

# Meta-alinhamento de ontologias utilizando a abordagem presa-predador

Nicolas Ferranti<sup>1</sup>, Stênio Sã Rosário Furtado Soares<sup>1</sup>, Jairo F. De Souza<sup>1</sup>

<sup>1</sup> Departamento de Ciência da Computação – Universidade Federal de Juiz de Fora (UFJF)  
36.036-900 – Juiz de Fora, MG – Brazil

{nicolas1, ssoares}@ice.ufjf.br, jairo.souza@ufjf.edu.br

**Abstract.** *Every year, several new ontology matchers are proposed in the literature, each one using a different heuristic, which implies in different performances according to the characteristics of the ontologies. An ontology meta-matcher consists of an algorithm that combines several approaches in order to obtain better results in different scenarios. To achieve this goal, it is necessary to define a criterion for the use of matchers. We presented in this work an ontology meta-matcher that combines several ontology matchers making use of the evolutionary meta-heuristic prey-predator as a means of parameterization of the same.*

**Resumo.** *Todo ano, diversos novos alinhadores de ontologias são propostos na literatura, cada um utilizando uma heurística diferente, o que implica em desempenhos distintos de acordo com as características das ontologias. Um meta-alinhador consiste de um algoritmo que combina diversas abordagens a fim de obter melhores resultados em diferentes cenários. Para atingir esse objetivo, é necessária a definição de um critério para melhor uso de alinhadores. Neste trabalho, é apresentado um meta-alinhador de ontologias que combina vários alinhadores através da meta-heurística evolutiva presa-predador como meio de parametrização das mesmas.*

## 1. Introdução

Ontologias são construídas por pessoas com diversos níveis de especialização e visão de domínio. Logo, conceitos que podem descrever o mesmo tipo de objeto podem se encontrar representados de formas distintas, tanto na sintaxe dos termos quanto na estrutura de relações, gerando um problema de heterogeneidade na semântica dos dados. Para solucionar problemas de heterogeneidade, é preciso uma forma de especificar, sem ambiguidade, os vocabulários subjacentes aos sistemas de informação [Farinelli and Almeida 2014].

O alinhamento de ontologias é uma etapa fundamental em aplicações tradicionais da área de banco de dados que por natureza lidam com estruturas heterogêneas, em alguns sistemas o processo é considerado uma etapa prévia fundamental para o uso do sistema como o processo de *message mapping* [Hai et al. 2007], em outros, como os sistemas Web que fazem uso de bases de Dados Ligados, a operação de alinhamento é necessária em tempo de execução [Souza et al. 2014].

O problema a ser resolvido consiste de definir relações entre conceitos das ontologias envolvidas, compatibilizando as estruturas de forma a representar a união dos conjuntos de dados em um novo modelo. O alinhamento de ontologias, como é denominado, é

um problema complexo e suas características possibilitam que seja abordado por diversas técnicas computacionais. Devido à alta heterogeneidade das ontologias, não existe uma técnica que se sobressaia dentre as outras em todos os aspectos [Xue and Tang 2017]. Logo, abordagens de meta-alinhamento podem ser utilizadas neste cenário. Um meta-alinhador combina diversas técnicas de alinhamento, a fim de explorar vários aspectos da heterogeneidade para evitar que o desempenho do alinhamento seja restrito a alguma característica das ontologias. A literatura sugere que os melhores resultados encontrados no meta-alinhamento de ontologias estão associados ao uso de algoritmos evolucionários devido à sua capacidade de adaptação e, conseqüentemente, adequação do uso de cada técnica [Souza 2012, Shi and Eberhart 1998, Xue and Tang 2017].

O uso de meta-heurísticas em meta-alinhadores é justificada pelo tamanho do espaço de busca do problema, que desencoraja o uso de abordagens exaustivas devido ao tempo de processamento demandado. Além disso, a literatura mostra que abordagens populacionais apresentam-se como boas alternativas na solução do problema. Embora o uso de Algoritmos Genéticos seja mais frequente entre as abordagens, outras abordagens bio-inspiradas constituem um campo promissor a ser exploradas na solução do problema. Na última década, muitas meta-heurísticas foram propostas e ainda são pouco exploradas na área de banco de dados [Sorensen et al. 2017].

Dentre as abordagens populacionais bio-inspiradas, o algoritmo presa-predador (PPA) possui características que são adequadas para o problema de meta-alinhamento, uma vez que permite que regiões promissoras do espaço de soluções sejam exploradas por um conjunto de agentes (soluções), pressionados a fugirem de regiões pouco atraentes em termos de valor da função objetivo, ao mesmo tempo que permite a exploração de novas regiões ao se atribuir a esses agentes um comportamento pseudoaleatório na definição do seu deslocamento.

O objetivo deste trabalho é tratar o problema de meta-alinhamento de ontologias através de uma adaptação do algoritmo presa-predador (PPA) e mostrar sua aplicabilidade para esse problema. Para analisar o comportamento da solução, foi utilizado o *benchmark* fornecido pela OAEI (*Ontology Alignment Evaluation Initiative*). Observou-se que o PPA é eficiente e eficaz ao parametrizar as técnicas de alinhamento de forma a obter uma solução que é próxima da ótima em tempo polinomial. Assim, o trabalho mostra que o PPA pode ser melhor explorado no cenário de alinhamento de ontologias.

## 2. Técnicas e Meta-alinhamento

O problema de alinhamento de ontologias pode ser tratado de diversas formas. Esta seção traz uma revisão dos métodos de alinhamento e meta-alinhamento presentes na literatura. Segundo [Otero-Cerdeira et al. 2015], técnicas de alinhamento de ontologias podem ser classificadas em níveis, seguindo duas lógicas de interpretação: como as técnicas lidam com a entrada fornecida, englobando alinhadores em nível dos elementos e em nível estrutural e a interpretação baseada no tipo da entrada fornecida, contemplando técnicas baseadas em conteúdo e em contexto. A especificação de cada tipo é apresentada a seguir:

- Alinhadores em nível dos elementos: técnicas que obtêm as correspondências considerando as entidades nas ontologias isoladamente, ignorando que são partes da estrutura da ontologia.

- Alinhadores a nível estrutural: técnicas que obtêm as correspondências analisando como as entidades se encaixam dentro da estrutura das ontologias.
- Baseadas em conteúdo: técnicas com foco na informação interna proveniente das ontologias que serão alinhadas.
- Baseadas em contexto: consideram para a correspondência as informações externas que podem surgir de relações entre ontologias ou outros recursos externos (contexto).

Com base no sistema de classificação presente em [Otero-Cerdeira et al. 2015], é possível agrupar as técnicas presentes na literatura por categoria, tornando mais fácil a comparação entre as mesmas. Em [Akbari et al. 2009], os autores apresentam uma medida de similaridade baseada na distância de Levenshtein para a comparação de *strings* aplicada em alinhamento de ontologias. Técnicas de comparação de *strings* fazem uso, por exemplo, do rótulo e da descrição das entidades da ontologia para determinar relações. Existem várias métricas para cálculo de distância que podem ser usadas nesses métodos, como Jaccard, n-gram, Levenshtein, TFIDF, euclidiana etc. Em [Joslyn et al. 2009] foram aplicadas técnicas da Teoria dos Grafos ao problema. Essas técnicas consideram as ontologias a serem alinhadas como grafos rotulados, ou até mesmo árvores, caindo no problema de isomorfismo entre dois grafos. Observa-se, entretanto, que não existe uma técnica que se sobressaia às demais de forma genérica e seja eficaz em todos os casos [Souza 2012]. Em alguns casos, a comparação de strings identifica facilmente a equivalência entre nomes semelhantes, enquanto a análise do grafo pode indicar que, devido às relações encontradas, duas classes das ontologias são equivalentes. Logo, é interessante estudar o uso dessas técnicas em conjunto para melhorar a acurácia de um alinhamento.

Com o uso de técnicas de alinhamento híbridas, o Lily [Wang and Wang 2016] é capaz de resolver alguns problemas relacionados a ontologias heterogêneas. Os resultados alcançados pela abordagem no *benchmark* da OAEI na campanha de 2016 foram superiores ou iguais a todos os outros alinhadores em relação à Medida-F [Achichi et al. 2016]. O Lily constrói um subgrafo semântico na tentativa de eliminar a interpretação heterogênea dos elementos das ontologias. Todo o cálculo de similaridade é realizado sobre esse subgrafo. A similaridade é computada por meio de técnicas de similaridade entre strings e similaridade estrutural. Ao final, as similaridades computadas são combinadas utilizando pesos experimentalmente definidos. Os pesos são fundamentais para definir o nível de confiança para uma dada abordagem e, nesse caso, são atribuídos estaticamente, prejudicando o desempenho da solução em ontologias cuja experimentação não foi aplicada. Uma alternativa viável é o desenvolvimento de meios para que o algoritmo possa se adaptar dinamicamente a uma dada entrada e atribuir automaticamente os pesos mais adequados para aquela ontologia. Abordagens que calibram o peso de técnicas de alinhamento em tempo de execução são denominadas meta-alinhadoras de ontologias.

O termo meta-alinhamento de ontologias [Euzenat et al. 2007] descreve sistemas que parametrizam automaticamente um conjunto de funções de alinhamento de ontologias. [Martinez-Gil and Aldana-Montes 2012] define um conjunto de características comuns no que tange aos meta-alinhadores de ontologias: (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 aplicadas em tempo de execução uma vez que o processo executado por elas é determinista e as relações não mudam de uma execução para a outra; (2) O processo de meta-alinhamento deve ser automático, logo,

deve ser possível que seja implementado por alguma ferramenta de alinhamento; (3) O processo deve se comportar como um especialista, caso a melhor função de alinhamento não seja conhecida, o processo deve ser capaz de experimentar pesos e combinações a fim de retornar a função mais próxima possível da melhor função de alinhamento; e (4) Uma estratégia de meta-alinhamento é avaliada com a função de alinhamento retornada.

O meta-alinhamento lida com a integração de alinhadores heterogêneos, visando encontrar os melhores parâmetros que possam afetar os resultados do alinhamento. O problema é modelado como um problema de otimização e as abordagens mais importantes empregam heurísticas em conjunto com algoritmos evolutivos, gulosos ou baseados em conjuntos de regras [Souza et al. 2014]. Como a literatura não apresenta o uso da meta-heurística presa-predador para a calibração de funções de alinhamento, e algoritmos evolucionários são frequentemente empregados para esse fim, este trabalho experimentou o uso do presa-predador para a calibragem das funções de alinhamento.

### 3. Trabalhos Relacionados

O problema de meta-alinhamento de ontologias é um problema relativamente recente e que ainda possui várias características a serem exploradas, ainda que as propostas conhecidas tenham apresentado bons resultados. Nesta seção são apresentados trabalhos relacionados que fazem uso de meta-heurísticas para tratar de alinhamento de ontologias, destacando suas principais características e contribuições.

O uso de meta-heurística tem sido explorado para resolver o problema de meta-alinhamento, como em [Souza 2012, Xue and Tang 2017, Bock and Hettenhausen 2012]. No GNoSIS+ [Souza 2012], é utilizado algoritmo genéticos para parametrizar um conjunto preestabelecido de alinhadores. O aprendizado do algoritmo é baseado em um grupo de alinhamentos de referência definidos na entrada por um engenheiro de ontologias. A premissa é que alguns relacionamentos podem ser facilmente apontados, então o AG calibra as funções do sistema baseado na referência a fim de prepará-lo para uma situação real de aplicação. É interessante destacar a representação do problema pelo GNoSIS+. Considerando  $\Xi = \{F_1, F_2, \dots, F_n\}$  um conjunto de funções de alinhamento, cada cromossomo possui  $n$  genes ( $|\Xi| = n$ ) e cada gene representa um valor real  $w \in [0, 1]$  que representa o peso a ser aplicado sobre cada função. O objetivo é minimizar a diferença entre o valor encontrado e o valor definido pelo engenheiro de ontologias para um relacionamento em específico. [Xue and Tang 2017] também emprega um algoritmo evolucionário com a mesma representação de indivíduo, entretanto, a função objetivo passa a ser maximizar o valor da média harmônica da Medida-F. A Medida-F é uma medida que leva em conta as taxas de precisão e cobertura entre os mapeamentos obtidos pelo algoritmo com os que eram esperados. A função objetivo de cada trabalho guia os respectivos algoritmos para caminhos diferentes. Para a abordagem de [Xue and Tang 2017], é necessário avaliar cada item do resultado obtido a cada iteração com a base de referência, acarretando em um custo computacional maior do que apenas comparar o resultado obtido com o valor de confiança definido pelo engenheiro, como é feito em [Souza 2012].

O MapPSO [Bock and Hettenhausen 2012] é uma solução que emprega a técnica de enxame de partículas para lidar com o problema de meta-alinhamento. O enxame de partículas é uma técnica com inspiração natural baseada no comportamento social de indivíduos, como, por exemplo, a revoada de pássaros para encontrar um local com ali-

mento suficiente [Shi and Eberhart 1998]. A abordagem busca apenas relações do tipo equivalência (1:1) e utiliza um alinhador predefinido que implementa uma função de distância. A função de distância define um nível de similaridade para um dado par de conceitos. É importante notar que o MapPSO não calibra um conjunto de funções alinhadoras, pois utiliza apenas uma. Entretanto pode ser considerado uma abordagem de meta-alinhamento por buscar um alinhamento ótimo fazendo uso de alinhadores predefinidos. A representação do indivíduo difere dos trabalhos anteriores. Nesta abordagem, cada solução é representada como um alinhamento candidato. Suponha que  $\vec{X}_p$  represente um alinhamento de duas ontologias constituído de  $k = 5$  correspondências ( $c$ ). A partícula é representada por  $\vec{X}_p = \{c_{(p,1)}, c_{(p,2)}, c_{(p,3)}, c_{(p,4)}, c_{(p,5)}\}$  onde cada  $c_{(p,i)}$  indica um valor confiança para o relacionamento  $(p, i)$ . A função objetivo do MapPSO busca encontrar a maior quantidade de alinhamentos possíveis, podendo prejudicar o desempenho em ontologias onde, por natureza, a taxa de correspondências é baixa.

A literatura mostra que abordagens evolutivas apresentam bons resultados quando aplicadas no alinhamento de ontologias, como é o caso do GOAL [Martinez-Gil and Aldana-Montes 2011] que obteve 97% de medida-f sobre o *benchmark* da OAEI, o que permite fomentar que o uso da meta-heurística presa-predador tem potencial para construir resultados efetivos para o problema. A definição da representação do indivíduo impacta na forma como o esforço da abordagem pode ser reproduzido. Uma vez que a representação seja baseada no conjunto de pesos, os parâmetros encontrados podem ser armazenados e recuperados sem muito esforço, enquanto que a representação baseada no conjunto de alinhamentos candidatos requer que todo o processo seja executado novamente. Logo, os pesos das abordagens de conjuntos de pesos tem contribuição melhor para a construção de um meta-alinhador mais genérico.

#### 4. Solução Proposta

Para tratar o problema, é aplicada a meta-heurística presa-predador baseada na interação entre animais. A primeira versão da meta-heurística foi introduzida inicialmente por [Laumanns et al. 1998], onde as presas não se movem naturalmente, estando sujeitas ao movimento dos predadores para que então possam responder de forma a se adaptar no espaço de busca, melhorando a qualidade da solução. [Tilahun and Ong 2015] apresentam uma abordagem também baseada na interação presa-predador entre animais, entretanto, o comportamento e a forma como os indivíduos interagem entre si se difere dos outros trabalhos. O trabalho de [Tilahun and Ong 2015] apresenta o comportamento da meta-heurística presa-predador que é adaptada para neste trabalho.

Considere  $S$  um conjunto de correspondências conhecidas de equivalência. O conjunto  $S$  é formado por tuplas  $(e_{1i}, e_{2i}, =, s_i)$ , onde  $e_{1i}$  e  $e_{2i}$  são entidades de ontologias distintas,  $=$  denota a relação do tipo equivalência e  $s_i$  é a similaridade conhecida, informada pelo engenheiro de ontologias, entre  $e_{1i}$  e  $e_{2i}$ . Seja  $f$  uma função de similaridade composta da soma ponderada de outras funções, ao aplicar a função  $f$  em  $e_{1i}$  e  $e_{2i}$ , espera-se encontrar o valor  $s_i$ , ou seja,  $f(e_{1i}, e_{2i}) = s_i$ . Como exemplo, considere o conjunto  $S' = (e_{11}, e_{21}, =, 1), (e_{12}, e_{22}, =, 1), (e_{13}, e_{23}, =, 1)$ , com todas as correspondências possuindo similaridade igual a 1. Considerando uma função  $\tilde{f}'(e_{11}, e_{21}) = g_1(e_{11}, e_{21})p_1 + g_2(e_{11}, e_{21})p_2 + g_3(e_{11}, e_{21})p_3$ , onde  $g_i$  representa o valor de similaridade definido pela função  $i$  que são constantes do problema e  $p_i$  representa o peso atribuído à função  $i$ . Logo, para cada alinhamento conhecido fornecido na entrada,

é possível construir um sistema linear:

$$\begin{aligned}\bar{f}'(e_{11}, e_{21}) &= s_1 \cdot g_1(e_{11}, e_{21})p_1 + g_2(e_{11}, e_{21})p_2 + g_3(e_{11}, e_{21})p_3 = 1 \\ \bar{f}'(e_{12}, e_{22}) &= s_2 \cdot g_1(e_{12}, e_{22})p_1 + g_2(e_{12}, e_{22})p_2 + g_3(e_{12}, e_{22})p_3 = 1 \\ \bar{f}'(e_{13}, e_{23}) &= s_3 \cdot g_1(e_{13}, e_{23})p_1 + g_2(e_{13}, e_{23})p_2 + g_3(e_{13}, e_{23})p_3 = 1\end{aligned}\quad (1)$$

O objetivo é encontrar os melhores valores de  $p_i$  de forma a minimizar a soma das diferenças entre o valor encontrado e o valor esperado. Como a meta-heurística presa-predador é populacional, a representação do indivíduo é baseada no conjunto de pesos  $p_i$ . Para modelar o problema, cada indivíduo recebe um conjunto de valores reais  $p_i \in [0, 1]$  cuja alteração no valor impacta diretamente na confiança atribuída às funções associadas.

Um conjunto de soluções  $p_i$  viáveis é construído de forma aleatória, onde o somatório de todos pesos de cada solução não deve ultrapassar 1. Para cada solução  $x_i$ , é atribuído um valor de sobrevivência  $SV(x_i)$ , calculado à partir da função objetivo do problema. Seja  $F(x)$  a função objetivo descrita como a soma das diferenças, ou seja, onde quanto menor o valor, melhor é a avaliação da solução, definimos:

$$SV(x_i) = 1/F(x_i) \quad (2)$$

Isso pode ser considerado como quão bem localizada está uma presa para fugir de um predador, no caso onde  $F(x_i)$  é 0 ou próximo de 0, o que seria o resultado ótimo,  $F(x_i)$  recebe uma constante da ordem de  $1^{-10}$ . Após o valor de sobrevivência (SV) de cada membro da solução ser calculado, o membro com o menor SV será designado como um predador e o resto como presas. Uma vez que as presas e o predador são definidos, as presas precisam fugir do predador e tentar seguir as melhores presas em termos de valores de sobrevivência ou encontrar um bom esconderijo ao mesmo tempo. O que leva à definição de como se dará esse movimento. Ao tratar do movimento, é preciso definir duas questões: a direção e o tamanho do passo.

#### 4.1. Cálculo da direção

Considerando que as presas precisam fugir ou tentar se esconder, é sorteado um número dentro de uma probabilidade fixa que define se a presa deve seguir as mais aptas ou procurar se esconder na vizinhança. Caso uma presa  $x_i$  escolha seguir as demais, tomando  $\{x_1, x_2, \dots, x_p\}$  como o conjunto de presas com valor de sobrevivência maior que  $x_i$ , o cálculo da nova direção é dado pela Equação 3, onde  $r_{ij}$  representa a distância entre as duas presas e  $\tau$  um valor escolhido para ponderar o peso do valor de sobrevivência. Se a probabilidade de seguir não for alcançada, uma direção aleatória  $y_r$  é construída e então avaliada na presa  $x_i$  calculando a distância para o predador.

$$y_i = \sum_{j=1}^p e^{SV(x_j)\tau - r_{ij}} (x_j - x_i) \quad (3)$$

$$d_1 = \|x_{predador} - (x_i + y_i)\| \quad (4)$$

$$d_2 = \|x_{predador} - (x_i - y_i)\| \quad (5)$$

Se  $d_1 < d_2$  então tomar a direção  $-y_r$  faz com que a presa  $x_i$  fique mais distante do predador, caso contrário é utilizada  $y_r$ . Após os cálculos de direção, fugindo ou seguindo, é necessário calcular quanto o indivíduo vai caminhar na direção encontrada.

#### 4.2. Cálculo do tamanho do passo

O tamanho do passo define o quão longe a presa vai caminhar na direção escolhida. Ressalta-se que a natureza do problema de meta-alinhamento é contínua, tornando inviável a exploração de todo o espaço de busca. Logo, a definição do passo é importante para que não se perca uma boa solução no meio do caminho. Uma vez que uma presa longe do predador não correrá tão rápido quanto uma perto, o passo é dado por:

$$\lambda_i = \frac{\lambda_{MAX}\varepsilon_1}{e^{\beta|SV(x_i)-SV(x_{predador})|^\omega}} \quad (6)$$

Onde,  $\lambda_{MAX}$  representa o maior tamanho do passo,  $\varepsilon_1$  um número escolhido aleatoriamente de forma uniforme no intervalo  $[0, 1]$  e as constantes  $\beta$  e  $\omega$  são definidas previamente antes da execução do algoritmo. Foi acrescentado um valor de granularidade ( $G$ ) na equação do passo que contribui para controle do salto, generalizando a equação de movimento das presas temos, para a direção  $y_i$  adequada à escolha de seguir ou fugir:

$$x_i \leftarrow x_i + G\lambda_i\left(\frac{y_i}{\|y_i\|}\right) \quad (7)$$

O predador sempre se movimentará na direção da presa com pior SV, com um certo nível de aleatoriedade, como descreve a Equação 3.8:

$$x_{predador} \leftarrow x_{predador} + \lambda_{MAX}(\varepsilon_5)\left(\frac{y_r}{\|y_r\|}\right) + \lambda_{MIN}(\varepsilon_6)\left(\frac{x'_i - x_{predador}}{\|x'_i - x_{predador}\|}\right) \quad (8)$$

Onde  $\lambda_{MIN}$  e  $\lambda_{MAX}$  são constantes definidas previamente representando o passo mínimo e máximo respectivamente,  $\varepsilon_5$  e  $\varepsilon_6$  são valores reais aleatórios no intervalo  $[0,1]$ ,  $y_r$  uma direção gerada aleatoriamente e  $x'_i$  representa a posição da pior presa.

#### 4.3. Intensificação da solução

Caso a presa em avaliação seja a de melhor SV em toda a população, não ocorre caminhar. Segundo [Tilahun and Ong 2015], é aconselhável que nesta presa seja executado um processo de intensificação da solução a cada iteração. Na nossa solução, é utilizada uma busca local que percorre todos os pesos de uma solução criando duas novas soluções para cada peso visitado. O processo se dá pela soma e subtração do valor de granularidade  $G$  em cada peso do indivíduo, ou seja, se  $x_i$  é um indivíduo com conjunto de pesos  $(g_1, \dots, g_n)$ , ao iterar sobre o primeiro peso são criadas duas novas soluções  $(g_1 + G, \dots, g_n)$  e  $(g_1 - G, \dots, g_n)$ . O processo é executado para cada peso e, ao final, o melhor aprimorado é escolhido para substituir o antigo indivíduo se sua aptidão for superior. Com a definição das funções de movimento e de intensificação, os passos do algoritmo podem ser especificados como:

1. Definir os parâmetros e gerar um conjunto de soluções viáveis

2. Calcular o valor de sobrevivência para cada presa e definir a melhor presa, o predador e as presas restantes
3. Fazer com que as presas e o predador se movimentem
4. Se o critério de parada for atendido, terminar a execução, senão, voltar ao passo 2

No intuito de diversificar a população criada pelo algoritmo, foi definido um número  $\kappa = 10$  que representa a quantidade de vezes que o processo deve se repetir, executando todos os passos desde a criação da população até a busca local ao final, caso seja determinado.

#### 4.4. Integração do calibrador

O trabalho desenvolvido faz uso da ferramenta de alinhamento apresentada em [Souza 2012]. A ferramenta possui arquitetura distribuída, incorpora um conjunto de funções de similaridade sintática que podem ser selecionadas para uso no meta-alinhamento e permite que o módulo calibrador de parâmetros seja substituído por outras abordagens, desde que o padrão das mensagens seja mantido e respeitando o fluxo de execução. O fluxo de execução se inicia com os arquivos de entrada. Considerando que a abordagem é supervisionada, cada teste precisa informar o conjunto de pré-alinhamentos de treinamento, juntamente com o par de ontologias, o alinhamento de referência completo para avaliação final e por último, o conjunto de funções que serão utilizadas para treinar o algoritmo. As funções de alinhamento são pré-cadastradas no sistema e fazem uso de métricas distintas para avaliar aspectos diversificados das entidades da ontologia. Algumas das métricas das funções presentes no conjunto teste são apresentadas a seguir:

- Similaridade das entidades com base na similaridade entre os comentários que descrevem as entidades
- Similaridade baseado na semelhança entre os termos da entidade
- Similaridade com base nos identificadores em comum que compõem um subgrafo das relações de cada entidade
- Similaridade das entidades com base nas instâncias de mesmo identificador
- Similaridade das entidades com base na semelhança entre os termos que identificam propriedades de tipo de dado e de objeto

Em seguida, o sistema linear é criado e as estruturas que avaliarão os pesos encontrados. É chamado o calibrador com a abordagem presa-predador e, assim, o meta-alinhador computa os alinhamentos candidatos utilizando os pesos encontrados e retornando as equivalências mais relevantes. Para escolher quais alinhamentos são mais relevantes, foi adotado um método que computa as funções e os pesos para cada par de entidades candidatas. O método ordena pelos maiores graus de similaridade encontrados e seleciona sempre o maior par como relacionamento escolhido, removendo os escolhidos do restante da lista. A Figura 1 representa a sequência dos dados dentro do sistema.

#### 5. Avaliação

Para avaliação da abordagem, foi utilizado o *benchmark* da OAEI, a qual é utilizada para avaliar abordagens de alinhamento, mas que pode também ser utilizado por meta-alinhadores. Das bases disponíveis no benchmark, foi escolhida a base dentro do domínio de referências bibliográficas. Nesta base, a ontologia de referência é descrita sobre a linguagem OWL-DL e serializada em RDF/XML. Esta base é adequada para analisar meta-alinhadores porque possui um conjunto de testes que representam alterações sistemáticas

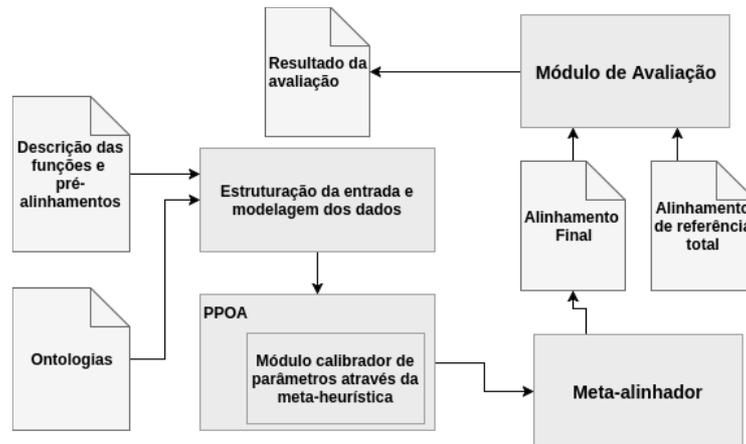


Figura 1. Fluxograma de execução do algoritmo

na ontologia de referência, de forma a analisar qualidades e defeitos de alinhadores de ontologias. No contexto de meta-alinhamento, os testes podem indicar o quão bem uma abordagem de meta-alinhamento se adapta para diferentes cenários. Ao todo, são 56 casos de testes, divididos em 3 categorias, a qual chamamos de 1xx, 2xx e 3xx. Os testes 1xx são testes no nível de linguagem de descrição, os teste 2xx são testes de alterações sistemáticas nos elementos da ontologia (rótulos, relações, propriedades, instâncias e hierarquia), e os teste 3xx analisam alinhamentos com ontologias externas. É importante destacar que a faixa de testes 3xx representa o cenário mais próximo do real, pois objetiva alinhar a ontologia de referência com ontologias reais sobre referências bibliográficas.

Para a execução desse experimento, foram desenvolvidas três versões da abordagem. A primeira versão, denotada por PP Simples, não realiza a busca local. Assim, apenas as operações de movimento foram executadas, fazendo com que a melhor presa permanecesse imóvel até que seja ultrapassada por uma melhor solução. A segunda versão, denotada por PP Simples + BLI, faz com que a melhor solução seja submetida à busca local, avaliando a vizinhança e seguindo para o melhor aprimorante de forma iterativa, enquanto um vizinho melhor pudesse ser encontrado. Já a terceira versão, denotada por PPA, reproduz o comportamento do algoritmo com o movimento da população de acordo com o que foi apresentado e uma busca local simples na melhor presa, fazendo apenas um movimento na direção do melhor vizinho uma única vez a cada rodada. Para os experimentos, as constantes foram ajustadas como: probabilidade de seguir em 50%,  $G = 0.005$ ,  $\lambda_{MIN} = 1$ ,  $\lambda_{MAX} = 20$ ,  $\tau = 0.09$ ,  $\beta = 1$ ,  $\omega = 1$ . Onde os valores de  $\beta$  e  $\omega$  são recomendados por [Tilahun and Ong 2015] e os demais definidos empiricamente.

O sistema foi avaliado através das métricas de Precisão, Cobertura e *Medida-F*. A Tabela 1 apresenta a média de cada métrica para os três modelos dentro das faixas de teste do *benchmark*. O melhor resultado observado foi o do terceiro modelo, que contempla o movimento dos indivíduos no espaço de solução e a busca local executada na melhor presa que retorna o melhor vizinho imediato a essa presa, caso o mesmo exista.

Como o terceiro modelo apresentou resultado médio melhor que os outros, foi realizada uma análise mais aprofundada neste modelo. Uma vez que o algoritmo possui fatores randômicos, a execução foi repetida a fim de avaliar a estabilidade do modelo. Os

**Tabela 1. Resultados dos testes nos três modelos**

	Faixa de instâncias	Valores Médios dentro da faixa		
		Precisão	Cobertura	<i>Medida-F</i>
PP Simples	101-104	1,000	1,000	1,000
	201-247	0,920	0,920	0,920
	301-304	0,763	0,736	0,749
	Média Total:	<b>0,913</b>	<b>0,915</b>	<b>0,914</b>
PP Simples+BLI	101-104	1,000	1,000	1,000
	201-247	0,939	0,946	0,942
	301-304	0,823	0,790	0,806
	Média Total:	<b>0,936</b>	<b>0,938</b>	<b>0,937</b>
PPA	101-104	1,000	1,000	1,000
	201-247	0,946	0,952	0,949
	301-304	0,820	0,786	0,802
	Média Total:	<b>0,941</b>	<b>0,943</b>	<b>0,942</b>

resultados médios para um conjunto de execuções são apontados na Tabela 2.

**Tabela 2. Resultados médios do conjunto de execuções no modelo PPA**

	Média	Desvio Padrão
Precisão	0,90678125	0,0562465884
Cobertura	0,90844791	0,0560783869
Medida-F	0,90703632	0,0558457086

Ao comparar os resultados do PP Simples com o PP Simples+BLI, é possível ver que a busca local iterativa tem impacto positivo na qualidade das soluções, como apontam as medidas das faixas 2xx e 3xx, aprimorando a melhor solução encontrada pelo PP Simples. O resultado esperado foi encontrado, onde a aplicação da meta-heurística da forma como foi definida (PPA) superou as demais, o meta-alinhador utilizando o presa-predador como calibrador de pesos apresentou resultados promissores nos parâmetros empregados, tais resultados tendem a melhorar com a customização das funções de movimento, uma proposta de trabalhos futuros.

No que tange ao cenário de aplicação, o PPA pode ser empregado onde é necessário atribuir alinhamentos candidatos a ontologias com poucas referências e retornar uma primeira versão de alinhamento para os engenheiros. Outro cenário, onde espera-se que o desempenho seja melhor, é aquele em que se deseja refazer um alinhamento já existente dado que houve alguma alteração em uma das ontologias. A expectativa de melhores resultados se baseia no fato de que o conjunto de alinhamentos de referência de entrada é maior, provendo uma capacidade descritiva maior para o algoritmo. Da maneira como foi desenvolvido, o PPA permite que o usuário possa controlar o esforço da abordagem, através dos parâmetros, isso significa que a solução pode ser produzida de acordo com a necessidade do utilizador, por consequência a qualidade da solução também sofre impacto. Logo, com a configuração adequada é possível que o PPA possa ser utilizado em tempo de execução, o que não foi um problema nos testes realizados, dado o tamanho das ontologias. Ontologias com maior volume de informação, de larga escala, requerem

um tempo maior tanto no calibramento quanto no meta-alinhamento em si. Como este trabalho executa as duas tarefas em tempo de execução, a performance do sistema tende a cair, podendo inviabilizar sua aplicação nesse cenário. A OAEI possui conjuntos de testes que lidam com o alinhamento de ontologias em larga escala.

## 6. Considerações Finais

Este trabalho apresentou uma abordagem para meta-alinhamento de ontologias que faz uso da meta-heurística presa-predador. A natureza do problema em conjunto com a forma como foi modelado permite que, quando uma boa solução for encontrada, os pesos associados a essa solução possam ser persistidos e utilizados para reproduzir o experimento em ontologias que atendam aos critérios estabelecidos pelo teste. Uma vez que os relacionamentos entre as entidades das ontologias estão estabelecidos, não há necessidade de reprocessar o algoritmo para encontrar uma nova solução, como ocorre, por exemplo, em problemas de roteamento de veículos, onde novas soluções devem ser geradas com frequência pois as restrições variam ao longo do tempo. Logo, encontrar uma solução perto da ótima dentro de um conjunto de execuções em tempo polinomial é suficiente, pois os parâmetros encontrados podem ser aplicados em qualquer momento.

O método de alinhamento utilizado, que seleciona sempre o par de entidades com maior similaridade, tende a ser substituído em trabalhos futuros, uma vez que um método que explore as similaridades encontradas e busque o conjunto de alinhamentos que maximize o somatório das similaridades envolvidas pode contribuir com uma melhor estabilidade do modelo, considerando que a similaridade correta pode não ser a de maior valor encontrado mas está situada entre as melhores.

Como limitações do trabalho, podemos destacar que o *benchmark* da OAEI é criado de forma sistemática com base em dados sintéticos, o que dá margem para que possam ocorrer variações de desempenho e acurácia para ontologias do mundo real. Ainda assim, o *benchmark* é difundido na literatura, sendo a principal referência para testes de alinhadores de ontologias. O desempenho do algoritmo desenvolvido está atrelado às configurações definidas pelo usuário. Logo, é importante realizar um estudo sobre os valores ideais desses parâmetros. Ainda, a abordagem proposta está baseada em uma única função objetivo. Porém, objetivos conflitantes poderiam ser definidos, por exemplo, adicionar como objetivo a maximização do número de correspondências totais. Assim, abordagens multi-objetivos, conhecidas de outras áreas da computação, poderiam ser exploradas e adaptadas para essa abordagem.

## Referências

- Achichi, M., Cheatham, M., Dragisic, Z., Euzenat, J., Faria, D., Ferrara, A., Flouris, G., Fundulaki, I., Harrow, I., Ivanova, V., et al. (2016). Results of the ontology alignment evaluation initiative 2016. In *OM: Ontology Matching*, pages 73–129. No commercial editor.
- Akbari, I., Fathian, M., and Badie, K. (2009). An improved mlma+ and its application in ontology matching. In *Innovative technologies in intelligent systems and industrial applications, 2009. CITISIA 2009*, pages 56–60. IEEE.
- Bock, J. and Hettenhausen, J. (2012). Discrete particle swarm optimisation for ontology alignment. *Information Sciences*, 192:152–173.

- Euzenat, J., Shvaiko, P., et al. (2007). *Ontology matching*, volume 18. Springer.
- Farinelli, F. and Almeida, M. (2014). Interoperabilidade semântica em sistemas de informação de saúde por meio de ontologias formais e informais: um estudo da norma openehr. *XVII Encontro Nacional de Pesquisa em Ciência da Informação*, 17(1).
- Hai, D. et al. (2007). *Schema matching and mapping-based data integration: Architecture, approaches and evaluation*. VDM Verlag.
- Joslyn, C. A., Paulson, P., and White, A. (2009). Measuring the structural preservation of semantic hierarchy alignments. In *Proceedings of the 4th International Conference on Ontology Matching-Volume 551*, pages 61–72. CEUR-WS. org.
- Laumanns, M., Rudolph, G., and Schwefel, H.-P. (1998). A spatial predator-prey approach to multi-objective optimization: A preliminary study. In *Parallel Problem Solving from Nature—PPSN V*, pages 241–249. Springer.
- Martinez-Gil, J. and Aldana-Montes, J. F. (2011). 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. *The Knowledge Engineering Review*, 27(4):393–412.
- Otero-Cerdeira, L., Rodríguez-Martínez, F. J., and Gómez-Rodríguez, A. (2015). Ontology matching: A literature review. *Expert Systems with Applications*, 42(2):949–971.
- Shi, Y. and Eberhart, R. (1998). A modified particle swarm optimizer. In *IEEE International Conference on Evolutionary Computation*, pages 69–73.
- Sorensen, K., Sevaux, M., and Glover, F. (2017). A history of metaheuristics. *Handbook of Heuristics*.
- Souza, J. F. (2012). *Uma abordagem heurística uni-objetivo para calibragem em meta-alinhadores de ontologias*. PhD thesis, Pontifícia Universidade Católica do Rio de Janeiro.
- Souza, J. F., Siqueira, S. W. M., Melo, R. N., and de Lucena, C. J. P. (2014). Análise de abordagens populacionais para meta-alinhamento de ontologias. In *iSys-Revista Brasileira de Sistemas de Informação*, pages 75–97.
- Tilahun, S. L. and Ong, H. C. (2015). Prey-predator algorithm: A new metaheuristic algorithm for optimization problems. *International Journal of Information Technology & Decision Making*, 14(06):1331–1352.
- Wang, P. and Wang, W. (2016). Lily results for oaei 2016. In *OM@ ISWC*, pages 178–184.
- Xue, X. and Tang, Z. (2017). An evolutionary algorithm based ontology matching system. *Journal of Information Hiding and Multimedia Signal Processing*.