomo C , é possível utilizar uma função $\alpha : \Xi^{e,e'} \times C \mapsto \mathfrak{R}$ que retorna o valor da similaridade entre os elementos e e e' após a aplicação de cada peso $g_i \in C$ nos valores de similaridade $v_i \in \Xi^{e,e'}$.

O GNoSIS+ parte do princípio de que algumas correspondências podem ser facilmente identificadas por engenheiros de ontologias sem muito esforço, mesmo em ontologias muito grandes. Assim, seja CP as correspondências fornecidas por engenheiros de ontologias, o GNoSIS+ trata o problema como uma minimização da inequação (9), onde s é o grau de similaridade definido pelo engenheiro de ontologias para a correspondência entre e e e' .

$$\sum_{i=1}^{|CP|} |s_i - \alpha(\Xi^{e_i, e'_i}, C)| \geq 0 \quad (9)$$

Uma vez que não são utilizadas medidas como precisão e cobertura para avaliar a qualidade da solução, o sistema é capaz de computar de forma mais eficiente o valor da função objetivo. Por outro lado, a abordagem é dependente do conjunto de correspondências conhecidas passadas como entrada. O conjunto deve conter correspondências que ilustrem os tipos de heterogeneidades presentes nas ontologias para que os pesos calculados pelo meta-alinhador corresponda aos tipos de diferenças (sintática, semântica, estrutural) entre as duas ontologias.

3.6. MapPSO

Uma solução para o problema utilizando uma técnica de inteligência de partículas é adotada pelo MapPSO. A solução adotada utiliza a abordagem de enxame de partículas. A otimização por enxame de partículas simula o comportamento social de organismos naturais, como a reunião de pássaros ou peixes para encontrar um local com alimento suficiente [Shi and Eberhart 1998]. Nesta abordagem, os autores tem como objetivo identificar o melhor alinhamento que contenha o máximo de correspondências 1:1 possíveis. Para tal, os autores utilizam um alinhador para avaliar a similaridade entre conceitos de ontologias, sendo a abordagem independente de alinhador. Como restrição, o alinhador deve implementar uma medida de distância, ou seja, deve retornar um valor numérico entre 0 e 1, onde valores próximos de 0 correspondem a uma grande similaridade entre os conceitos avaliados e valores próximos de 1 corresponde a uma baixa similaridade. Uma vez que a abordagem utiliza somente uma função de similaridade, ela se difere das abordagens anteriormente apresentadas por não ser aplicável para calibrar ou selecionar alinhadores para pares de ontologias. Mas, pela definição 2.9, pode ser considerada uma abordagem de meta-alinhamento por realizar a busca de um alinhamento ótimo utilizando alinhadores pré-definidos.

Como a abordagem se baseia em medidas de distância para avaliar correspondências, valores baixos de avaliação denotam melhores correspondências. Ainda, como os autores objetivam maximizar o tamanho do alinhamento, ou seja, o número de correspondências dentro de um alinhamento, o objetivo da abordagem é identificar

$$A^* = \operatorname{argmin} F(A) \quad (10)$$

onde $F(A)$ é uma função que mede a qualidade do alinhamento A de acordo com as suas correspondências e, ao mesmo tempo,

$$A^* = \operatorname{argmax} |A| \quad (11)$$

de todos os possíveis alinhamentos candidatos A . Assim, o número de correspondências de um alinhamento não pode aumentar sem diminuir a qualidade total do alinhamento e vice-versa. Estes dois objetivos são combinados na função objetivo abaixo, sendo C_i o conjunto de classes da ontologia o_i :

$$A^* = \frac{0.4 \times (\min(|C_1|, |C_2|) - |A|) + 1.6 \times F(A)}{2} \quad (12)$$

Nesta abordagem, cada solução é representada como um alinhamento candidato. O tamanho n (dimensão) da solução é, no máximo, igual ao número máximo de correspondências 1:1, ou seja, $n \leq \min(|C_1|, |C_2|)$. As soluções podem variar em relação à sua dimensionalidade, mas cada solução possui um tamanho fixo. Uma solução (partícula) é definida como um vetor

$$X_i = (c_{(i,1)}, c_{(i,2)}, \dots, c_{(i,k)}) \quad (13)$$

onde $i \in \{1, \dots, N\}$, $c_{(i,j)}$ é uma correspondência, N é o tamanho da população e $k \leq n$.

Uma vez que o MapPSO força a busca por alinhamentos com muitas correspondências, a abordagem pode encontrar dificuldades ao ser aplicada em cenários nos quais o alinhamento ótimo possui poucas correspondências em relação ao tamanho das ontologias ou quando o engenheiro de ontologias deseja encontrar somente alinhamentos com correspondências com uma confiança muito alta definidas por um *threshold*. Ainda, o tamanho da representação da solução pode dificultar a adoção da abordagem para cenários de alinhamentos de ontologias com muitos conceitos.

3.7. MapEVO

O mapEVO utiliza a técnica de programação evolutiva, a qual simula espécies que competem entre si no espaço do problema. Ao contrário de algoritmos genéticos, a atualização da população não envolve recombinação genética e são utilizados somente operadores de mutação [Whitley 2001].

O MapEVO foi implementado como uma variação do MapPSO e, com isso, possuem muitas semelhanças. No MapPSO, uma solução (indivíduo) é um enxame de partícula, enquanto no MapEVO os indivíduos representam espécies em evolução. No MapEVO alguns indivíduos podem ser extintos e outros podem se auto-reproduzir. Nesta auto-reprodução, os filhos sofrem mutações aleatórias. Na abordagem baseada em enxame de partículas, a população é constante através das gerações, mas as posições no espaço de busca podem mudar de acordo com a memória da partícula e a comunicação entre elas. A função objetivo é a mesma do MapPSO, a qual é composta de uma avaliação das correspondências no alinhamento candidato e uma avaliação do alinhamento como um todo.

O MapEVO foi desenvolvido com o objetivo de provar que a definição da função objetivo e a representação do indivíduo é mais importante do que a meta-heurística baseada em população implementada. Como as abordagens são muito similares, os autores mostram em [Bock et al. 2011] que as ferramentas obtiveram resultados muito próximos por conta do uso da mesma função objetivo e representação do indivíduo.

4. Análise comparativa das abordagens baseadas em população

A avaliação de abordagens para alinhamento de ontologias geralmente são analisadas através de métricas como *f-measure*, precisão, entre outras. Para facilitar a análise, alguns

benchmarks estão disponíveis. Entre eles, o mais reconhecido são os *datasets* da OAEI (Ontology Alignment Evaluation Initiative) [Euzenat et al. 2011]. Contudo, embora seja importante que alinhadores possuam alta precisão ou cobertura, outros aspectos também são importantes, principalmente para meta-alinhadores. Para esse tipo de sistema, é importante verificar a necessidade de treinamento do sistema, sua aplicabilidade para alinhamentos em larga escala ou uso em aplicações que necessitam realizar alinhamento em tempo de execução, entre outras características.

Neste trabalho, optou-se por comparar os meta-alinhadores baseando-se nos critérios propostos em [Martinez-Gil and Aldana-Montes 2012]. Dada a dificuldade em comparar esse tipo de ferramenta, o autor propõe uma categorização *fuzzy* com uma escala de 1 a 5. É possível comparar as ferramentas a partir dos resultados apresentados na literatura e nas características apontadas pelos autores. São cinco os critérios a serem comparados:

- **Resultados:** Este critério diz respeito à média harmônica dos resultados encontrados no *benchmark* da OAEI. Vale ressaltar que algumas discrepâncias podem ocorrer ao comparar de forma bruta os resultados apresentados por cada pesquisador. Por exemplo, alguns trabalhos apresentam os resultados alcançados por um subconjunto do *benchmark* somente, enquanto outros trabalhos apresentam o resultado de todos os *datasets*. Ainda, alguns trabalhos não apresentam o resultado discriminado por *dataset*, mas somente um resultado consolidado. Contudo, objetivo aqui não é realizar um *ranking* de ferramentas, mas entender o escopo de cada abordagem. Assim, embora algumas discrepâncias possam ocorrer, decidimos adotar o resultado médio mais alto de cada ferramenta.
- **Desempenho:** Este critério está relacionado ao tempo necessário para obter os resultados. Como alguns pesquisadores não disponibilizam as ferramentas para testes de terceiros, não é intenção deste trabalho medir o tempo de cada ferramenta. Contudo, é de conhecimento que o tempo necessário para chegar a um resultado depende da estratégia utilizada (força bruta, busca local, etc), além da forma de representação e avaliação da solução. Desta forma, as abordagens são classificadas com base nesta análise. Por outro lado, alguns trabalhos submetidos à avaliação da OAEI após 2011 tem disponibilizado informações quanto ao seu desempenho.
- **Treinamento:** Uma das desvantagens das ferramentas de meta-alinhamento de ontologias é a quantidade de esforço necessário para treinar a ferramenta. Embora essa seja uma característica inerente de ferramentas de aprendizado de máquina, também é considerado o tempo necessário para otimizar as ferramentas heurísticas que necessitam de um pré-processamento ou de um esforço manual para que o engenheiro de ontologias possa fornecer dados prévios.
- **Comportamento do usuário:** A importância do especialista que opera meta-alinhadores está na capacidade de ajustar corretamente a relevância de cada alinhador para alcançar um melhor resultado. Este critério mede se é possível simular o comportamento do especialista que treina a ferramenta. Ou seja, se a ferramenta é capaz de fornecer soluções intermediárias e de otimizar essas soluções para encontrar a relevância desejada para cada alinhador. Meta-alinhadores que fornecem resultados numéricos dos pesos de cada alinhador são melhor classificados neste critério.

- Classificação (*Ranking generation*): É possível construir *rankings* para alinhadores utilizando técnicas de meta-alinhamento. Isto é possível pois muitas técnicas de meta-alinhamento são capazes de identificar uma aptidão numérica para cada combinação de alinhadores. Assim, pode-se gerar um *ranking* de abordagens alternativas.

A figura 2 apresenta o resultado da comparação dos sete meta-alinhadores.

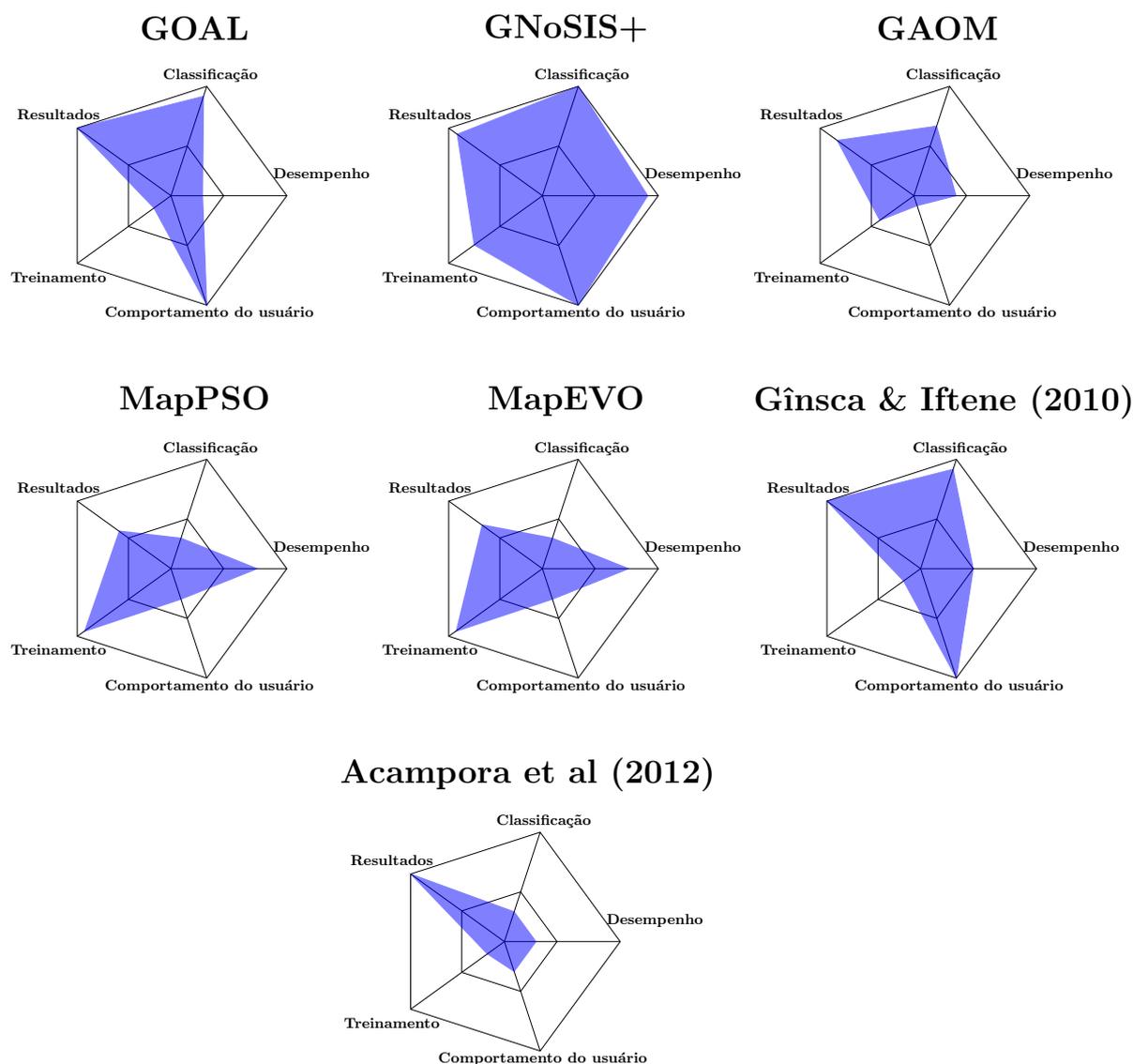


Figura 2. Gráfico de teia para comparação entre os meta-alinhadores baseados em populações

A modelagem do problema de meta-alinhamento definida nos meta-alinhadores GAOM, MapPSO, MapEVO e na abordagem de [Acampora et al. 2012] não resultam um conjunto de pesos. Por esse motivo, estas abordagens receberam um valor menor para o critério classificação. Já as demais abordagens retornam um conjunto de pesos, sendo possível identificar a relevância de cada alinhador no alinhamento final. O GNoSIS+, diferente dessas abordagens, acrescenta a possibilidade de permitir, na entrada do problema,

que composições de funções sejam definidas, o que facilita na identificação da relevância dos alinhadores.

A nota de resultados dos meta-alinhadores foi definida de acordo com os resultados alcançados no *benchmark* da OAEI. A exceção está nas abordagens de [Acampora et al. 2012] e a de [Gînsca and Iftene 2010], pois os mesmos não avaliaram suas propostas com o *benchmark* da OAEI ou não divulgaram os resultados alcançados. Para essas abordagens, foi então atribuída a nota máxima de resultados, uma vez que o objetivo do trabalho não é classificar as ferramentas, mas mostrar seus pontos fortes. Como os autores dessas abordagens não divulgaram seus resultados, consideramos mais adequado atribuir nota máxima do que nota mínima.

Em relação à simulação do comportamento do usuário, o GAOM, MapPSO, MapEVO e a abordagem de [Acampora et al. 2012] não simulam o comportamento de um usuário especialista, isto é, não é possível parar a simulação para retornar os valores que melhor configuram o sistema até aquele momento. Os demais sistemas permitem a simulação de comportamento do usuário.

O desempenho dos meta-alinhadores está relacionado com a forma com que o meta-alinhador está modelado para avaliar a qualidade de suas soluções intermediárias. Os meta-alinhadores GOAL, GAOM e a proposta de [Acampora et al. 2012] necessitam computar ou avaliar todos os alinhamentos para cada indivíduo da população em cada geração. Esse processo é extremamente custoso. A abordagem de [Gînsca and Iftene 2010], por sua vez, tenta minimizar esse problema realizando um pré-processamento de todas as medidas de similaridades possíveis entre todas as combinações de pares de entidades. Contudo, a função objetivo dessa proposta necessita avaliar, para cada indivíduo, a qualidade da sua representação através de medidas de qualidade como precisão e cobertura, a qual demanda também certo processamento, principalmente em modelos com muitas entidades. O GNoSIS+, por sua vez, possui uma função objetivo que não utiliza medidas de qualidade de alinhamento e, por isso, é rápida. Além disso, o GNoSIS+ necessita executar cada alinhador somente uma vez para cada correspondência fornecida pelo usuário especialista. Uma vez computado esses valores (e considerando que o número de correspondências fornecidas é muito menor que o número de entidades das ontologias), o GNoSIS+ se comporta como a abordagem de [Gînsca and Iftene 2010], a qual não necessita recomputar as similaridades. O MapPSO e o MapEVO representam indivíduos como sequências de correspondências e cada indivíduo é avaliado através da quantidade total de ocorrências na solução e da soma das distâncias contidas na solução, ambas fáceis de serem computadas.

Em relação à necessidade de treinamento, é considerado o treinamento necessário para preparar os dados para o início do meta-alinhamento, como a necessidade de pré-processamento, a necessidade de dados de referência para guiar o algoritmo ou a verificação dos modelos durante o processo. O meta-alinhador GOAL, bem como as abordagens de [Acampora et al. 2012] e [Gînsca and Iftene 2010] necessitam de alinhamentos de referência para guiar o comportamento do algoritmo. O comportamento do meta-alinhador GAOM é guiado por uma função objetivo que verifica uma solução e as duas ontologias de entrada para cada indivíduo da população. O MapPSO e MapEVO não necessitam de modelos de referência para guiar o comportamento do algoritmo e não necessitam de uma fase de pré-processamento. O GNoSIS+, por sua vez, necessita de um

conjunto reduzido de dados referência.

A tabela 1 apresenta um resumo das principais características das abordagens apresentadas. Vale salientar que a adoção de uma abordagem de meta-alinhamento depende da análise das características mais importantes para o cenário de aplicação da abordagem. Em alguns cenários, por exemplo, pode ser mais interessante usar uma abordagem que não necessite de treinamento, mesmo que esta não alcance resultados tão próximos ao estado da arte.

5. Conclusões

O meta-alinhamento de ontologias é um campo promissor para o problema de alinhamento de ontologias, uma vez que não existem alinhadores capazes de prover alinhamentos ótimos para todos os cenários. Este artigo trata o problema de meta-alinhamento sob o ponto de vista de um problema de otimização e foram apresentadas as principais técnicas baseadas em populações para meta-alinhamento de ontologias. Estas técnicas compartilham características importantes para o problema de meta-alinhamento, como a capacidade de serem facilmente paralelizáveis e de permitirem que a execução seja repentinamente terminada e os resultados intermediários possam ser coletados. O meta-alinhamento representa um esforço para fazer com que a tarefa de alinhamento seja um processo mais independente dos usuários, do contexto e até mesmo dos dados envolvidos.

Em [Shvaiko and Euzenat 2012], os autores analisam o campo de alinhamento de ontologias e identificam oito desafios que necessitam ser resolvidos para que esse campo de pesquisa consiga alcançar melhores resultados. Entre os desafios apresentados, está a capacidade de selecionar e compor alinhadores e de fornecer resultados em tempo de execução. Esse artigo apresentou as abordagens que tentam resolver essas questões e as comparou em relação às características definidas pela literatura.

Diversos problemas ainda estão em aberto neste campo de pesquisa. Alguns meta-alinhadores são capazes de otimizar a precisão ou cobertura dos alinhamentos, como o sistema GOAL. Contudo, pode-se facilmente obter uma alta precisão em detrimento da cobertura (e vice-versa). Alguns sistemas tentam otimizar o *f-measure*, a qual é uma média entre precisão e cobertura. Dessa forma, espera-se evitar o problema de privilegiar uma medida ou outra. Porém, esse tipo de abordagem não evita o desvio indesejado entre a precisão e a cobertura. Por exemplo, o melhor *f-measure* pode ser composto de uma alta precisão e uma baixa cobertura ou vice-versa. Segundo [Martinez-Gil and Aldana-Montes 2012], esse problema pode ser tratado com a aplicação de algoritmos multiobjetivos. Ainda, os sistemas de meta-alinhamento ainda trabalham, na sua maioria, com alinhamentos simples do tipo equivalência 1:1 ou 1:n. Em muitos casos não é possível resolver o problema com alinhamentos simples, sendo necessários alinhamentos mais complexos como reconhecimento de subclasses entre ontologias, mapeamento entre propriedades (por exemplo, "Nome completo" = "Nome" + "Sobrenome"), alinhamento de classes com instâncias, entre outros.

Referências

Acampora, G., Loia, V., Salerno, S., and Vitiello, A. (2012). A hybrid evolutionary approach for solving the ontology alignment problem. *International Journal of Intelligent Systems*, 27(3):189–216.

Tabela 1. Resumo das características principais das abordagens de meta-alinhamento de ontologias

	Abordagem	Representação do indivíduo	Função objetivo	Resultados na literatura (<i>f-measure</i>)
GNoSIS+	algoritmo genético	Vetor de numéricos onde cada posição representa o atribuído a um alinhador	Minimizar a diferença entre a similaridade com os pesos atuais e a similaridade fornecida pelas correspondências de entrada	0,96
GOAL	algoritmo genético	Vetor de numéricos onde cada posição representa o atribuído a um alinhador	Maximizar precisão e recall	0,97
GAOM	algoritmo genético	Vetor de numéricos onde o valor j da posição i representa uma correspondência do conceito e_i para o conceito e_j	Maximizar o número de correspondências	0,90
MapEVO	programação evolutiva	Um conjunto de correspondências. Nesta abordagem, cada indivíduo pode ter tamanho distinto.	Maximizar o número de correspondências	0,66
MapPSO	enxame de partículas	Um conjunto de correspondências. Nesta abordagem, cada indivíduo pode ter tamanho distinto.	Maximizar o número de correspondências	0,64
Acompora	algoritmo memético	Vetor de numéricos onde o valor j da posição i representa uma correspondência do conceito e_i para o conceito e_j	Minimizar a soma das distâncias de cada correspondência	N/D
Gînska	algoritmo genético	Um vetor de numéricos onde cada posição representa um valor de configuração do alinhador (como <i>threshold</i>) e uma sequência de bits onde cada bit representa uma correspondência	Maximizar precisão e recall	N/D

- Beheshti, Z. and Shamsuddin, S. M. H. (2013). A review of population-based meta-heuristic algorithm. *International Journal of Advances in Software Computing & Its Applications*, 5(1).
- Bock, J., Danschel, C., and Stumpp, M. (2011). Mappso and mapevo results for oaei 2011. In *Proceedings of 6th ISWC workshop on ontology matching*, pages 179–183.
- Bock, J. and Hettenhausen, J. (2012). Discrete particle swarm optimisation for ontology alignment. *Information Sciences*, 192:152–173.
- Bock, J., Rudolph, S., and Mutter, M. (2012). More than the sum of its parts – holistic ontology alignment by population-based optimisation. *Foundations of Information and Knowledge Systems*, 7153:71–90.
- Bouquet, P., Euzenat, J., Franconi, E., Serafini, L., Stamou, G., and TessarisDe, S. (2004). D2.2.1 specification of a common framework for characterizing alignment. Technical Report DIT-04-090, Department of Information and Communication Technology, University of Trento, Trento, Italy.
- Buranarach, M. (2001). *The foundation for semantic interoperability on the World Wide Web*. PhD thesis, Department of Information Science and Telecommunications, School of Information Sciences, University of Pittsburgh.
- Do, H. H. (2006). *Schema Matching and Mapping-based Data Integration: Architecture, Approaches and Evaluation*. VDM Verlag, Saarbrücken, Germany, 1 edition.
- Duchateau, F., Bellahsene, Z., and Coletta, R. (2008). A flexible approach for planning schema matching algorithms. In *Proceedings of the On The Move Conferences*, volume 1, pages 249–264. Springer-Verlag.
- Duchateau, F., Coletta, R., Bellahsene, Z., and Miller, R. (2009). (not) yet another matcher. In *Proceedings of the ACM Conference on Information and Knowledge Management CIKM'09*, pages 1537–1540. ACM Press.
- Ehrig, M., Staab, S., and Sure, Y. (2005). Bootstrapping ontology alignment methods with apfel. In *Proceedings of the International Semantic Web Conference ISWC'05*, pages 186–200. Springer-Verlag.
- Euzenat, J., Meilicke, C., Stuckenschmidt, H., Shvaiko, P., and Trojahn, C. (2011). Ontology alignment evaluation initiative: Six years of experience. *Journal on data semantics XV*, 6720:158–192.
- Euzenat, J. and Shvaiko, P. (2007). *Ontology matching*. Springer-Verlag New York Inc, 1 edition.
- Fensel, D., Facca, F. M., Simperl, E., and Toma, I. (2011). *Semantic Web Services*. Springer Verlag, New York, 1 edition.
- Gînsca, A.-L. and Iftene, A. (2010). Using a genetic algorithm for optimizing the similarity aggregation step in the process of ontology alignment. In *Proceedings of the 9th Roedunet International Conference (RoEduNet)*, pages 118–122, Sibiu, Romania. IEEE.
- Giunchiglia, F., Yatskevich, M., and Shvaiko, P. (2007). Semantic matching: Algorithms and implementation. *Journal on Data Semantics IX*, pages 1–38.

- Gracia, J. and Mena, E. (2008). Web-based measure of semantic relatedness. In *Proceedings of the Web Information Systems Engineering-WISE 2008*, pages 136–150. Springer-Verlag.
- Haupt, R., Haupt, S., and Wiley, J. (2004). *Practical genetic algorithms*. Wiley-Interscience, 2 edition.
- Huang, J., Dang, J., Vidal, J., and Huhns, M. (2007). Ontology matching using an artificial neural network to learn weights. In *Proceedings of the IJCAI Workshop on Semantic Web for Collaborative Knowledge Acquisition (SWeCKa-07)*, pages 80–85, Hyderabad, India.
- Mao, M., Peng, Y., and Spring, M. (2008). Neural network based constraint satisfaction in ontology mapping. In *Proceedings of the Conference on Artificial Intelligence AAAI '08*, pages 1207–1212. AAAI Press.
- Martinez-Gil, J. and Aldana-Montes, J. (2009). Evaluation of two heuristic approaches to solve the ontology meta-matching problem. *Knowledge and Information Systems*, 26(2):225–247.
- Martinez-Gil, J. and Aldana-Montes, J. F. (2012). An overview of current ontology meta-matching solutions. *Knowledge Engineering Review*, 27(4):393–412.
- Moscato, P. and Cotta, C. (2003). Una introducción a los algoritmos meméticos. *Revista Iberoamericana de Inteligencia Artificial*, 7(19):131–148.
- Santos, C. T., Euzenat, J., Tamma, V., and Payne, T. (2011). Argumentation for reconciling agent ontologies. In Eli, A., Kon, M., and Orgun, M., editors, *Semantic Agent Systems*, volume 1, chapter 5, pages 89—111. Springer.
- Shi, Y. and Eberhart, R. (1998). A modified particle swarm optimizer. In *Proceedings of the IEEE International Conference on Evolutionary Computation*, pages 69–73. IEEE Press.
- Shvaiko, P. and Euzenat, J. (2005). A survey of schema-based matching approaches. *Journal on Data Semantics IV*, pages 146–171.
- Shvaiko, P. and Euzenat, J. (2008). Ten challenges for ontology matching. In *Proceedings of the 7th International Conference on Ontologies, DataBases, and Applications of Semantics (ODBASE)*, pages 1164–1182, Monterrey (MX). Springer.
- Shvaiko, P. and Euzenat, J. (2012). Ontology matching: state of the art and future challenges. *IEEE Transactions on Knowledge and Data Engineering*, 1(99):1–20.
- Souza, J. F. (2012). *Uma abordagem heurística uni-objetivo para calibragem em meta-alinhamento de ontologias*. PhD thesis, Departamento de Informática, Centro Técnico Científico, Pontifícia Universidade Católica do Rio de Janeiro.
- Souza, J. F., Siqueira, S. W. M., and Melo, R. N. (2010). Evolution in ontology-based user modeling. In Lytras, M. D., de Pablos, P. O., and Damiani, E., editors, *Semantic Web Personalization and Context Awareness: Management of Personal Identities and Social Networking*. IGI Global, Hershey, Pennsylvania.
- Souza, J. F., Siqueira, S. W. M., and Melo, R. N. (2012). Providing information from brazilian politicians using linked data. In de Pablos, P. O., Lytras, M. D., Tennyson,

- R., and Gayo, J. E. L., editors, *Cases on Open-Linked Data and Semantic Web Applications*. IGI Global, Hershey, Pennsylvania. Em fase de editoração.
- Spoehr, D., Hollink, L., and Cimiano, P. (2011). A machine learning approach to multilingual and cross-lingual ontology matching. In *Proceedings of the 10th International Semantic Web Conference (ISWC)*, pages 665–680.
- Staab, S. and Stuckenschmidt, H. (2006). *Semantic Web and Peer-to-peer: Decentralized Management and Exchange of Knowledge and Information*. Springer Verlag, New York, 1 edition.
- Wang, J., Ding, Z., and Jiang, C. (2006). Gaom: Genetic algorithm based ontology matching. In *Proceedings of the IEEE Asia-Pacific Conference on Services Computing*, pages 617–620, Guangzhou, Guangdong. IEEE Press.
- Whitley, D. (2001). An overview of evolutionary algorithms: practical issues and common pitfalls. *Information and Software Technology*, 43(14):817–831.