Consulta de banco de dados em francês: fr2sql

Papel Conference · Junho 2015

CITAÇÕES
0

LÊ
2.165

2 autores , incluindo:

Universidade de Grenoble

14 PUBLICAÇÕES **29** CITAÇÕES

Ver Perfil

fr2sql: banco de dados Data Query em francês

Benoit Couderc, eu er'emy Ferrero

Consulta de banco de dados em francês: fr2sql

Bento Couderc [1] Jeremy Ferrero [2, 3]
(1) Aix Marselha University, Marselha, França (2) Compilatio, 276, rue du Mont Blanc, 74540 Saint-Felix, França
(3) LIG-GETALP, Université Grenoble Alpes, França
benoit.couderc@etu.univ-amu.fr, jeremy.ferrero@imag.fr

Resumo. **Bases de dados estão se tornando mais comum e estão se tornando cada vez mais importante nas aplicações e sites atuais. Eles** muitas vezes têm de ser usados por pessoas sem competência alta no assunto e não sei exatamente a sua estrutura. É por isso que os tradutores de linguagem natural para consultas SQL são desenvolvidos. Infelizmente, a maioria desses tradutores estão confinados a uma única base por causa da especificidade da arquitetura do mesmo. Neste trabalho, propomos um método para ser capaz de consultar qualquer banco de dados de perguntas em francês. Nós avaliamos nossa aplicação em duas bases na estrutura diferente e nós também mostram que ele suporta mais operações do que a maioria dos outros tradutores.

Abstract.

consulta de banco de dados em francês: fr2sql

Bancos de dados são cada vez mais comuns e são significativos Tornando-se cada vez mais em aplicações reais e sites. Eles muitas vezes utilizado por pessoas que não-têm grande competência neste domínio e que não sabem exatamente sua estrutura. É por isso que os tradutores de linguagem natural para consultas SQL são desenvolvidos. Infelizmente, a maioria desses tradutores é confinada a um único banco de dados, devido à especificidade da base arquitetônica. Neste trabalho, propomos um método para consultar qualquer banco de dados do francês. Nós avaliamos-nosso aplicativo é duas bases de dados diferentes e mostramos aussi que ele suporta mais operações do que a maioria outros tradutores.

palavras-chave: **Banco de dados (DB), consultas SQL, SQL tradutor francês, Interface Humano DB.**

palavras-chave: **Bancos de dados (DB), Structured Query Language (SQL), Tradutor francês para SQL, linguagem natural terfaces in- para bancos de dados (NLIDB).**

1 Introdução

Por muitos anos, os bancos de dados (DB) são inevitáveis para todos os sites ou aplicações que gerenciam grandes quantidades de informação, tais como contas de usuário (bancos, agências de transporte, redes sociais, jogos de vídeo, etc.). Internet tem gradualmente democratizado e popularizou, mas os bancos de dados, por sua vez permanecem abstrato para muitas pessoas. Algumas posições requerem nenhuma administração de dados de treinamento ou de TI ainda precisam de trabalhar em estreita colaboração com bancos de dados, tais como contabilidade ou secretariado por exemplo. Isto é, a fim de que uma pessoa não ter competência no domínio da DB gestão, não pode administrar diretamente, mas pelo menos entender como ele funciona, interagir com ele e acima de executar tarefas simples (consulta, adicionar, deletar), os tradutores da linguagem de consulta de banco de **dados de linguagem natural emergiu. Durante cinqüenta anos (Verde *et al.* 1961), o problema de consultar um banco de dados de linguagem natural é** recorrente e é o assunto de muita pesquisa. A maioria das ferramentas desenvolvidas são muito eficientes, mas infelizmente para a maioria compatível com a linguagem natural única fonte e / ou um banco de dados alvo. Eles são desenvolvidos exclusivamente para ser a interface de um banco de dados e são exclusivamente compatível com ele. Devido à estrutura, vocabulário e convenção de nomenclatura extremamente diferente de uma base para outra, portando bases multi-ferramenta não compatíveis em uma base diferente da prevista

inicialmente é difícil e faria de qualquer maneira ineficaz. É a partir dessa observação de que este artigo tem como objectivo conceber um tradutor para consultar qualquer banco de dados do francês. Tendo definido alguns conceitos e apresentou o estado da arte irá descrever como extrair a informação relacionada com um banco de dados alvo, a fim de conhecer sua estrutura e vocabulário, depois de atravessar esta informação com as palavras-chave pergunta, e gerar a saída da consulta SQL equivalente mais provável. *O pedido será gerado com base na presença, número e ordem de palavras-chave identificadas na sentença digitada pelo usuário.* Para definir apresentará a avaliação da nossa abordagem, comparando as capacidades da nossa aplicação para essas aplicações já existentes e avaliar o desempenho de um conjunto de consultas de teste.

2 A tradução de linguagem natural para SQL

2.1 Consultas

Um banco de dados é um dispositivo de computador que é armazenado um conjunto de informações. Em um banco de dados relacional, a informação é armazenada na forma de matrizes, chamadas de tabelas. Um banco de dados cional rela- pode ter uma ou mais tabelas, ou não ligados entre si. As entradas (dados) são combinados no que se chama as colunas (ou campos). Um grupo de colunas, relativamente a uma entidade (objecto) forma uma mesa.

Um diagrama, também chamado modelo de dados é um diagrama (por exemplo, Figura 4) ou uma descrição textual que descreve a distribuição e a organização de dados dentro de um banco de dados. Ele conta as características de cada tipo de dados e as relações entre eles. Um esquema relacional é a maneira mais comum para descrever um banco de dados relacional.

O SQL (Structured Query Language) é uma linguagem padrão para operações de desempenho (consultas, modificações, apagar, etc.) em bancos de dados relacionais.

O objetivo deste artigo é fornecer uma aplicação para consultar um banco de dados relacional usando uma pergunta em francês. Nós tratamos somente consultando um banco de dados que está usando o comando SELECT, a sintaxe é:

```
SELECT FROM table_list column_list [JOIN
jointure_expression] [WHERE
conditional_expression] [GROUP BY
group_by_column_list] [TENDO
conditional_expression] [ORDER BY
order_by_column_list]
```

A sintaxe de um comando SELECT é sempre construído da mesma maneira. Após a palavra-chave *SELECIONAR* que lista as colunas que contêm as informações que você deseja recuperar. após a *DE* indica como (s) mesa (s) são essa informação. Todas as linhas seguintes marcados em colchetes são opcionais. Ele permite, respectivamente, e, a fim de especificar tabelas adicionais, se as articulações são necessárias, adicionar restrições à ordem consulta e grupo ou valores de retorno. Sabendo disso, quando o usuário insere um tipo de solicitação:

Qual é a idade do estudantes que têm nomear John?

A aplicação deve executar uma consulta como esta:

SELECIONAR idade DE alto ONDE firstname = ' JEAN '

A transição da primeira frase do segundo representa toda a questão do projeto. Há, acima em negrito, *elementos-chave* comum entre a entrada de demanda e o pedido para produzir. É sobre a relação entre estes elementos do que a maioria das ferramentas, incluindo o nosso, estão posicionados.

2.2 Estado da arte

O leitor deve consultar (Androutsopoulos *et al.* 1995) (Cimiano & Minock, 2009) a um estado de arte completa.

Uma das primeiras questões levantadas consultando um banco de dados de usuário por não ter eficiência connais- nesta área é que ele não tem nem a estrutura nem o vocabulário usado na base que que busca questionar. As soluções mais triviais são ou limitar o vocabulário que pode ser usado tanto para empregar uma gramática rigorosa que utilizam regras limitam as frases que ele pode construir. O trabalho de (Rao *et al.* 2010) seguindo o primeiro método, eles forçam o utilizador para introduzir uma questão, formatadas de acordo com o dicionário de palavras específico conhecidos pela aplicação, assim, ser limitado para uma base particular. A Applications franceses WORLD-2000 (Pasero, 1997) e DISC (Sabatier e Pasero, 1998) preferem o segundo método. Estes só trabalhar em um banco de dados pré-set nega sua respectiva base de dados, porque eles têm o vocabulário e todas as regras gramaticais apropriadas para a sua operação só nesta base nele. O problema desta categoria de métodos, de acordo com o estudo (Androutsopoulos *et al.* 1995), é que eles não são adequados para o usuário, que então se sente "preso". É por isso que meios menos restritivos também são utilizados. O trabalho de (Po Pescu *et al.* 2003) que, utilizando nenhuma enciclopédia, requerem que o usuário digite uma frase livre, mas contendo um léxico forma exata ou racinalisé comparação com o DB já são mais permissivas. No entanto, este método ainda requer que o usuário tenha um conhecimento perfeito da estrutura básica, e mais particularmente os seus nomes de colunas e tabelas. Pesquisas mais recentes resolver este problema através da combinação das palavras-chave da base e aqueles da frase digitada pelo usuário a um dicionário palavra específica para a base (Deshpande & Devale,

2012) ou, mais geralmente em um dicionário de sinônimos (Chaudhari, 2013).

Chandra (Chandra, 2006) também relata problemas lingüísticos e ambigüidade. Ele observou que a linguagem usada na pergunta do usuário muitas vezes não é o mesmo que no banco de dados e isso causa problemas para se conectar. O trabalho de (Mohite & Bhojane, 2014) apresentou o mesmo fenômeno, que eles chamam *o problema de ortografia (* erro de ortografia). Isso porque, se o usuário erra no ortografia de uma palavra-chave, palavra que representa uma tabela ou coluna por exemplo, na entrada do aplicativo, ele distorce todo o sistema impedindo uma correspondência a ser encontrado . É por esta razão que os sistemas usando enciclopédias (Chaudhari, 2013) ou aspectos semânticos (Djahantighi *et al.* 2008) são cada vez mais numerosos. *recursos lexicais externos são essenciais para a portabilidade tradutores.*

Chaudhari (Chaudhari, 2013) desenvolveu um tradutor relativamente simples, mas já mais ambivalente. Ele é o conteúdo para identificar o tipo de aplicação (selecionar ou excluir), para transformar os números escritos em números em palavras, remover as aspas e sinais de pontuação, para extrair palavras-chave e construir a consulta em conformidade. Ele usa um dicionário de sinônimos para completar com a mão para ampliar o vocabulário aceito pelo sistema. A desvantagem deste método é que cada vez que você quiser usar esta ferramenta em uma nova base, devemos fazer entradas manuais no dicionário.

Para resolver definitivamente os problemas devido ao vocabulário básico de restrições, algumas pesquisas envolve, além da gestão dos sinónimos, uma aspecto semântico (Djahantighi *et al.* 2008), a fim de encontrar o significado de equivalência. Na maioria das vezes através de métodos de aprendizagem, a fim de estabelecer correspondências entre uma pergunta e uma resposta esperada (e Moschitti Giordani, 2012) ou uma pergunta e um pedido (Giordani & Moschitti, 2009). Para a geração de consultas também várias técnicas referem. Algumas gramáticas uso com regras pré-definidas (Alexander *et al.* 2013), outras gramáticas probabilísticas (Deshpande & Devale, 2012) ou PLC (Kaur *et al.* 2013). Alguns até mesmo usar um sistema de aprendizagem (Giordani & Moschitti, 2009; Minock, 2010; & Moschitti Giordani, 2012). Eles, então, explorar a estrutura do DB para gerar consultas SQL que os candidatos são ordenados do mais plausível para menos plausível com uma SVM-ranker baseado no sistema de anel de árvore (Giordani & Moschitti, 2012).

O fracasso da maioria destes métodos (Rao *et al.* 2010; Pasero, 1997; Pasero & Sabatier, 1998) é que eles só são eficazes em um banco de dados específico, aquele para o qual sua gramática ou dicionário é fornecido como trabalho (Safari & Patrick, 2014) que é apenas funcional em seu banco de dados gestão de uma clínica ou de pesquisa (Chen, 2014) que gere uma base de dados geográfica, ou aplicação (Alexander *et al.*

2013) que tem apenas matrizes regras de equivalência e chaves primárias e as ligações básicos sobre os quais ela trabalha.

lado industrial, no entanto, existe uma maior facilidade de transporte entre bases de dados. A abordagem comum na maioria

produtos operacionais tem como objetivo fornecer uma interface criativa associada a uma gramática semântica (Minock, 2010) que interpreta solicitações do usuário em seu banco de dados como no EnglishQuery ferramenta (Microsoft, 2000) (descrito no (Popescu *et al.* 2003)). English2SQL (Hurricane Electric, 2012), por sua vez, analisa a sentença e determinar o seu significado com um algoritmo não revelada que ignora a gramática. Isso permite que tanto para determinar o significado das frases que são longos e complexos, e também lidar com algumas características, tais como o problema da *restrições mudos*. Este problema recorrente em ferramentas através da realização de correspondências de pesquisa entre a sentença de entrada pelo usuário e as entidades do DB ocorre quando o nome da coluna que deve ser feito coação não é indicado na sentença. A frase a seguir ilustra esse problema.

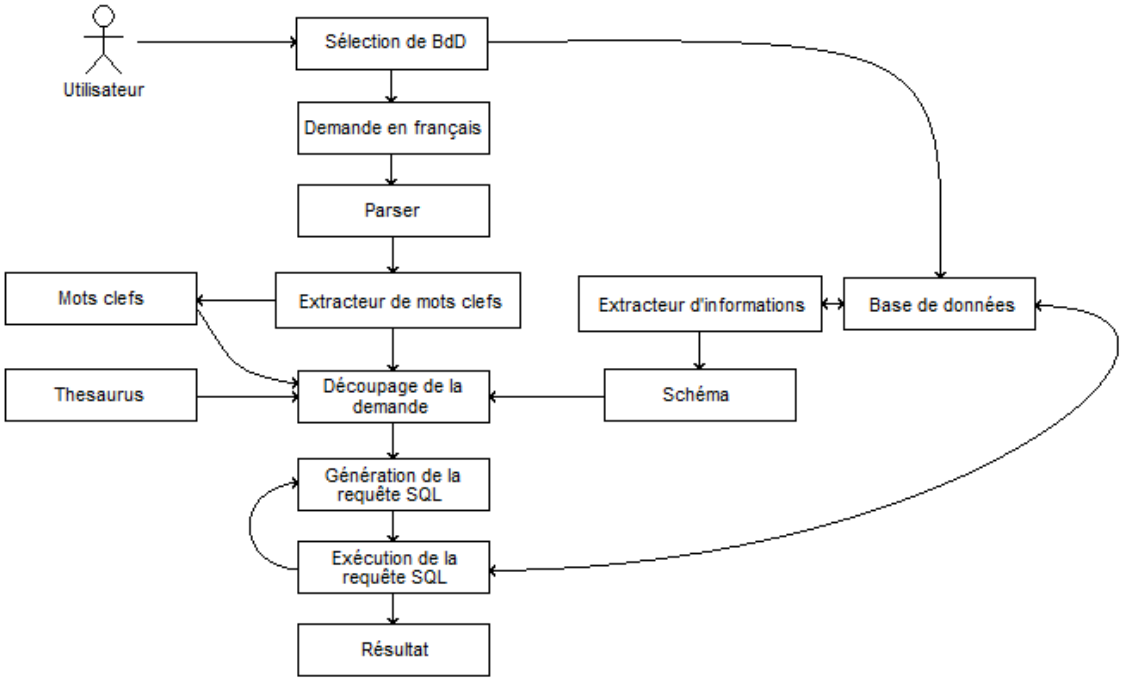
Qual é a idade do estudantes é chamar John?

É mais complexo para operar um jogo a fim de saber qual coluna procurar a palavra *John* dentro busca todas as colunas da tabela *alto* procurando um valor *John* O que a ferramenta English2SQL que considera cada palavra da frase digitada pelo usuário que não é um nome de tabela ou coluna, pode ser um valor de coluna.

Para definir incluem trabalhos da comunidade de banco de dados (Pound *et al.* 2010; Patil & Chen, 2012) que a tentativa de trazer o interrogatório por consulta palavra-chave em linguagem natural. Nós nos posicionar de uma forma original no que diz respeito ao estado da arte, pelas seguintes razões:

- fonte de linguagem natural é francês;
- o método apresentado é portátil (instantaneamente operacional em qualquer banco de dados SQL);
- o método apresentado tem gramática suficiente cientemente permissiva para que o usuário não sente quaisquer restrições lexicais ou sintáticos.

3 A nossa abordagem



F IGURA 1 - Esquema arquitectura sintetizado operação fr2sql do sistema.

A Figura 1 mostra o funcionamento global da ferramenta fr2sql. Inicialmente, o usuário seleciona o banco de dados que iria consultar e retorna uma aplicação no campo fornecido para esta finalidade. A aplicação será então extrair

titulares palavras que significam entrada demanda pelo usuário (seção 3.1). Em seguida, ele irá recuperar a estrutura e as informações necessárias para a sua utilização na base de dados que tenha sido seleccionado (secção 3.2). Usando um dicionário de sinónimos (secção 3.3), uma correspondência é encontrada entre as palavras-chave extraídos a partir do pedido de utilizador e as entidades de base. Uma entrada de procura de corte é realizada de acordo com registo de jogos (secção 3.4). Uma segunda pesquisa é realizada a fim de encontrar palavras que possam indicar plementar SUP- ao selecionar operações (Seção 3.5). Uma vez que o aplicativo tenha determinado que tipo de consulta foi solicitada e que elementos, dependendo do corte operado e registro de jogos, ela gera (seção 3.6).

3.1 Extraindo palavras significativas

Em primeiro lugar, a ideia é recuperar apenas palavras significativas na frase digitada pelo usuário. Na verdade, é importante para, eventualmente, operar um jogo (a correspondência) entre conceitos inseridos pelo usuário e os elementos do banco de dados. Para isso, portanto, mantém especificamente os nomes comuns, que podem ser o nome de uma tabela ou coluna, mas também nomes próprios, números, adjetivos, etc. capaz de actuar sobre eles de um valor de coluna desejado.

Para fazer isso, utilizar a ferramenta TreeTagger (Schmid, 1994), a fim de ser filtrados palavras de paragem de acordo com a sua classe gramatical (preposições, pronomes, determinantes, etc.) e efectuar um Stemming (decorrentes) palavras restantes. Ao considerar a frase:

Qual é a idade do estudantes que têm nomear John?

O filtro deve devolver os itens " ***idade, estudante, primeiro nome, John*** **". A ordem das palavras é preservada e sua importância nas etapas seguintes.**

3.2 Recuperação da arquitetura do banco de dados

A segunda etapa do processo consiste em recuperar a arquitetura (estrutura) do banco de dados em que você vai querer executar consultas e este, a fim de conhecer as entidades (colunas, tabelas, chaves primárias e secundárias, etc.) a fim de permitir que um segundo tempo, em correspondência com as palavras extraídos a partir do pedido do utilizador na secção
3.1.

Dois métodos têm sido implementadas para alcançar este objectivo. O primeiro método é recolher as informações necessárias, consultando o banco **de dados usando consultas SQL como "** ***MOSTRAR TABELAS, SHOW COLUMNS, descreva, etc.*** **"**.

O segundo método, por sua vez, é analisar o fi le backup ou criar o banco de dados. Com este método, um pré-processamento para a conexão banco de dados não é necessário, mas um esquema SQL universal é necessário (alguns comandos são sintaticamente diferente no MySQL ou Oracle, por exemplo). Note-se que fr2sql só é compatível com um banco de dados SQL.

3.3 Emparelhamento de um dicionário de sinónimos

Como observado no estudo (Mohite & Bhojane, 2014), se o usuário não digitar uma frase corretamente orto graphiée ou cuja linguagem utilizada não é exatamente o mesmo que o banco de dados, não correspondência entre a sentença e os elementos básicos serão encontrados e há resultados relevantes serão devolvidos. Portanto, é importante para maximizar o número de palavras que resultarão em correspondência relevante entre uma entrada de procura de palavras-chave e um elemento da DB. Para fazer isso, em paralelo com o processo indicado nas secções anteriores, um dicionário de sinónimos é carregado. Para cada palavra, então você tem acesso a uma série de conceitos que contêm todas as palavras na língua em que ele pode ser substituído. Por exemplo, as palavras "estudantes" e "estudantes" representam o mesmo conceito. Um conceito é uma idéia, um sentido representado por uma palavra ou grupo de palavras. Assim, um conceito é representado por uma palavra transportadora significado lexical e toda a sua possível substituição de palavras na tabela. O objetivo do tradutor é tornar disponível consultando um banco de dados para uma pessoa com conhecimento da estrutura ou palavras-chave (nomes de tabela e coluna) e por isso é provável que use um sinónimo uma palavra

usado na base, em vez da própria palavra. É, portanto, melhor para representar uma palavra para um conceito, uma matriz de todas as palavras pelas quais ele pode ser substituído (uma matriz de sinônimos, incluindo ele próprio) e não apenas por si só. Desta forma, para consultar a tabela *alta,* o usuário pode digitar a palavra *estudante*.

A Tabela 1 mostra uma parte de palavras alternativas correspondentes ao sentido palavras transportador extrai o exemplo frase na secção 3.1.

palavras significativas	palavras Substituição
idade	idade, era, período, geração, ...
estudante	escola, estudante, aprendiz, estudante, estudante de escola secundária ...
primeiro nome	nome cristão, apelido, ...
John	-

T ABLE 1 - Tabela de algumas palavras alternativas disponíveis para a sentença estudada.

Neste estudo, o dicionário utilizado é o v.2.3 thesaurus 20 de dezembro de 2011 LibreOf fi v.3.4. Este recurso está disponível gratuitamente na internet.

A fim de permitir que todos os horários nomear as colunas e tabelas de banco de dados, uma interface thesaurus administração também foi desenvolvido, permitindo que qualquer usuário sem conhecimento especial para adicionar, excluir ou modificar à vontade sinônimos para cada palavra. Então, se a tabela que contém informações de estudantes, por exemplo, nome *ETUD_UNIV_01,* e não sinônimo foi, portanto, automaticamente adicionado ao nome da tabela, o usuário pode inserir manualmente a palavra *estudante* synonymously e equivalência irá atualizar automaticamente também a adição de todos os sinônimos da palavra *estudante*

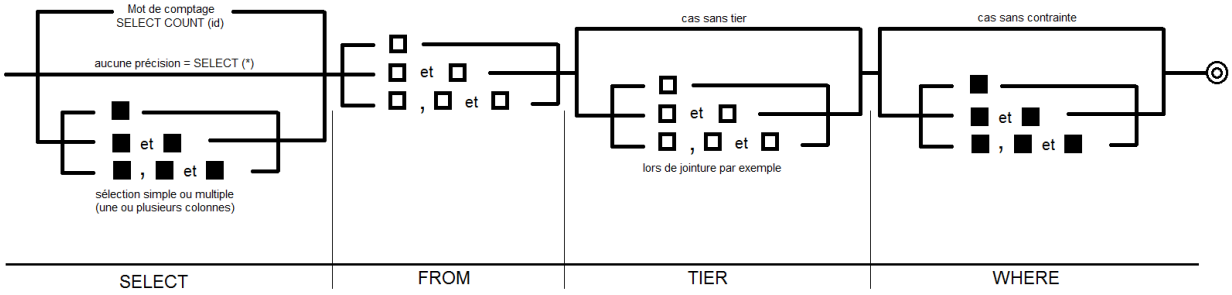
contido no tesauro.

3,4 divisão do pedido

Nesta fase do processo, cada entrada de procura de palavra-chave pelo utilizador é extraído. A aplicação EGA- LEMENTO disponível uma lista de sinônimos para cada uma dessas palavras-chave. A idéia agora é para encontrar uma correspondência entre as palavras-chave na demanda (ou sinónimo) e entidades de base, a fim de realizar uma segmentação da procura em termos de correspondências encontradas e assim conhecida a melhor estrutura a consulta para gerar. Ao fazer a correspondência, todas as palavras são colocadas em minúsculo, todos os sinais diacríticos (acentos, cedilha, etc.) são padronizados. Cada palavra-chave é encontrada com etiquetas de acordo com se é uma coluna ou uma tabela do banco de dados consultada, ou até mesmo algo mais permanece desconhecido por agora. Em primeiro lugar, uma divisão da frase de entrada, mostrada na Figura 2, é executada com base em palavras-chave com a tag "mesa" e "coluna" encontrado na frase, a fim de saber qual o segmento da frase corresponde a qual parte da aplicação para construir . A presença de um segmento Selecione e uma é obrigatória na sentença, a primeira refere-se qual será o tipo de seleção e que (s) elemento (s) exatamente, a segunda especi fi es para onde olhar, o (s) mesa (s), o elemento de selecção. Os segmentos de linha e onde estão, entretanto, opcional. O segmento TIER usado em explícita junta (seção 3.5) e onde especificar, se houver, as tensões sobre a seleção. a fim de saber qual o segmento da frase corresponde a qual parte da aplicação para construir. A presença de um segmento Selecione e uma é obrigatória na sentença, a primeira refere-se qual será o tipo de seleção e que (s) elemento (s) exatamente, a segunda especi fi es para onde olhar, o (s) mesa (s), o elemento de selecção. Os segmentos de linha e onde estão, entretanto, opcional. O segmento TIER usado em explícita junta (seção 3.5) e onde especificar, se houver, as tensões sobre a seleção. a fim de saber qual o segmento da frase corresponde a qual parte da aplicação para construir. A presença de um segmento Selecione e uma é obrigatória na sentença, a primeira refere-se qual será o tipo de seleção e que (s) elemento (s) exatamente, a segunda especi fi es para onde olhar, o (s) mesa (s), o elemento de selecção. Os segmentos de linha e onde estão, entretanto, opcional. O segmento TIER usado em explícita junta (seção 3.5) e onde especificar, se houver, as tensões sobre a seleção. Os segmentos de linha e onde estão, entretanto, opcional. O segmento TIER usado em explícita junta (seção 3.5) e onde especificar, se houver, as tensões sobre a seleção. Os segmentos de linha e onde estão, entretanto, opcional. O segmento TIER usado em explícita junta (seção 3.5) e onde especificar, se houver, as tensões sobre a seleção. *Dependendo do número e posição das palavras-chave na frase, o corte não é o mesmo e, portanto, não dão origem à mesma estrutura saída da consulta.* Note-se especialmente que, se uma aplicação não contém qualquer palavra semelhante a uma tabela, ele será necessariamente inválida e, portanto, gerar um erro.

3.5 Determinação da estrutura consulta

Em seguida, em cada um dos segmentos obtidos ao cortar (secção 3.4), as palavras-chave analisados com etiquetas até agora *desconhecido*. Estas palavras podem ser factores de presença de um referido pedido de contagem, cálculos algébricos, uma negação, etc, ou, alternativamente, um valor que deve ser removida uma restrição. Dessa forma, se uma palavra referindo-se a contar como "quanto" é encontrada no primeiro segmento da frase, a um correspondente para selecionar o sistema de identificação por e pedido para gerar como um pedido de contagem, que é isto é, um SELECT COUNT (*), a primeira parte do primeiro segmento na Figura 2. a aplicação funciona da mesma maneira, com um sistema de reconhecimento de palavras-chave em segmentos SELECT e / ou quando, por muitos outros



F IGURA 2 - corte esquemático feito sobre a sentença digitada pelo usuário, a fim de saber qual o tipo de consulta para saída, os quadrados brancos representam as tabelas e quadrados pretos, colunas.

tipos de operações. A Tabela 2 apresenta de forma voluntária não se limitando determinadas palavras-chave que conduzem a essas operações.

pesquisas	operações
"Quantos", "quantos ele tem," ...	contar
"Não [...] não", "Não [...] não" ...	negação
"Maior que", "maior que" ...	superioridade
"Menos de", "menos do que" ...	inferioridade
"Qual é a soma", "acrescenta," ...	agregado
"O que é a média" ...	média

T ABLE 2 - equivalências palavras-chave identificadas e operações para gerar durante a seleção.

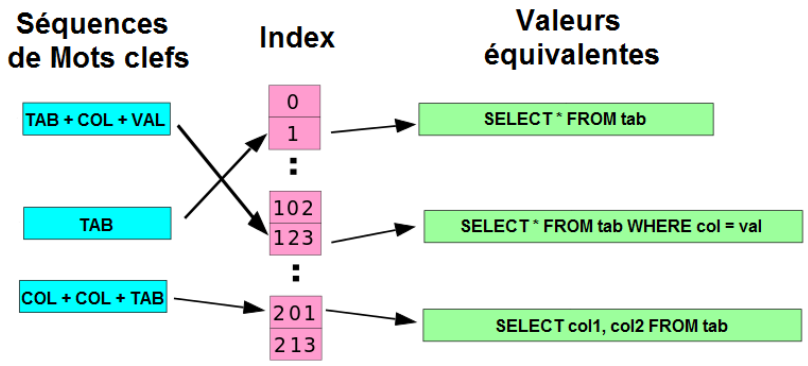
Nós tratamos neste artigo somente associações internas (INNER JOIN). junções existem dois tipos de interior, implícita e explícita. Quando há uma selecção ou restrição numa coluna que não faz parte da mesa da DE, ou seja, quando a mesa à qual pertence a coluna de destino não é mencionado na frase de entrada é uma junção implícita. Deve nesse caso, fazer uma junção entre a tabela da coluna de destino ea tabela do DE, que é especificado na sentença. No caso de uma junção explícita, a tabela na qual você deve executar a associação é especificado diretamente na frase, como no exemplo: "Quais são os alunos com um professor cujo primeiro nome é John? ". É, neste caso, onde o segmento CAMADA aparece na **Figura 2, existe e contém uma ou mais tabelas. Aqui devemos fazer uma junção entre a tabela *alto* e *professor* a fim de ser capaz de selecionar os** alunos enquanto uma restrição sobre os nomes dos professores. Se a seleção da coluna ou restrição não é nem no quadro da DA ou em uma tabela acessada juntando-se a tabela da DA, em seguida, o pedido é impossível construir. O edifício junta-se a aplicação é possível graças a Seção 3.2, como fr2sql conhecer as chaves primárias e estrangeiras de cada tabela pode ser inferido implicitamente ligações reais entre mesas e assim sabe se uma tabela pode ser conectado a outro, e em caso afirmativo, por que (s) mesa (s) passar. Na verdade, fr2sql pode criar junta-se através de mais **tabelas se necessário (como na Figura 4, para aceder à tabela *professor* da mesa *alto* através da mesa *ensinar* e *classe)*.**

3.6 Gerando a consulta usando uma gramática "frouxa"

A permissividade das aplicações existentes é devido ao seu módulo de harmonização demasiado "tolerante", uma vez que procura um conjunto finito de dados em uma área relativamente grande. É então o papel das regras de suas gramáticas rígidas para reduzir ainda mais o espaço de possíveis pedidos para oferecer a mais plausível. Em fr2sql é um pouco o inverso é operado. A correspondência bidireccional primeiro reduz o espaço de possíveis consultas, como ele executa um cruzamento entre um pequeno conjunto de dados e um outro pequeno conjunto de dados. É então uma gramática frouxa que gera o pedido de saída, as regras da gramática não são utilizados para discriminante minar ou direções possíveis consultas, mas apenas para gerar a solicitação previamente determinado pelo correspondente.

Liore correspondência significativamente a discriminação (corresponde) e, assim, permite, posteriormente, usar apenas uma gramática que não pretende ter regras mais exigentes pode ser dada de que as etapas anteriores já filtrada a maioria dos fontes de erro (jogos falsos positivos), mas que só tem como objetivo gerar um pedido de saída.

Devido ao fato de que muitas regras, cada um mostrando a construção de uma possível aplicação é, tomamos a decisão de indexá-los usando uma tabela hash para que adquiram mais rapidamente (olhando em uma tabela indexada por números inteiros que estão ocorrendo mais rapidamente do que procurando uma chave feita de string). Para fazer isso, um número inteiro é atribuído a cada elemento chave da aplicação, uma para elementos do tipo *mesa* 2 para *colunas* 3 para *valores,* 4 para uma contagem de palavras, etc. Por conseguinte, a aplicação de entrada dá origem a uma série de números que são concatenados, formando assim um todo único. 0 corresponde ao facto de que o elemento anterior na cadeia pode estar presente de 1 a N vezes dentro da regra. Então, temos uma imagem de regras indexados por inteiros. Para encontrar a regra equivalente a uma solicitação de entrada, é suficiente para procurar o número inteiro correspondente à estrutura da procura, o que também acontece a ser a chave para o valor que representa a estrutura de consulta para saída na tabela .



F IGURA 3 - Hash estruturas tabela de indexação para gerar consultas de acordo com a estrutura da frase de entrada.

Alguns exemplos de equivalentes são apresentados na Tabela 3. Estes exemplos são mais trivial, eles são destinados apenas aqui para expor a equivalência feita pelo rescaldo de palavras-chave e os tipos de consultas resultantes.

regras	operações notáveis
NB_KW TABELA +	selecção com contagem
TABELA	seleção de todas as colunas na tabela
TABELA + (tabela) + *e* + TABELA	o mesmo conjunto de aplicativos em mais tabelas
TABELA COLUNA +	seleccionando uma única coluna
COLUNA + (coluna) + *e* + TABELA COLUNA +	a selecção de várias colunas
TABELA COLUNA + + VAL	seleção com o estresse

T ABLE 3 - A equivalência entre as palavras-chave identificadas e consultas para gerar.

Agora que o movimento da estrutura é conhecida a saída, é suficiente para substituir as marcas *mesa variável* e *coluna* o seu verdadeiro valor ou nome (correspondência obtida no banco de dados).

O pedido é gerado com base na presença, número e ordem de palavras-chave identificadas na sentença digitada pelo usuário. Chamamos isso de gramática "LAX", pois tem suficientemente regras para dar a impressão ao usuário que ele aceita todas as formas de demanda. Além disso, fazer veri fi cado que a presença e ordem de palavras-chave, quase todos os links de texto ou formas de escrita são possíveis. Este destina-se a ser suficientemente permissiva para o usuário se sente sem restrições lexicais ou sintáticos. Note-se que *problemáticos constrangimentos mudos* não é suportado pelo fr2sql. Aplicações como "Quais são os estudantes chamando John? "Ou" Quais são os estudantes de 18 anos? "Não vai ser devidamente processados pela aplicação. Aqui, a coluna que deve ser feito a coerção é implícita, não é claramente especificada, por isso é impossível para o aplicativo para encontrá-la. Um ser humano pode entender que, no primeiro caso, é o nome do aluno

você quer e no segundo caso, a sua idade, mas o sistema, tal como concebido, não tem como chegar lá. Note também que se a mesma pergunta é feita em várias tabelas no mesmo aplicativo, como na frase "O que são os estudantes e professores com mais de 25 anos? "Então, o sistema é capaz de produzir vários pedidos de saída dentro de dois para atender a demanda (3th linha da tabela 3 e 2th e 3th ramificação de segmento na Figura 2).

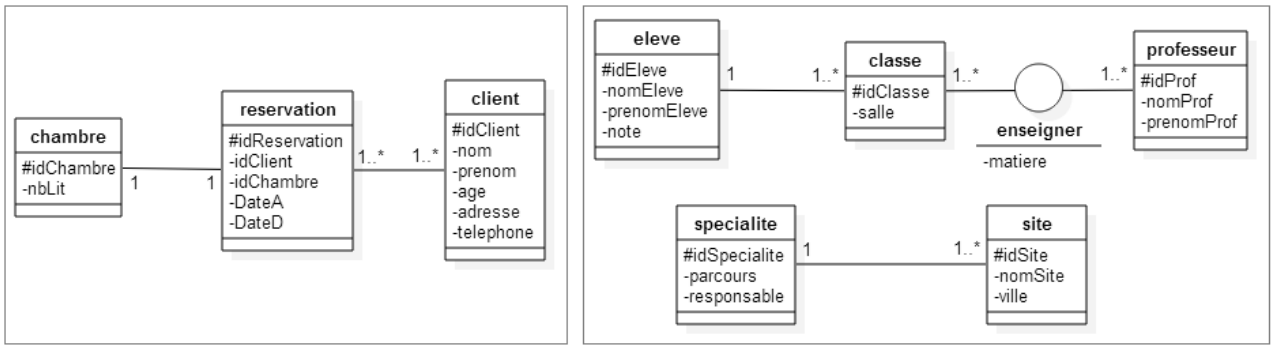
Uma vez que um candidato consulta SQL inicial obtida, tenta executar no banco de dados. Se o pedido for inválido é que ele é mal construída (nome da coluna em vez de mesa, esquecendo-estima, falta de uma coluna, etc.). O sistema então tenta construir de forma diferente, seja na execução, ele retornará um erro novamente, o sistema retorna uma mensagem identificador precisamente o tipo de erro. Este sistema certamente reduz viagens de erros, mas especialmente para conhecer a classe do erro retornado. Com base no trabalho de (Androutsopoulos *et al.* 1995), um dos supostos defeitos de tradutores de linguagem natural para outra língua, não é clara. Com este sistema, este problema é parcialmente resolvido.

4 Avaliação e Testes

4.1 testes básicos

Não há testes básicos para uma ferramenta francês e apoiar uma pluralidade de tabelas que foram feitas disponíveis até à data, é mais de 200 consultas que contêm todos os componentes padrão de uma consulta convencional (único ou múltiplo, contagem e cálculo algébrico, junções, disjunção, junta-se, condições negações, limitações e programação), que foram testadas em duas bases diferentes, a fim de ilustrar a forma de multi-cavidades de portabilidade nossa abordagem. Os pedidos foram todos escritos à mão, incluindo 100 por pessoas com conhecimento da DB e 100 por pessoas que não têm qualquer nesta área. Todos haviam recebido informações sobre mesas grosseiras bases de testes, mas nenhum estava ciente de sua estrutura.

Ambos os bancos de dados de testes têm diferentes estruturas e convenções de nomenclatura. A Figura 4 mostra os padrões de relacionamento dessas bases, são reprodutíveis e utilizável para o trabalho futuro.



F IGURA 4 - bancos de dados de teste do relacionamento diagramas.

4.2 resultados

Nós adotamos a definição de *exatidão* e *retorno* descrito em (Minock *et al.* 2008) (Popescu *et al.* 2003). A entrada de sentença pelo usuário pode solicitar a geração de várias solicitações ou pode produzir nenhuma consulta.

$$\text{precisão} = \frac{\text{cartão (respostas correctas) retornados}}{\text{cartão (respostas devolvidas)}}$$

$$\text{recordação} = \frac{\text{cartão (respostas correctas) retornados}}{\text{cartão (frases de entrada)}}$$

Pode ser encontrada na tabela 4 que o nosso método oferece suporte como muitos, se não mais operações do que as ferramentas existentes, sendo compatível instantaneamente em qualquer banco de dados. É também de notar que só fr2sql é compatível com todas as bases e gerencia o sinônimo de modo a não restringir aplicações a um vocabulário usado no banco de dados.

	MUNDO	SQL-HAL	English2SQL	fr2sql
coluna de seleção	SIM	SIM	SIM	SIM
Seleção sobre a mesa	SIM	SIM	SIM	SIM
seleção múltipla	NO	SIM	SIM	SIM
contando	SIM	SIM	SIM	SIM
contagem múltipla	NO	SIM	SIM	SIM
única restrição	SIM	SIM	SIM	SIM
mute restrição	SIM	NO	SIM	NO
disjunção	SIM	SIM	SIM	SIM
conjunção	SIM	SIM	SIM	SIM
restrição cruz	SIM	SIM	SIM	SIM
Datas de gestão	NO	SIM	SIM	NO
arranjo	NO	NO	NO	SIM
comparação	SIM	SIM	SIM	SIM
álgebra	NO	SIM	SIM	SIM
negação	SIM	NO	SIM	SIM
sinonímia	NO	NO	NO	SIM
Associe-selecção	SIM	NO	SIM	SIM
Junte por muito tempo	SIM	NO	SIM	SIM
consultas aninhadas	SIM	NO	SIM	NO
Compatibilidade na base de dados múltiplos	NO	SIM	SIM	SIM
Restringir o vocabulário ou gramática	SIM	SIM	SIM	NO

T ABLE 4 - As operações apoiadas pelos vários traduções de aplicações.

Os resultados na Tabela 5 ilustram o desempenho do nosso método com base no tipo de aplicação. Questões resultantes seleção perfeita mostrar resultados muito melhores (0,957 média F-score) uma acção resultando num (F-score 0,761) participar. Note também que as frases, resultando em consultas aninhadas não são refletidas adequadamente no aplicativo que não consegue simplesmente no momento.

	exatidão	recordação	F-medida
Qualquer tipo de consultas	0939	0850	0,892
única seleções	1	0,969	0984
com juntas	0832	0702	0761
com condições	0987	0880	0930

T ABLE 5 - Desempenho por categoria solicitação fr2sql de aplicação.

5 Conclusões

Embora nós não poderia fazer claramente comparativo, de acordo com o estado da arte, o nosso método mostra resultados globais equivalente para a maioria das aplicações actuais (uma precisão maior do que 0,90 para um lembrete maior
0,85), no entanto, com uma fraqueza nas articulações (0,761 F-medida). Note também a impossibilidade de que para gerenciar restrições mudos e gerar consultas aninhadas.

Em trabalhos futuros, pretendemos tratar os constrangimentos silenciosas, mantendo verbos como palavras-chave e adicionando algumas regras de gramática. Isto, por exemplo, definir que "o aluno chamando John" e significa o mesmo que "o aluno cujo nome é John" ou mesmo como "Student 18" é equivalente a "Student

tendo a idade de 18 anos ". Além disso, pretende-se detectar o idioma da entrada do pedido por parte do utilizador, a fim de usar uma enciclopédia referente ao idioma e ajustar as regras baseadas na linguagem e tornar o sistema robusto para outros que não o francês idiomas.

Para concluir, embora melhorado, embora esta abordagem para consultar qualquer banco de dados SQL, atendendo assim as metas de portabilidade garantidos, mantendo o desempenho dentro da média das aplicações existentes e cobrindo uma vasta gama de 'operações de seleção.

referências

A Lexander R., R UKSHAN P & H AHESAN S. (2013). Natural Language Interface para Web banco de dados (NLWIDB). *Em corr.*

A NDROUTSOPOULOS I., R itchie G. & T HANISCH P. (1995). Interfaces de linguagem natural para bancos de dados - Uma introdução. *Em Journal of Natural Language Engineering,* 1 29-81. C Handra Y. (2006). *Interfaces de linguagem natural para bancos de dados.* Tese de Doutoramento. University of North Texas, EUA. C HAUDHARI PP (2013). Declaração de Linguagem Natural para SQL Query Tradutor. *Em International Journal of Computer Applications,* 82 (5), 18-22. C HEN W. (2014). Parametrizado SQL espacial geográfica de tradução para a Question Answering. *Em Computing Semântica (ICSC) 2014 IEEE International,* p. 23-27. C IMIANO P & H INOCK M. (2009). Interfaces linguagem natural: Qual é o problema? - A análise quantitativa baseada em dados. *Na Conferência Internacional 14 de Aplicações de Linguagem Natural para Sistema de Informação (NLDB)* 5723.

D ESHPANDE AK & D Evale PR (2012). Natural Language Processing Consulta Usando Contexto probabilística gratuito gramática. *No Jornal Internacional de Avanços na Engenharia e Tecnologia,* 3 568-573. D JAHANTIGHI FS, N OROUZIFARD Mr. D AVARPANAHAND S & S HENASSA MH (2008). Usando Processamento calibre Natural Lan de modo a criar consultas SQL. *Em Anais da Conferência Internacional sobre Informática e Comunicação Engenharia,* p. 600-604. G IORDANI A. & M OSCHITTI A. (2009). Mapeamento semântico entre as perguntas de linguagem natural e consultas SQL através sintática emparelhamento. *Em Processamento de Língua Natural e Sistemas de Informação,* 5723 207-221. G IORDANI A. & M OSCHITTI A. (2012). Traduzindo Perguntas para consultas SQL com Generative Analisadores Discriminati- vely Reranked. *Em coling 2012: Posters,* p. 401-410. G REEN BF, W OLF AK, C HOMSKY C & G AUGHERY K. (1961). Basebol: Uma questão-respondente automático. em

Trabalhos apresentados no 09-11 maio de 1961 IRE-ocidental Joint Computer Conference AIEE-ACM, IRE-AIEE-ACM '61 (Ocidental), p. 219-224, New York, NY, EUA: ACM. H URRICANE E lectric IS (2012). English2SQL alimentado por he.net. Software. K AUR J., CHAUHAN B & K OREPAL JK (2013). Implementação do Processador de Consultas Usando Autómatos e Processamento de Língua Natural. *Em International Journal of Scientifique c e Research Publications,* 3.

M ICROSOFT (2000). TechNet: Desenvolvendo com Inglês Query. Software. M INOCK M. (2010). C-frase: Um sistema para a construção de interfaces de linguagem natural robustos para bancos de dados. *Em Dados e Engenharia do Conhecimento,* 69 (3), 290-302. M INOCK M., Ó LOFSSON P & N Aslund A. (2008). Para a construção robustos Interfaces de Linguagem Natural para da- tabases. *Na Conferência Internacional 13 de Aplicações de Linguagem Natural para Sistema de Informação (NLDB)* 5039,

187-198. M OHITE A & B HOJANE V. (2014). Desafios e etapas de implementação do Natural Language Interface para extração de infor- mações de banco de dados. *Em International Journal of Technology recentes e Engenharia (IJRTE)* 3.

P Asero R. (1997). Uma interface de francês para um banco de dados discográfico. Software. P Asero R & S ABATIER P. (1998). Uma interface de francês para um banco de dados sobre os estados do mundo. Software. P ATIL R. & C. HEN Z. (2012). Struct: Incorporando informações contextuais para pesquisa Inglês Consulta é bases de dados relacionais. Em TW G ING, G. Y U, J. L. U & WW 0011, Eds., *CHAVES,* p. 11-22: ACM. P OPESCU AM, E Tzioni O & K Autz H. (2003). Para uma Teoria da Natural Interfaces Idioma para Databases. *Em Proceedings da 8ª Conferência Internacional sobre User Interfaces Inteligentes,* p. 149-157.

P ound J., I Lyas IF & W EDDELL G. (2010). acesso expressivo e flexível aos dados web-Extraído: A linguagem de consulta estruturada com base da palavra-chave. em *Proceedings da ACM Conferência Internacional de 2010 SIGMOD sobre Gestão de Dados,* SIGMOD '10, p. 423-434, New York, NY, EUA: ACM. R AO G., A Garwal C, C HAUDHRY S. K ULKARNI N & P ATIL DS (2010). Natural Language Processing consulta usando gramática semântica. *Na Revista Internacional de Ciência da Computação e Engenharia,* 2 219-223. S Afari G & P ATRICK JD (2014). consulta com base restrita de linguagem natural de bases de dados clínicos. *Em Journal of Biomedical Informatics.*

S CHMID H. (1994). Probabilística Parte-Speech Tagging Utilizando Árvores de Decisão. em *Anais da Conferência Internacional sobre novos métodos de processamento de linguagem.*