1
Introdução

Com o advento da Internet, o acesso à informação foi disseminado em escala global. Atualmente, essa disseminação permite que qualquer pessoa consiga realizar consultas e obter respostas geralmente satisfatórias.

Apesar disso, grande parte da informação está espalhada e desestruturada. No geral, não é possível – de forma sistemática – comparar diferentes bancos de dados e entender que eles tratam do mesmo domínio. Por exemplo, duas grandes lojas de comércio eletrônico como a *Amazon* e *Barnes and Noble* possuem um banco de dados com informações semelhantes mas que não são comparadas por dificuldade de se gerar um procedimento automatizado de comparação.

Esse problema ocorre em diferentes domínios. Vários órgãos de dados geográficos possuem dados complementares que poderiam ser integradas para gerar informações mais completas e precisas. Empresas que estão se unindo poderiam integrar os diferentes bancos de dados para facilitar seu processo de união.

Todos esses problemas são resolvidos da mesma maneira: alinhando seus esquemas de bancos de dados para entender que informações são equivalentes em ambos e, dessa maneira, possibilitar a integração da informação examinada.

1.1.
Alinhamento de esquemas

Um *esquema conceitual de banco de dados* ou, simplesmente, um *esquema* é uma descrição em alto nível de como os conceitos de um banco de dados estão organizados. O alinhamento de um esquema de origem S com um esquema de destino T define conceitos em T nos termos dos conceitos de S.

O problema de encontrar o alinhamento entre esquemas se torna um desafio quando diferentes vocabulários são usados para se referir ao mesmo conceito do mundo real [7]. Neste caso, uma abordagem conveniente, às vezes chamada de *extensiva*, *baseada em instâncias* ou *semântica*, é detectar como um mesmo

objeto do mundo real é representado em diferentes bancos de dados e usar a informação obtida desta maneira para alinhar os esquemas. Essa abordagem é fundamentada na interpretação tradicionalmente aceita de que “termos têm a mesma extensão quando denotam a mesma coisa” (tradução livre) [29].

Neste trabalho, a proposta é a criação de uma infraestrutura de software que permite a implementação de várias técnicas (algoritmos) de alinhamento de esquemas. No decorrer do texto, utilizaremos como exemplo o algoritmo de alinhamento baseado em instâncias apresentado pela primeira vez em [23]. Resumidamente, a técnica é baseada em um algoritmo que usa funções de similaridade para avaliar a proximidade semântica dos elementos de dois diferentes esquemas.

A infraestrutura tem o objetivo de acomodar diferentes algoritmos de alinhamento. Para fomentar a comparação entre eles, a ferramenta deve possuir boa capacidade de armazenamento dos dados gerados por esses algoritmos (proveniência) de forma que seja possível comparar o meio e o fim das técnicas. Por exemplo, algoritmos podem ser divididos em etapas que realizam operações complementares. Essas operações podem ser armazenadas de modo que o operador consiga perceber quando uma delas está prejudicando todo o processo. Além disso, o resultado final obtido pode ser comparado entre as técnicas para identificar qual é a mais eficiente e eficaz.

Apesar de, com algumas alterações, o *Matchmaking* poder ser utilizado com qualquer outra tecnologia de banco de dados, nesse trabalho a infraestrutura foi criada focada em esquemas OWL (*Ontology Web Language*) que será descrita melhor no capítulo 2.

1.2. Trabalhos relacionados

Rahm e Bernstein [30] apresentam uma pesquisa inicial de técnicas de alinhamento de esquemas. Euzenat e Shvaiko [10] tratam do caso de técnicas de alinhamento de ontologias. Bernstein e Melnik[2] listam os requisitos para sistemas de gerenciamento de modelos que suportam alinhamento de esquemas.

De acordo com estes trabalhos, técnicas de alinhamento de esquemas podem ser classificadas como *baseadas no esquema*, *extensivas* (ou *baseadas em*

instâncias) ou *hibridas* [30]. Nas técnicas baseadas no esquema, as evidências para gerar elementos de alinhamento são extraídas das definições do esquema, como nomes, descrições, tipos de dados, relacionamentos e afirmações. Por exemplo, a propriedade *Cliente.nome* do esquema de origem pode ser correspondente à propriedade *Comprador.nomeCompleto* do esquema de destino, já que seus nomes (‘*nome’* e ‘*nomeCompleto*’) são sintaticamente similares.

Na técnica extensiva (ou baseada em instância), as evidências são extraídas a partir dos dados associados aos conceitos dos esquemas. Por exemplo, se as propriedades *Classe1.propriedade1* e *Classe2.propriedade2* do banco de dados de origem e de destino, respectivamente, apresentam os valores apresentados na Figura 1, então, apesar de os nomes das propriedades não estarem sintaticamente relacionados, as duas propriedades parecem ser correspondentes porque elas possuem o mesmo conjunto de valores. Além disso, se existe um conhecimento prévio de que a propriedade *Livro.autor* pode assumir valores do conjunto apresentado na Figura 1, então se pode inferir que as duas propriedades anteriores devem ser equivalentes a *Livro.autor* porque seus conjuntos de valores são semelhantes.

Classe1.propriedade, *Classe2.propriedade* ∈ {“Guimarães Rosa”,”Machado de Assis”,”William Shakespeare”}

Livro.autor ∈ {“Edgar Allan Poe”,”Guimarães Rosa”,”Machado de Assis”,”William Shakespeare”}

Figura 1 - Conjunto de propriedades de Classe1.propriedade1, Classe2.propriedade2 e Livro.autor.

Por fim, técnicas híbridas combinam as evidências das duas técnicas anteriores para produzir os alinhamentos.

Melnik e Garcia-Molina [28] propuseram uma técnica baseada em esquemas. Os esquemas de origem e de destino são modelados como um grafo de maneira que cada nó representa uma possível correspondência entre dois conceitos dos esquemas. O alinhamento entre os esquemas é o conjunto de alinhamentos com mais conceitos similares. A similaridade de cada nó depende

das características dos seus conceitos e da similaridade de seus vizinhos. Um processo iterativo propaga as similaridades dos nós para os seus vizinhos até que todas as similaridades convirjam em um valor estável.

Madhavan et al. [25] modelam os esquemas de origem e de destino como dois grafos, onde cada nó representa um conceito do esquema. O alinhamento de esquemas consiste no par de nós mais similar. A similaridade de cada par depende da similaridade sintática dos nomes dos conceitos e da estrutura do sub-grafo abaixo dele.

Do e Rahm [8] apresentam um sistema de alinhamento de esquemas chamado COMA. Sendo uma plataforma que combina múltiplos algoritmos de uma maneira flexível, ela provê um grande espectro de algoritmos de alinhamento individuais. Desse modo, uma abordagem nova é a possibilidade de reutilizar os resultados de outras operações de alinhamento e combiná-las.

Doan et Al. [9] propõem um sistema (LSD – *Learning Source Description system*) que emprega e estende as técnicas de aprendizado de máquinas atuais para encontrar os alinhamentos de forma semi-automática. Ele primeiro solicita que o usuário apresente os alinhamentos semânticos de um pequeno conjunto de fontes de dados e então usa esses alinhamentos, juntamente com os dados, para treinar um conjunto de agentes. Cada agente explora um tipo diferente de dado, seja do esquema de origem ou dos seus dados. Depois que os agentes estiverem treinados, o LSD primeiramente descobre os alinhamentos semânticos para uma nova fonte de dados aplicando esses agentes e então combina as suas informações utilizando um meta-agente.

Wang et al. [34] descrevem uma técnica baseada em consultas de sondagem para alinhar bancos de dados da Web. Essa técnica necessita da intervenção humana para selecionar um conjunto típico de instâncias usado na sondagem. Brauner et al. [6] aplicam esta idéia para alinhar *Web Services* de bancos de dados geográficos.

Brauner et al. [5] descrevem um algoritmo de alinhamento baseado no cálculo da similaridade entre domínios de propriedades de diferentes bancos de dados da Web.

Casanova et. al. [17] descrevem uma abordagem para alinhamento de catálogos baseado em funções de similaridade que se aplicam ao alinhamento de tesauros e de propriedades.

Leme et al. [19,18] descrevem uma abordagem de alinhamento de esquemas em OWL baseado em funções de similaridade que se aplicam a um subconjunto de OWL com a mesma expressividade da UML.

Madhavan et al. [26] propõem o uso de um conjunto de esquemas e alinhamentos para ajudar nos algoritmos de alinhamento. Os autores usam algoritmos preditivos que calculam a similaridade entre os elementos dos esquemas, adaptado na arquitetura do PayGo [27].

Bilke e Naumann [3] propõem uma técnica para alinhamento de propriedades baseado em instâncias duplicadas. Brauner et al. [4] adaptaram a mesma idéia para alinhar dois tesauros. Primeiramente, o método encontra os K pares de tuplas mais similares e então compara os valores dos pares de propriedades. Os candidatos a alinhamento são escolhidos como sendo os pares de propriedades que possuem mais valores em comum. O processo de localizar dados duplicados considera que cada tupla é representada por uma única cadeia de caracteres. Duas tuplas são consideradas equivalentes se suas representações como cadeias de caracteres são similares. Entretanto, se o número de propriedades de cada banco de dados é drasticamente diferente, duas linhas equivalentes podem ter representações muito diferentes, resultando em um alinhamento fraco.

Finalmente, Udrea et al. [33] apresentam o algoritmo ILIADS (*Integrated Learning In Alignment of Data and Schema*) para alinhamento de ontologias. O método combina agrupamentos similares e inferência lógica incremental. Para o calculo da similaridade, o algoritmo leva em conta a estrutura léxica do esquema e as informações dos dados. No agrupamento de entidades da ontologia (classes, propriedades e instâncias), novos axiomas e conseqüências lógicas são criados. Por exemplo, se enquanto estiver alinhando duas ontologias de medicina A e B, o algoritmo adicionar o axioma

```
(A:E-Coli-Poisoning, owl:sameAs, B:E-Coli),
```

ao conjunto de axiomas existente

```
(A:discoveredBy, owl:inverseOf, B:discoverer),
(A:discoveredBy, owl:Type, owl:FunctionalProperty),
(B:T.S.Esherich, B:discover, B:E-Coli) e
(A:E-Coli-Poisoning,A:discoveredBy, A:TheodorEscherich)
```

Isso irá produzir a seguinte implicação:

```
(A:TheodorEscherich, owl:sameAs, B:T.S. Esherich)
```

1.3. Organização do trabalho

Esse trabalho está organizado da seguinte forma. O capítulo 2 introduz os principais conceitos utilizados ao longo do texto, além de descrever a técnica de alinhamento de esquemas que usaremos de exemplo ao longo da dissertação. O capítulo 3 apresenta o *Matchmaking*, uma infraestrutura para alinhamento de esquemas, descreve sua arquitetura e como ela foi implementada. O capítulo 4 descreve a interface de usuário criada para interagir com o *Matchmaking*, além de apresentar detalhadamente como o algoritmo descrito no capítulo 2 foi implementado na ferramenta. Por fim, o capítulo 5 traz as conclusões desse trabalho e indica possibilidades de trabalhos futuros.