

# Comparando a eficácia na recuperação de questionários: QSMatching vs Vector model vs Fuzzy

## Comparing effectiveness in questionnaires retrieval: QSMatching vs Vector model vs Fuzzy

Richard Henrique de Souza<sup>1</sup>, Carina Friedrich Dorneles<sup>1</sup>

<sup>1</sup>Universidade Federal de Santa Catarina (UFSC)  
Florianópolis, SC Brasil

richard.henrique@ufsc.br, dorneles@inf.ufsc.br

**Abstract.** *Elaborating a useful questionnaire represents an important task for descriptive research. Poorly elaborated questions can lead to answers with meaningless or naive interpretations. Therefore, it may be interesting to reuse, partially or totally, questionnaires already created with the same purpose. In this paper we compare QSMatching with the vector model and fuzzy to calculate the similarity between questionnaires and consequently to obtain a ranking of questionnaires according to the user's query. In order to verify the effectiveness, an experiment was carried out comparing QSMatching, vector model and fuzzy. The result of the analysis of the experiment shows that QSMatching is more effective than others models for questionnaires retrieval.*

**Keywords.** *QSMatching approach; vector model; fuzzy; questionnaire; similarity; ranking.*

**Resumo.** *Elaborar um questionário útil representa uma tarefa importante para a pesquisa descritiva. Perguntas mal elaboradas podem levar a respostas com interpretações sem sentido, sutis ou ingênuas. Portanto, pode ser interessante reutilizar, parcial ou totalmente, questionários criados com o mesmo propósito. Neste trabalho, comparamos o QSMatching com os modelos vetorial e fuzzy para calcular a similaridade entre questionários e, conseqüentemente, obter uma ordenação de questionários de acordo com a consulta do usuário. Para verificar a efetividade, foi realizado um experimento comparando as abordagens QSMatching, modelo vetorial e fuzzy. O resultado da análise do experimento mostra que o QSMatching é mais efetivo que outros modelos para recuperação de questionários.*

**Palavras-Chave.** *Abordagem QSMatching; modelo vetorial; fuzzy; questionário; similaridade; ordenação.*

## 1. Introdução

Um questionário de pesquisa é um instrumento de coleta de dados, consistindo de uma série ordenada de questões, respondidas sem a presença do entrevistador. As perguntas variam dependendo das circunstâncias [Sheatsley et al. 1983]. Os questionários podem ser uma fonte para obter dados estatísticos que ajudam a realizar estudos de caso, comparações, argumentos e coletar opiniões. Pode ser evidência da validade dos resultados de uma pesquisa [Waslawick 2014, Lakatos and Marconi 2010]. Um exemplo de uso de questionários é na pesquisa quantitativa-descritiva dentro da pesquisa de campo que se refere ao delineamento ou análise das características de fatos ou fenômenos, a avaliação de programas, ou o isolamento de variáveis principais [Lakatos and Marconi 2010].

Quanto à forma, as perguntas do questionário, em geral, podem ser classificadas em diversos tipos [Sheatsley et al. 1983, Hovy et al. 2000, Lakatos and Marconi 2010, Vieira 2009]. No entanto, o presente trabalho considera a classificação das perguntas em dois tipos: aberto e fechado [Sheatsley et al. 1983, Lakatos and Marconi 2010]. As perguntas abertas admitem um maior número de respostas diferentes entre os entrevistados, onde cada respondente pode responder livremente, permitindo que respostas mais inigualáveis sejam obtidas. Em perguntas fechadas, o pesquisador define as alternativas que podem ser apontadas pelo entrevistado, que deve apontar o que melhor se adapta às suas características, ideias ou sentimentos [Sheatsley et al. 1983].

Conforme discutido nos trabalhos anteriores [Souza and Dorneles 2017a, Souza and Dorneles 2017b, Souza and Dorneles 2018], elaborar um questionário útil representa uma tarefa importante, e não é uma tarefa fácil, dado o tempo e o esforço aplicados em sua construção [Patten 2016]. O questionário deve ser muito bem organizado e conter uma ordem lógica para o entrevistado, evitando estruturas confusas e complexas, ou perguntas muito longas [Villar et al. 2008].

Devido ao cuidado inerente e dificuldades de projetar questionários, pode ser interessante reutilizar um trabalho existente de forma total ou parcial, uma vez que, possivelmente, muitos questionários semelhantes tenham sido desenvolvidos. Do ponto de vista da pessoa que elabora o questionário, existem algumas vantagens na reutilização de questionários similares, tais como: (i) as perguntas poderiam ser incorporadas em um novo questionário, a fim de ajudar o pesquisador a obter mais informações para sua pesquisa; (ii) questionários similares podem conter dados que corroboram com a nova pesquisa, demonstrando tendências ou mudanças ao longo do tempo [Souza and Dorneles 2018]. Desse modo é interessante ter um sistema que recupere questionários a partir do fornecimento de um questionário ou caso o usuário não tenha elaborado o seu questionário o sistema deve permitir a consulta também por uma palavra-chave ou uma frase ou uma pergunta.

O desafio encontrado em um processo de busca de questionários de pesquisa é resolver o problema da heterogeneidade das questões: a mesma questão pode ser organizada de maneiras diferentes [Souza and Dorneles 2018]. Por exemplo, (i) considere a pergunta 4 do questionário da parte A da Figura 1: “Qual foi a sua renda total em 2011?”, enquanto a pergunta 4 do questionário da parte B da Figura 1: “Qual o seu lucro ano passado?”. Observa-se que as duas questões são similares. No entanto, os dois questionários são simi-

lares? Nesse caso, a resposta é sim. Por outro lado, no próximo exemplo: (ii) a pergunta 1 do questionário da parte A da Figura 1, “A qual faixa etária você pertence? a.()Abaixo de 17 b.() 17-20 c.()21-30 d.()31 – 40 ...”, enquanto a pergunta 1 questionário B é “Qual a sua idade?”. Obviamente, neste caso, as questões são tão “genéricas” que não é possível afirmar que os questionários são similares, por isso é importante identificar os questionários que apresentam similaridades em mais de uma questão [Souza and Dorneles 2018].

A		B	
Finanças Pessoais		Planejamento Financeiro	
1. A qual faixa etária você pertence? a.()Abaixo de 17    d.()31 – 40 b.()17 – 20        e.()41 - 50 c.()21 – 30        f.()51 - 60	4. Qual foi a sua renda total em 2011? g.()Acima de 60	1. Qual a sua idade?	2. Qual é a sua nacionalidade?
2. Qual é o seu nível de educação? a.()Analfabeto / Primário incompleto b.()Primário c.()Secundário incompleto d.()Secundário e.()Curso técnico secundário	f.()Bacharelado g.()Mestrado h.()Doutorado i.() Outro: _____	3. Qual é a sua escolaridade?	3. Qual é a sua escolaridade?
3. Você vive com algum empresário ou pequeno empresário? a.()Sim    b.()Não	5. Qual foi a renda familiar de sua casa em 2011? 6. Qual é a sua nacionalidade? 7. Em que tipo de moradia você vive? a.()Morando com parentes b.()Morando com um parente idoso c.()Atualmente no sistema carcerário d.()Casa ou apartamento de propriedade da família	4. Qual o seu lucro ano passado?	4. Qual o seu lucro ano passado?
	8. Qual é o seu estado civil? a.()Casado(a)    c.()Viúvo(a) b.()Solteiro(a)    d.()Outra: _____	5. Você faz viagens a negócios?	5. Você faz viagens a negócios?
	9. Quando você viaja, qual é o motivo? a.()Turismo.    b.()Negócio.	6. Por que você faria um planejamento financeiro?	6. Por que você faria um planejamento financeiro?
		7. Como você construiu sua empresa? a.() dinheiro próprio b.() empréstimo bancário c.() dinheiro de parentes d.() Outros: _____	7. Como você construiu sua empresa? a.() dinheiro próprio b.() empréstimo bancário c.() dinheiro de parentes d.() Outros: _____
		8. Qual é o ramo da sua empresa?	8. Qual é o ramo da sua empresa?

**Figura 1. Exemplos de questionários**

Outro problema é que as palavras-chaves para o entendimento da pergunta podem estar nas alternativas e, portanto, afetam o cálculo da similaridade entre a consulta e o questionário. Por exemplo, (iii) considere a pergunta 9 do questionário da parte A da Figura 1: “Quando você viaja, qual é o motivo? a.()Turismo. b.()Negócio.?”, enquanto a pergunta 5 da parte B é “Você faz viagens a negócios?”. Note que a pergunta 9 do questionário da parte A em conjunto com a alternativa “Negócio” é fortemente similar a pergunta 5 do questionário da parte B. Portanto deve-se levar em consideração a análise das alternativas para averiguar se as perguntas são similares [Souza and Dorneles 2018].

Durante o levantamento do estado da arte, realizado em trabalhos anteriores [Souza and Dorneles 2017a, Souza and Dorneles 2017b, Souza and Dorneles 2018], foram encontrados diversos trabalhos para sistemas de CQA <sup>1</sup>, cujo objetivo é a busca por questões. Contudo, estes trabalhos não tratam de recuperação de questionários como um todo. Neste sentido, como questionários podem ser considerados documentos, torna-se natural tentar utilizar abordagens clássicas de recuperação de documentos em recuperação de questionários. De modo que o modelo vetorial [Souza and Dorneles 2017a] e o modelo fuzzy foram selecionados como promissores candidatos.

O presente trabalho é uma extensão do artigo publicado no SBSI2018 [Souza and Dorneles 2018] <sup>2</sup>. Nesta extensão são apresentados os resultados dos experimentos para avaliar a utilização dos modelos vetorial, fuzzy e da abordagem QSMatching <sup>3</sup>, na identificação e recuperação de questionários similares. Os experimentos consistem em verificar a eficácia no cálculo da similaridade entre questionários, e, conseqüentemente, obter uma ordenação de acordo com a relevância

<sup>1</sup>Community of questions and answers - Comunidade de perguntas e respostas

<sup>2</sup>O artigo publicado no SBSI2018 [Souza and Dorneles 2018] contém a comparação do modelo vetorial com a abordagem QSMatching.

<sup>3</sup>QSMatching é a palavra Matching com os prefixos “Q” e “S”, onde “Q” corresponde à Questionário e “S” corresponde à Similaridade.

do questionário em relação à consulta. Para avaliar a aplicabilidade da abordagem em questionários de pesquisa, é realizado um experimento em um conjunto de questionários reais extraídos da web. Os resultados das três abordagens são comparados e uma análise é realizada de modo a verificar qual abordagem é a mais eficaz. A análise de precisão e revocação mostra que o QSMatching é mais eficaz que os outros dois modelos. As contribuições que o presente trabalho traz em relação aos trabalhos publicados anteriormente são: (i) a comparação da abordagem QSMatching com o modelo fuzzy; (ii) o algoritmo e o detalhamento do cálculo de similaridade do QSMatching.

Este artigo está organizado da seguinte maneira. A Seção 2 apresenta alguns conceitos que guiam o presente trabalho. Na Seção 3, são descritos a abordagem QSMatching para a recuperação de questionários e as definições que compõem as regras para o cálculo da similaridade entre questionários. Experimentos, resultados e metodologia adotados são apresentados na Seção 4. A Seção 5 apresenta os trabalhos relacionados. Finalmente, a Seção 6 mostra conclusões e trabalhos futuros.

## 2. Background

Nessa seção são descritos alguns conceitos básicos que guiam esse trabalho. Inicialmente, a premissa que norteia o presente trabalho é descrita. Posteriormente, uma breve descrição dos modelos vetorial e fuzzy são apresentadas, considerando os conceitos usados nos experimentos.

### 2.1. Recuperação de Questionários

A Recuperação de questionários tem como objetivo ajudar pessoas que gostariam de reutilizar ou pesquisar questionários similares. O presente trabalho considera a seguinte premissa no que tange a recuperação de questionários:

**Premissa:** Recuperação de questionários independente de domínio.

As três abordagens utilizadas no presente trabalho atendem a premissa, e portanto são passíveis de ser comparadas. Vale ressaltar que os modelos vetorial e o fuzzy são comumente utilizados na recuperação de documentos de texto [Baeza and Ribeiro-Neto 2011, Manning and Schütze 1999], e podem ser aplicados na recuperação de questionários, porque um questionário é usualmente um documento de texto [Vieira 2009]. Contudo, conforme explicado na Seção 3.1, o questionário pode ser convertido em uma estrutura em grafo, onde tal estrutura é utilizada pelo QSMatching.

### 2.2. Modelo Vetorial

O modelo vetorial é um dos modelos clássicos de recuperação de informação, onde cada documento é representado por um conjunto de termos, que podem ser constituídos por uma palavra ou grupo de palavras consecutivas em um documento. Antes da indexação, é necessário pré-processar os documentos (normalização, palavras irrelevantes, substantivos, etc) [Baeza and Ribeiro-Neto 2011]. No modelo vetorial, a consulta também é representada como um vetor de termos <sup>4</sup>.

<sup>4</sup>Exemplos que ilustram o modelo vetorial podem serem encontrados nos livros: [Baeza and Ribeiro-Neto 2011, pg 77], [Manning et al. 2010, pg 105], [Kowalski and Maybury 2000, pg 111] e [Liu 2007, pg 217]

Cada termo tem um peso específico para cada documento em uma base de documentos, que é calculada por meio da relação TF-IDF, onde TF é a frequência do termo no documento e IDF é o inverso da frequência do termo entre os documentos na coleção. Através do TF-IDF, é possível calcular a similaridade entre uma consulta e os documentos. Normalmente, a função cosseno é usada para definir a medida de similaridade entre uma consulta e um documento. Assim, a medida de similaridade é proporcional ao cosseno do ângulo entre o vetor que representa o documento e o vetor de consulta [Baeza and Ribeiro-Neto 2011, Manning et al. 2010, Kowalski and Maybury 2000, Liu 2007].

### 2.3. Modelo Fuzzy

A teoria dos conjuntos fuzzy lida com a representação de classes cujos limites não são bem definidos. A ideia principal é associar uma função de associação aos elementos da classe. Essa função recebe valores no intervalo  $[0, 1]$  com 0 correspondendo a nenhuma associação a uma determinada classe e 1 correspondente à participação plena. Assim, a participação em um conjunto Fuzzy é uma noção intrinsecamente gradual. Como resultado, a correspondência de um documento com os termos da consulta é aproximada (ou vaga). Isso pode ser modelado considerando que cada termo de consulta define um conjunto Fuzzy e que cada documento possui um grau de associação (geralmente menor que 1) nesse conjunto [Baeza and Ribeiro-Neto 2011, Kowalski and Maybury 2000].

No presente trabalho, utilizou-se o modelo fuzzy <sup>5</sup> baseado no algoritmo distância de Levenshtein (distância de edição). Por exemplo, para fazer uma pesquisa fuzzy usando a palavra “dourado” como parâmetro de busca, os possíveis retornos são as palavras “ourado” e “dourar” que são semelhantes a ortografia “dourado”.

## 3. QSMatching

**QSMatching** é uma abordagem desenvolvida com o propósito de recuperar questionários de forma ordenada, de acordo com os parâmetros de busca de um determinado usuário. A Figura 2 apresenta uma visão geral do QSMatching. Em geral, o usuário passa um conjunto de perguntas como entrada (que pode ser uma pergunta ou um questionário inteiro). O mecanismo pré-processa a consulta e, em seguida, calcula a medida de similaridade entre a consulta e os questionários armazenados no banco de dados de acordo com regras de similaridades predefinidas. Após o cálculo da similaridade, é realizada a ordenação dos questionários de acordo com o resultado deste cálculo. Finalmente, a lista ordenada dos questionários é retornada para o usuário [Souza and Dorneles 2017b]. Cabe ressaltar que se usuário tenha apenas uma “ideia” do questionário, o usuário pode usar uma ou mais palavras na consulta uma vez que a abordagem prevê o uso de palavras, frases, perguntas ou questionário inteiro.

O QSMatching baseia-se na premissa de que os questionários estão em um banco de dados. Para o cálculo de similaridade é utilizada a medida de “pontuação do questionário” ( $QS$ ), então, o ranking dos questionários é realizado de acordo com o valor ( $QS$ ), onde quanto maior este valor, mais relevante é o questionário.

<sup>5</sup>Exemplos que ilustram o modelo fuzzy podem ser encontrados nos livros: [Baeza and Ribeiro-Neto 2011, pg 95] e [Kowalski and Maybury 2000, pg 32]

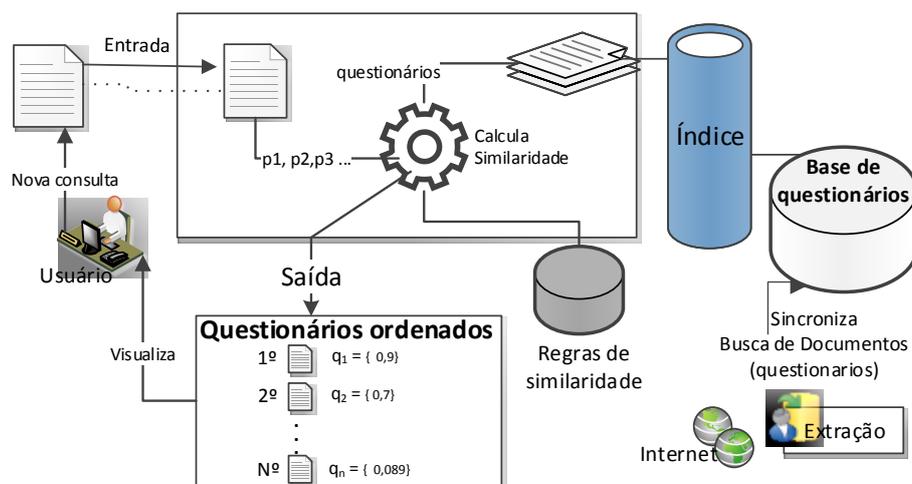


Figura 2. QSMatching

### 3.1. Modelo de Dados

A Figura 3 representa o modelo de dados utilizado para representar um questionário, composto por um conjunto de questões. Nesse modelo, cada questão é associada a um tipo de pergunta (aberto ou fechado). Se o tipo de questão for fechado, a pergunta contém alternativas (um exemplo é mostrado na Figura 1). Cada alternativa representa uma opção de resposta para o entrevistado [Souza and Dorneles 2017a].

De acordo com o modelo descrito na Figura 3 e de acordo com a definição de questionário feita por Picard [Picard 1980], um questionário pode ser representado por um grafo conectado, conforme apresentado na Definição 1.

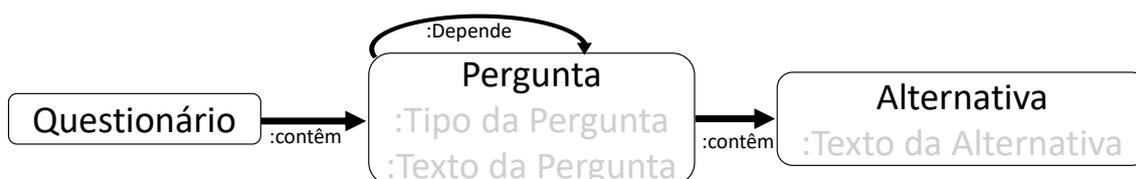


Figura 3. Modelo de dados

**Definição 1 Grafo derivado do questionário**. Um questionário  $Q = (V, T)$  é um grafo, onde  $V$  representa os vértices, de modo que o conjunto de vértices ( $V$ ) tem a partição  $V = P \cup A$ , onde  $P$  consiste de vértices chamados de perguntas e  $A$  consiste de vértices chamados alternativas.  $T$  representa as arestas, tal que  $T \in (P \times A) \cup (P \times P)$  [Souza and Dorneles 2017b].

### 3.2. Similaridade

QSMatching utiliza a média dos valores de similaridade entre perguntas, de modo a considerar suas respectivas alternativas (quando existir alternativas), para definir o valor de similaridade entre uma consulta e um questionário. A medida de similaridade entre as perguntas, por sua vez, é dada pelo número de *tokens iguais*.

Os *tokens* são análogos aos radicais<sup>6</sup> das palavras contidas na pergunta com alternativas. Então, os *tokens* iguais são as palavras que contêm o mesmo *token* (radical), e a contabilização é realizada pela intersecção entre dois conjuntos de *tokens*. Por exemplo, se uma *string* contém a palavra “ganho” e uma outra contém a palavra “ganhar”, então, estas duas *strings* teriam o mesmo *token* “ganh”, e portanto, é contabilizado como um *token* igual. A Definição 2 é usada para obter *tokens* de uma lista de palavras (*string*)<sup>7</sup>.

**Definição 2 Lista de tokens.** Seja  $T_{(s)} = \{\textit{tokens}\}$ . Onde  $s$  é uma *string* e  $T$  é a lista de *tokens*, então  $T_{(s)}$  é a lista de *tokens* da *string*  $s$  após o pré-processamento [Souza and Dorneles 2017b].

Para exemplificar, considere que  $s$  contém a *string* “Quando você viaja, qual é a razão? a() Turismo. b. () Business.”. Neste caso, a lista de *tokens* para a *string*  $s$  é  $T_{(s)} = \{\text{quand, voce, via, qua, raz, turism, busi}\}$ .

A quantidade de *strings* utilizadas no cálculo de similaridade de um questionário dependerá do número de perguntas e do número de alternativas. Para calcular a similaridade entre as perguntas de um questionário, deve-se considerar a estrutura das perguntas, mais especificamente, considerar as alternativas. Neste sentido, no pré-processamento de uma pergunta também é realizado a separação das *strings* conforme a quantidade de alternativas. Para uma pergunta o conjunto de *strings* é obtido de acordo com a Definição 3.

**Definição 3 Conjunto das Listas de tokens.** Seja  $L_{(p)} = \{T_{(p_0)}, T_{(p_{a1})}, \dots, T_{(p_{a_n})}\}$ , onde  $L$  é um conjunto de listas de *tokens* e  $p$  é a pergunta, então  $T_{(p_0)}$  é o lista de *tokens* da pergunta  $p$  sem alternativas, e  $T_{(p_{a1})}$  é a lista de *tokens* da pergunta  $p$  mais a primeira alternativa de  $p$ , e  $T_{(p_{a_n})}$  é a lista de *tokens* da questão  $p$  mais a última alternativa de  $p$  [Souza and Dorneles 2018].

Se  $p$  é a pergunta “Quando você viaja, qual é a razão? a() Turismo. b. () Business.”, então o conjunto das listas de *tokens* conterá três listas de *tokens*: (1) um para a pergunta sem alternativas<sup>8</sup>, (2) outro para pergunta mais a primeira alternativa<sup>9</sup>, (3) o último para a pergunta mais a segunda alternativa<sup>10</sup>.

Após obter as *strings* e as suas respectivas listas de *tokens* é possível realizar a contabilização de *tokens* iguais entre duas *strings* conforme a Definição 4:

**Definição 4 Quantidade de tokens iguais.** Seja  $s_1$  e  $s_2$ , duas *strings*, então a quantidade de *tokens* iguais é dado por  $Ni_{(s_1, s_2)} = |T_{(s_1)} \cap T_{(s_2)}|$ , onde  $Ni$  é a quantidade de *tokens* iguais entre duas cadeias [Souza and Dorneles 2017b].

Considere que a *string*  $s_1$  seja “Você viaja a negócios?” e que a *string*  $s_2$  seja “Quando você viaja, qual é o motivo?”, então, a quantidade *tokens* iguais seria igual a 2

<sup>6</sup>A ideia é similar ao uso de radicais, contudo os *tokens* são o truncamento das palavras que ocorre no pré-processamento de texto.

<sup>7</sup>Definição 2 é uma abstração da implementação do Lucene [luc 2018]

<sup>8</sup> $p_0$  = “Quando você viaja, qual é o motivo?”, então  $T_{(p_0)} = \{\text{quando, voce, via, qua, motiv}\}$

<sup>9</sup> $p_{a1}$  = “Quando você viaja, qual é o motivo? a() Turismo.”, então  $T_{(p_{a1})} = \{\text{quando, voce, via, qua, motiv, turis}\}$

<sup>10</sup> $p_{a2}$  = “Quando você viaja, qual é o motivo? b.() Negócios”, então  $T_{(p_{a2})} = \{\text{quando, voce, via, qua, motiv, negoci}\}$

(sendo os *tokens* “voce” e “via” oriundos das palavras “você” e “viaja” presentes nas duas *strings*).

Aplicando a Definição 3, obtém-se um conjunto de listas de *tokens*. Então, aplicando a Definição 4 para cada lista dentro desse conjunto, gerar-se um outro conjunto com quantidades de *tokens* iguais, conforme a Definição 5. Esses 2 conjuntos, necessariamente, devem possuir a mesma quantidade de elementos.

**Definição 5 Conjunto de quantidade de tokens iguais.** *Seja  $L_{(p_1)}$  e  $L_{(p_2)}$ , dois conjuntos de listas de tokens, então o conjunto de quantidade de tokens iguais é dado por  $Nb_{(L_{(p_1)}, L_{(p_2)})} = \{Ni_{(p_{10}, p_{20})}, \dots, Ni_{(p_{10}, p_{2n})}, Ni_{(p_{11}, p_{20})}, \dots, Ni_{(p_{1n}, p_{2n})}\}$ , onde  $Nb$  é um conjunto de números de tokens iguais entre duas perguntas com cada alternativa [Souza and Dorneles 2018].*

Seja  $p_1$  a pergunta “Você viaja a negócios?” ( $T_{(p_1)} = \{voce, via, negoci\}$ ) e  $p_2$  é a pergunta “Quando você viaja, qual é o motivo? a) Turismo. b. () Negócios.” (onde obtém-se em três conjuntos de listas de *tokens*: (1)  $T_{(p_{20})} = \{quando, voce, via, qua, motiv\}$ , (2)  $T_{(p_{21})} = \{quando, voce, via, qua, motiv, turis\}$ , (3)  $T_{(p_{22})} = \{quando, voce, via, qua, motiv, negoci\}$ ). Ao aplicar as Definições 4 e 5, o conjunto de quantidade de *tokens* iguais é  $Nb = \{2, 2, 3\}$ .

Após contabilizar o número de *tokens* iguais, é necessário normalizar esse valor, visto que pode haver variação do número de *tokens* em cada conjunto de listas de *tokens*. Sendo assim, a Definição 6 é descrita a seguir.

**Definição 6 Pontuação de tokens.** *Sejam  $s_1$  e  $s_2$  duas strings, então a pontuação de tokens é dada por  $Ri_{(s_1, s_2)} = \frac{Ni_{(s_1, s_2)} + Ni_{(s_2, s_1)}}{2}$ , onde  $Ri$  é a pontuação de tokens entre  $s_1$  e  $s_2$  [Souza and Dorneles 2018].*

Como o número de *tokens*  $T(s_1)$  e  $T(s_2)$  podem ser diferentes entre si,  $Ni$  é dividido pelo número de *tokens*  $T(s_1)$  e também pelo número de *tokens*  $T(s_2)$ , sendo que a média deles é a pontuação de *tokens* entre  $s_1$  e  $s_2$ . Além disso, vale destacar que o valor da pontuação de *tokens* é calculado de forma a pertencer ao intervalo  $[0, 1]$ , onde quanto mais próximo de 1, maior será similaridade entre as strings  $s_1$  e  $s_2$ .

Considere que a *string*  $s_1$  seja “Você viaja a negócios?” e que a *string*  $s_2$  seja “Quando você viaja, qual é o motivo?”, então, a pontuação de *tokens*<sup>11</sup> entre  $s_1$  e  $s_2$  é igual a  $0,53 \left(\frac{\frac{2}{3} + \frac{2}{5}}{2}\right)$ .

Conforme visto nas Definições 3 e 5, uma pergunta pode ter um conjunto de *strings*, e, conseqüentemente, ao ser comparada a outra pergunta, pode-se ter um conjunto de pontuações de *tokens* para cada possível combinação. Então, para o cálculo de similaridade, é selecionado o maior valor de pontuação possível entre duas perguntas conforme a Definição 7.

**Definição 7 Pontuação de tokens final.** *Sejam  $p_1$  e  $p_2$  duas perguntas, então a pontuação de tokens final é dada por  $Ets_{(p_1, p_2)} = MAX(Nb_{(L_{(p_1)}, L_{(p_2)})})$ , onde  $Ets$  é a pontuação*

<sup>11</sup>Lembrando que  $T_{(s_1)} = \{voce, via, negoci\}$  e  $T_{(s_2)} = \{quando, voce, via, qua, motiv\}$

de tokens final entre  $p_1$  e  $p_2$  e  $MAX$  é a função que calcula o valor de  $Ets$  comparando o resultado de cada  $Ri$  possível no conjunto de quantidades de tokens iguais  $L_{(p_1)}$  e  $L_{(p_2)}$  e retornando o valor mais alto [Souza and Dorneles 2017b].

A ideia da Pontuação de *tokens* final é verificar o nível de similaridade entre as perguntas da maneira mais simples possível. Nesse sentido, a *pontuação de tokens final* é o valor mais alto entre todas as pontuações de *tokens* das strings que compõem as perguntas que são comparadas.

Considere o exemplo mencionado anteriormente, onde  $p_1$  é “Você viaja a negócios?” e  $p_2$  é “Quando você viaja, qual é o motivo? a.) Turismo b. () Negócio”. Então, tem-se 3 *pontuação de tokens* porque  $p_2$  tem duas alternativas: (1)  $Ri_{(p_1,p_2)} = \frac{\frac{2}{3} + \frac{2}{3}}{2} = 0,53$ ; (2)  $Ri_{(p_1,p_2a_1)} = \frac{\frac{2}{3} + \frac{2}{6}}{2} = 0,496$ ; (3)  $Ri_{(p_1,p_2a_2)} = \frac{\frac{3}{3} + \frac{3}{6}}{2} = 0,75$ . Neste exemplo, a **pontuação de tokens final** é dada por  $Ets_{(p_1,p_2)} = MAX(Nb_{(L_{(p_1)},L_{(p_2)})}) = MAX(0,53; 0,496; 0,75) = 0,75$ . Neste exemplo, é possível perceber a importância de considerar as alternativas no processo de cálculo da similaridade entre as perguntas do questionário, uma vez que sem a alternativa, a similaridade é menor do que avaliando a pergunta em conjunto com a alternativa b.

A Pontuação de *tokens* é calculada para cada conjunto de  $Nb$ . Em seguida, o valor mais alto é escolhido como a pontuação de *tokens* final. Depois de obter a pontuação de *tokens* final para cada pergunta, entre a consulta do usuário e os questionários do *dataset*, é possível calcular a similaridade entre eles conforme a Definição 8.

**Definição 8 Pontuação do questionário.**  $QS = \frac{1}{n} \sum_{i=1}^n MAX(C_{Ets})$ , onde  $n$  é o número de perguntas na consulta.  $C_{Ets}$  representa o conjunto de valores  $Ets$  relacionadas a pergunta da consulta.  $MAX(C_{Ets})$  é a função que retorna o maior valor de  $Ets$  da pergunta da consulta em relação a todas as perguntas do questionário.  $QS$  é a média dos valores oriundos da função  $MAX(C_{Ets})$ .  $QS$  indica a pontuação de similaridade das questões entre consulta e questionário.

Considere que uma consulta contenha duas perguntas (1) “Você viaja a negócios?” e (2) “Qual foi a sua renda ano passado?” e que um questionário para comparação seja o questionário A da Figura 1 (o qual contém nove perguntas), então, para cada pergunta da consulta, é calculada a pontuação de *tokens* final (Definição 7) em relação a cada pergunta do questionário. Portanto, são obtidos dezoito valores que correspondem à pontuação de *tokens* final (nove para cada pergunta da consulta). Após calcular esses valores, é possível obter o maior valor correspondente para cada pergunta da consulta. Então, o valor de  $MAX(C_{Ets})$  para a pergunta 1 da consulta é 0,75 (conforme o cálculo descrito ao exemplificar as Definições 2 até a 7) porque é o maior valor entre os nove valores de pontuação de *tokens* final, e o valor de  $MAX(C_{Ets})$  para a pergunta 2 da consulta é de 0,667 (ver pergunta 4 do questionário). Ao aplicar a Definição 8, obtém-se a similaridade entre a consulta e o questionário cujo o valor, neste caso, é de 0,708 ( $\frac{0,75+0,667}{2}$ ).

Embora as regras de similaridade apresentem algumas semelhanças ao modelo vetorial, o QSMatching difere-se ao considerar a estrutura do questionário, principalmente na separação das *strings* de cada pergunta conforme o número de alternativas, a fim de calcular o maior valor de similaridade possível entre duas perguntas, enquanto o modelo vetorial avalia o questionário sem considerar sua estrutura.

### 3.3. Algoritmo

O algoritmo 1 apresenta de maneira simplificada o cálculo de similaridade entre o questionário e as perguntas que são usadas como parâmetro de consulta de um usuário, o qual será chamado a partir daqui de apenas *query*. O Algoritmo inicia (1) com a função *PontuaçãoDoQuestionario* que recebe um questionário da base de dados (*Q*) e a *query* do usuário (onde pode ser uma palavra, frase, pergunta ou questionário) como parâmetros para o cálculo de similaridade. (2) O algoritmo atribui o valor zero para a variável temporária “somatoria” (a qual é usada para realizar a somatória descrita na Definição 8). (3) Um laço de repetição é inicializado a fim de percorrer por todas as perguntas da *query* (se a *query* conter apenas uma palavra ou uma sentença ou uma pergunta, então, o laço será repetido apenas uma vez, e o algoritmo considerará que nos três casos a *query* contém uma pergunta). (4) O algoritmo atribui o valor zero para a variável “Ets” (a qual será usada para calcular a pontuação de *tokens* final conforme descrito na Definição 7). (5) Um loop de repetição é inicializado a fim de percorrer por todas as perguntas do questionário. (6) A variável “temp” recebe o valor da pontuação de *tokens* final entre a pergunta da *query* com uma pergunta do questionário, conforme a Definição 7, calculado na função “calculaEts”. (7) A variável “Ets” recebe o maior valor de pontuação de *tokens* final possível. Por exemplo, se o questionário contém nove perguntas, então existem nove valores de “Ets” possíveis para cada pergunta da consulta. Dentre os nove valores calculados para uma pergunta do questionário, apenas o maior valor é considerado para o cálculo de similaridade, conforme descrito na Definição 8. (9) A variável “Somatória” recebe a soma dos valores de pontuação de *tokens* final de cada pergunta da *query*, conforme descrito na Definição 8. (11) A variável “QS” recebe o valor de similaridade entre a *query* e o questionário. Finalmente, (12) a função retorna o valor de “QS”.

---

#### Algorithm 1 Similaridade entre Questionários

---

```

1: função PONTUAÇÃOODOQUESTIONARIO(Q, query)
2:   Somatoria  $\leftarrow$  0
3:   para cada pergunta da query faça
4:     Ets  $\leftarrow$  0
5:     para cada pergunta do questionário faça
6:       temp  $\leftarrow$  calculaEts(perguntaDaQuery, PerguntaDoQuestionario)
7:       Ets  $\leftarrow$  MAX(Ets, temp)
8:     fim para
9:     Somatoria  $\leftarrow$  Somatoria + Ets
10:  fim para
11:  QS  $\leftarrow$  (Somatoria/numeroDePerguntasDaQuery)
12:  return QS
13: fim função

```

---

## 4. Experimento

Os experimentos realizados têm dois objetivos: (i) comparar o QSMatching com os modelos vetorial e fuzzy para recuperação de questionários; (ii) verificar a efetividade da similaridade proposta no QSMatching descrito em Seção 3. A seguir, a seção mostra

Tabela 1. Questionários coletados

Questionários	Pesquisa de satisfação	Saúde	Pesquisa de mercado	Planejamento de eventos	Comportamento humano	Pesquisa com empregados	Tecnologia	Esportes	Demografia	Outros
Quantidade	21	17	14	10	9	8	7	6	5	40

as características dos questionários armazenados e utilizados, uma breve descrição das métricas utilizadas para avaliação, a metodologia de avaliação dos experimentos e os resultados obtidos em conjunto com uma análise do comportamento do processo de busca apresentado nesta seção. O *groundtruth* usado neste experimento é o mesmo utilizado em [Souza and Dorneles 2018].

#### 4.1. Dataset e métricas de avaliação

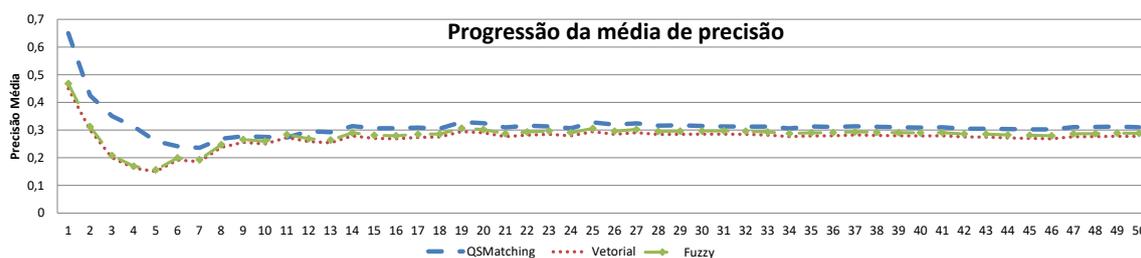
O banco de dados consiste em 137 questionários coletados na *Web* por meio de um *crawler*. A Tabela 1 mostra a quantidade de questionários coletados por domínio de pesquisa. Os questionários coletados contêm, em média, 16 perguntas, sendo que, o menor contém apenas 5 perguntas e o maior contém 57 perguntas. Todos os questionários juntos contêm 2.411 perguntas. Desse total, 365 são questões abertas e as 2.046 restantes são perguntas fechadas, o que significa que 84,86% das questões contêm alternativas.

As seguintes métricas clássicas para avaliar os resultados foram usadas: revocação, precisão, *f-value*, DCG (*Discounted Cumulated Gain*), MAP (*Mean Average Precision*) e MAE (*Mean Absolute Error*) [Baeza and Ribeiro-Neto 2011]. Essas métricas são úteis para comparar o desempenho do QSMatching em relação aos modelos vetorial e fuzzy em 50 consultas.

A quantidade de consultas está de acordo com uma regra geral que é frequentemente aplicada no campo de recuperação de informações [Buckley and Voorhees 2000]. Por essa regra, um conjunto de consultas de tamanho entre 25 e 50 é geralmente empregado na avaliação dos mecanismos de busca. A Figura 4 mostra o comportamento das consultas sobre o *dataset*. A Figura 4 representa a evolução da média da precisão de acordo com número de consultas realizadas, conforme ilustrado, os valores de precisão tendem a se estabilizar, neste caso específico, em 13 a 14 consultas.

#### 4.2. Metodologia

De maneira geral, os experimentos foram conduzidos em sete etapas. Na etapa 1, os questionários disponíveis na Internet foram coletados e armazenados no *dataset* (137 questionários foram coletados no total).



**Figura 4. Evolução da média da precisão**

Na etapa 2, os dados coletados foram analisados com o objetivo de elaborar os parâmetros de busca a serem utilizados nos experimentos, a fim de montar um *ground truth*. Foram selecionados 50 questionários dentre os 137 do dataset para serem usados como base para as consultas. No entanto, o uso de um questionário existente no *dataset* como uma consulta induz uma classificação cujo primeiro elemento é sempre relevante. Para minimizar esse efeito, optou-se por usar apenas parte do questionário, em alguns casos apenas uma palavra, ou frase, ou uma pergunta, ou algumas das perguntas presentes no questionário.

Na etapa 3, o *ground truth* é montado para formar uma base de comparação de modo a conter quais questionários devem ser retornados das consultas. Então, de forma manual, foram verificados e registrados quais questionários são relevantes e em que ordem eles devem ser apresentados para cada consulta elaborada na Etapa 2. Um grupo de três especialistas colaborou para construir o *ground truth*. Como resultado, obteve-se o ranking de questionários relevantes para cada consulta, construindo assim o *ground truth*. Por exemplo, a Tabela 2 contém um exemplo de ordenação do *ground truth*; a coluna “Relevância (Sim / Não)” indica se o questionário é relevante ou não, e a coluna “Relevância da Pontuação [0,2]” indica a escala usada para as métricas do DCG<sup>12</sup>. O *ground truth* é usado como base para comparar, por meio das métricas de avaliação, a eficácia das abordagens QSMatching, vetorial e fuzzy.

Na etapa 4, o modelo vetorial<sup>13</sup> foi utilizado para obter o ranking de questionários relevantes. As consultas foram selecionadas conforme descrito na Etapa 2.

Na etapa 5, o ranking de questionários relevantes foi obtido por meio da abordagem QSMatching. As mesmas consultas da etapa 4 foram usadas. O framework Lucene [luc 2018] foi usado para pré-processamento e tokenização, conforme descrito em Definição 2 da Seção 3.2. Então, o cálculo de similaridade é realizado conforme as Definições 2 até 8 do QSMatching descritas em Seção 3.2.

Na Etapa 6, o modelo fuzzy<sup>14</sup> foi utilizado para obter o ranking de questionários relevantes. As consultas foram as mesmas utilizadas nas etapas 4 e 5.

<sup>12</sup>Para aplicar a métrica de avaliação do DCG, no passo 6, as pessoas foram solicitadas a indicar uma escala de relevância entre 0 e 2, juntamente com o ranking, onde 2 indica que o questionário é de maior relevância, 1 indica que o questionário é relevante e 0 quando o questionário não é relevante.

<sup>13</sup>Lucene [luc 2018] foi usado, pois implementa o algoritmo do modelo vetorial.

<sup>14</sup>Lucene [luc 2018] foi usado, pois implementa o algoritmo do modelo fuzzy.

**Tabela 2. Exemplo de Ground Truth**

Query 1- Você viaja a negócios?  
2- Qual a sua renda ano passado?

Ranking	Questionário	Relevância (Sim /Não)	Pontuação [0,2] Relevância
1º	29	Sim	2
2º	70	Sim	2
3º	73	Sim	1
4º	72	Sim	1
5º	79	Sim	1
6º	07	Sim	1
-	1	Não	0
		...	
-	84	Não	0

Na etapa 7, com base no *ground truth*, os rankings gerados pelas abordagens QSMatching, vetorial e fuzzy foram comparados e avaliados. Esses resultados são mostrados na próxima Subseção.

### 4.3. Resultados

A eficácia da abordagem QSMatching foi analisada usando os resultados obtidos com os experimentos realizados. A Tabela 3 apresenta o número de consultas que obtiveram 100% de precisão para cada nível de revocação. Pode-se verificar que a abordagem QSMatching obteve 100% de efetividade no ranking do primeiro questionário relevante, enquanto o modelo vetorial obteve 82% de efetividade e o modelo fuzzy obteve 84% de efetividade. Nos primeiros 5 níveis de revocação, o QSMatching obteve uma precisão melhor do que os modelos vetorial e fuzzy em todas as consultas. A precisão do QSMatching é melhor que os modelos vetorial e fuzzy na média de 50 consultas, e o mesmo ocorre nas primeiras 25 consultas.

A Tabela 4 apresenta a evolução da precisão, revocação, f-value<sup>15</sup> e MAP para as 25 e 50 consultas realizadas usando o QSMatching, modelo vetorial e fuzzy. O resultado mostra que o QSMatching apresenta, em média, resultados superiores ao uso dos modelos vetorial e fuzzy ao lidar com a recuperação de questionários. Além disso, a revocação é melhor que a precisão em todos os modelos testados.

A Figura 5 mostra o gráfico gerado pelo cálculo do DCG (*Discounted Cumulated Gain*) comparando as três abordagens em relação ao *ground truth*. Ao comparar o QSMatching com o modelo vetorial, observa-se que o QSMatching apresentou um DCM maior em 82% das consultas realizadas, e em 2% o ganho foi o mesmo que no modelo vetorial. Nos restantes 16% das consultas, o modelo vetorial obteve um ganho acumulado melhor que o QSMatching, mas a diferença entre os modelos foi menor que 1 em 92%

<sup>15</sup>Considerando os 20 primeiros questionários retornados

**Tabela 3. Número de consultas com 100% de precisão nos 11 níveis padrões de revocação**

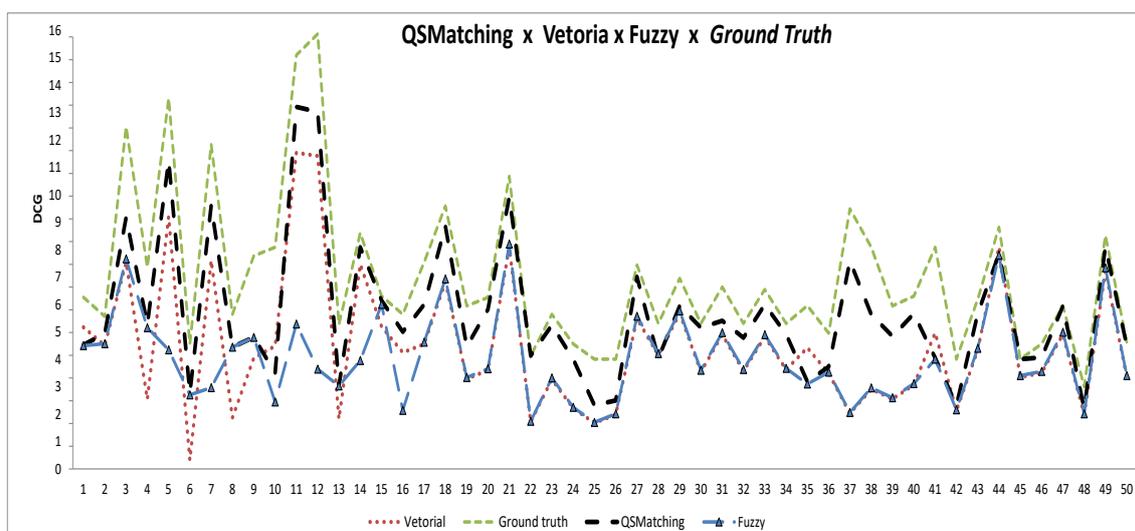
Revocação	Precisão	Número de consultas		
		QSMatching	Vetorial	Fuzzy
0	100%	50	41	42
0.1	100%	50	40	41
0.2	100%	44	37	39
0.3	100%	41	35	35
0.4	100%	35	33	34
0.5	100%	29	30	29
0.6	100%	22	21	22
0.7	100%	15	17	16
0.8	100%	9	13	10
0.9	100%	8	6	8
1	100%	3	4	4

**Tabela 4. Média da precisão, revocação, f-value e MAP**

		QSMatching	Vetorial	Fuzzy
25 Queries	Precisão	0,327	0,281	0,289
	Revocação	0,6801	0,6031	0,6149
	Medida-F	0,44558	0,3812	0,3889
	MAP	0,7335	0,6332	0,6453
	Tempo de execução	0,039	0,01	0,01
50 Queries	Precisão	0,299	0,207	0,218
	Revocação	0,6809	0,6213	0,6331
	Medida-F	0,4143	0,3561	0,3629
	MAP	0,662363	0,56407	0,57484
	Tempo de execução	0,031	0,01	0,01

dos casos em que o modelo vetorial foi superior. Isso se deve ao fato de que o modelo vetorial prediz um peso menor para termos que ocorrem com maior frequência no conjunto de textos selecionados, enquanto aumenta o peso daqueles que ocorrem raramente, por exemplo, se a consulta contiver a seguinte pergunta: “Com que frequência você voa?”. Os termos “Com”, “que”, “frequência” e “você” repetem em várias perguntas, e consequentemente, o modelo vetorial atribui um peso menor a esses termos e um peso maior ao mais importante termo da questão, que é “voar”. Ao comparar o modelo Fuzzy com o QSMatching, observa-se que a abordagem QSMatching obteve um CDG maior em 90% das consultas realizadas, enquanto nas demais consultas o desempenho foi igual. Ao analisar as consultas que obtiveram o mesmo desempenho, constatou-se que as “perguntas” similares entre a consulta e o questionário não possuíam alternativas.

Embora os modelos vetorial e fuzzy sejam mais rápidos na execução do que o QSMatching, o tempo de execução de 0,031 segundos do QSMatching é satisfatório se for considerado o ponto de vista de um usuário [Hoxmeier and DiCesare 2000].



**Figura 5. Comparação do ganho acumulado (DCG)**

Com base nos valores obtidos no DCG (Figura 5), o MAE (Mean absolute error) [Willmott and Matsuura 2005] para o QSMatching foi 1,2581, o modelo vetorial teve um MAE de 1,9356 e o modelo fuzzy teve um MAE de 1,899. QSMatching tem um menor erro absoluto médio do que os modelos vetorial e fuzzy quando comparado com o *ground truth*.

O experimento revelou que a abordagem QSMatching foi mais eficaz do que os modelos vetorial e fuzzy, especialmente quando os 5 primeiros questionários retornados foram analisados. No entanto, na abordagem do QSMatching houve uma variação entre os resultados obtidos e os esperados no ranking dos questionários que devem aparecer a partir da quarta posição. Essas perguntas possuem sinônimos para os termos da consulta, portanto não foram consideradas similares.

Apesar do bom desempenho, há pontos para melhoria no QSMatching. Desta forma, os seguintes trabalhos futuros são listados: (i) agregar a comparação de sinônimo de uma maneira similar às definições definições 2, 4, 7 e 8, de maneira a verificar se há um aumento da eficácia no QSMatching; (ii) usar abordagens conhecidas na comparação de strings para calcular a similaridade entre perguntas (por exemplo: a distância de Levenshtein [Yujian and Bo 2007] e jaccard [Ni wattanakul et al. 2013]); (iii) realizar um processo de tratamento dos termos que indiquem que a sentença é uma pergunta (por exemplo: cite, justifique, porque, ...), uma vez que a abordagem QSMatching considera a estrutura do questionário.

## 5. Trabalhos Relacionados

O desenvolvimento de sistemas para busca e ordenação de questionários de pesquisa ainda é um desafio [Souza and Dorneles 2018]. No entanto, é possível encontrar na literatura alguns trabalhos relacionados aos sistemas CQA [Lo et al. 2016, Chen et al. 2015, Hovy et al. 2000, Pôssas et al. 2005, Song et al. 2007, Grappy et al. 2011, Aslay et al. 2013, Dalip et al. 2013], que possuem características

que podem ser utilizadas na busca e ordenação de questionários e, portanto, podem ser analisados a partir dessa perspectiva. Por exemplo, trabalhos que analisam a qualidade da resposta a uma determinada pergunta pode ser visto de forma análogo quando a busca por questionários é realizada a partir de apenas uma palavra-chave ou de frases.

Entretanto, os trabalhos de CQA tratam de perguntas individuais (não são conjuntos de perguntas em um mesmo documento), além disso, 100% das perguntas são abertas, ou seja, não há perguntas de múltipla-escolha, por exemplo. De modo que não haveria forma de fazer uma comparação direta. Nesse sentido, procuramos trabalhos que fazem a recuperação por similaridade de documentos inteiros conforme descrito na Seção 2.

Além dos trabalhos em sistemas CQA, existem também trabalhos gerais de recuperação de informação, empregados para recuperar qualquer tipo de documento, como o trabalho de Possas *et al* [Pôssas et al. 2005]. Nesse sentido, existem na literatura várias formas de mensurar a similaridade textual. Alguns medem *strings* longas, como o TF-IDF. Outra maneira de medir é usando Frequência de Co-Ocorrência, que representa a frequência de ocorrência de dois termos seguidos em uma dada ordem [Anwar and Abulaish 2015]. Outros trabalhos usam *Distância do Google normalizada* (NGD) [Faulkner 2014, Cilibrasi and Vitanyi 2007], *métrica de similaridade com base na Wikipédia* (WLM) [Witten and Milne 2008] ou *Coefficiente de contingência* [Manning and Schütze 1999].

## 6. Conclusão

No presente trabalho, apresentamos um experimento para comparar as abordagens QSMatching, vetorial e fuzzy. O experimento mostrou um ganho significativo do QSMatching para pesquisar e calcular a similaridade em questionários de pesquisa. O experimento mostra que o QSMatching foi mais efetivo na recuperação de questionários do que os outros modelos (vetorial e fuzzy) na maioria das consultas, especialmente nos 4 primeiros níveis de revocação, onde apresentou superioridade em todas as consultas realizadas. Na média de precisão o QSMatching é melhor do que os outros modelos. No entanto, a partir do quinto ponto de revocação, os modelos testados obtiveram um desempenho semelhante. O experimento mostra que os modelos vetorial e fuzzy tem eficácia praticamente iguais ao recuperar questionários embora o modelo fuzzy tenha obtido resultados melhores nas métricas obtidas.

Embora os modelos vetorial e fuzzy sejam mais rápidos no tempo de execução do que a abordagem do QSMatching, o tempo de execução inferior a 1 segundo é satisfatório do ponto de vista do usuário.

Uma análise dos resultados indica que a proposta do QSMatching é promissora. Desta forma, os seguintes trabalhos futuros são listados: (i) agregar a comparação de sinônimo de uma maneira de maneira verificar se há um aumento da eficácia no QSMatching; (ii) usar abordagens conhecidas na comparação de *strings* para calcular a similaridade entre perguntas; (iii) realizar um processo de tratamento dos termos que indiquem que a sentença é uma pergunta (por exemplo: o que, por que, onde, quando...), uma vez que a abordagem QSMatching considera a estrutura do questionário.

## Referências

- (2018). Apache lucene. <https://lucene.apache.org/>. (Acessado: 01-2018).
- Anwar, T. and Abulaish, M. (2015). Ranking radically influential web forum users. *IEEE Transactions on Information Forensics and Security*, 10(6):1289–1298.
- Aslay, Ç., O’Hare, N., Aiello, L. M., and Jaimes, A. (2013). Competition-based networks for expert finding. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pages 1033–1036. ACM.
- Baeza, Y. and Ribeiro-Neto, B. (2011). Modern information retrieval-the concepts and technology behind search. pages 57–513. Pearson.
- Buckley, C. and Voorhees, E. M. (2000). Evaluating evaluation measure stability. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 33–40. ACM.
- Chen, R.-C., Spina, D., Croft, W. B., Sanderson, M., and Scholer, F. (2015). Harnessing semantics for answer sentence retrieval. In *Proceedings of the Eighth Workshop on Exploiting Semantic Annotations in Information Retrieval*, pages 21–27. ACM.
- Cilibrasi, R. L. and Vitanyi, P. M. (2007). The google similarity distance. *IEEE Transactions on knowledge and data engineering*, 19(3):370–383.
- Dalip, D. H., Gonçalves, M. A., Cristo, M., and Calado, P. (2013). Exploiting user feedback to learn to rank answers in q&a forums: a case study with stack overflow. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pages 543–552. ACM.
- Faulkner, A. R. (2014). *Automated classification of argument stance in student essays: A linguistically motivated approach with an application for supporting argument summarization*. PhD thesis. Copyright - Database copyright ProQuest LLC; ProQuest does not claim copyright in the individual underlying works; last update - 2016-06-05.
- Grappy, A., Grau, B., Falco, M.-H., Ligozat, A.-L., Robba, I., and Vilnat, A. (2011). Selecting answers to questions from web documents by a robust validation process. In *Web Intelligence and Intelligent Agent Technology (WI-IAT), 2011 IEEE/WIC/ACM International Conference on*, volume 1, pages 55–62. IEEE.
- Hovy, E. H., Gerber, L., Hermjakob, U., Junk, M., and Lin, C.-Y. (2000). Question answering in webclopedia. In *TREC*, volume 52, pages 53–56.
- Hoxmeier, J. A. and DiCesare, C. (2000). System response time and user satisfaction: An experimental study of browser-based applications. *AMCIS 2000 Proceedings*, page 347.
- Kowalski, G. J. and Maybury, M. T. (2000). Information storage and retrieval systems: theory and implementation. 8:333.
- Lakatos, E. M. and Marconi, M. d. A. (2010). Fundamentos da metodologia científica. In *Fundamentos da metodologia científica*, pages 19–292. Altas.
- Liu, B. (2007). Web data mining: exploring hyperlinks, contents, and usage data. page 643.

- Lo, S. L., Chiong, R., and Cornforth, D. (2016). Ranking of high-value social audiences on twitter. *Decision Support Systems*, 85:34–48.
- Manning, C., Raghavan, P., and Schütze, H. (2010). Introduction to information retrieval. *Natural Language Engineering*, 16(1):375.
- Manning, C. D. and Schütze, H. (1999). Foundations of statistical natural language processing. pages 1–373, Cambridge, MA, USA. MIT Press.
- Niwattanakul, S., Singthongchai, J., Naenudorn, E., and Wanapu, S. (2013). Using of jaccard coefficient for keywords similarity. In *Proceedings of the international multi-conference of engineers and computer scientists*, volume 1, pages 380–384.
- Patten, M. L. (2016). Questionnaire research: A practical guide. pages 1–145. Routledge.
- Picard, C. F. (1980). Graphs and questionnaires. volume 32, pages 10–380. Elsevier.
- Pôssas, B., Ziviani, N., Meira Jr, W., and Ribeiro-Neto, B. (2005). Set-based vector model: An efficient approach for correlation-based ranking. *ACM Transactions on Information Systems (TOIS)*, 23(4):397–429.
- Sheatsley, P. B., Rossi, P. H., Wright, J. D., and Anderson, A. B. (1983). *Questionnaire construction and item writing*. In Rossi, P.H., Wright, J.D. and Andersen, A.B. (eds) Handbook of Survey Research. New York: Academic Press, pp. 195-230.
- Song, W., Feng, M., Gu, N., and Wenyin, L. (2007). Question similarity calculation for faq answering. In *Semantics, Knowledge and Grid, Third International Conference on*, pages 298–301. IEEE.
- Souza, R. H. d. and Dorneles, C. F. (2017a). Analisando a eficácia do modelo vetorial de busca na ordenação de questionários. *XIII Simpósio Brasileiro de Sistemas de Informação*, pages 563–570.
- Souza, R. H. d. and Dorneles, C. F. (2017b). Qsmatching: an approach to calculate similarity between questionnaires. *The 19th International Conference on Information Integration and Web-based Applications & Services. Short paper*, pages 141–145.
- Souza, R. H. d. and Dorneles, C. F. (2018). Qsmatching vs vector model: comparing effectiveness in questionnaires retrieval. *XIV Simpósio Brasileiro de Sistemas de Informação*, pages 26–33.
- Vieira, S. (2009). Como elaborar questionários. pages 1–150. Atlas.
- Villar, L. M., Almeida, A. J. d., Lima, M. C. A. d., Almeida, J. L. V. d., Souza, L. F. B. d., and Paula, V. S. d. (2008). A percepção ambiental entre os habitantes da região noroeste do estado do rio de janeiro. *E. Anna Nery Revista Enfermagem*, 12(2):285–290.
- Waslawick, R. S. (2014). Metodologia de pesquisa para ciência da computação. pages 1–149, Rio de Janeiro. Elsevier.
- Willmott, C. J. and Matsuura, K. (2005). Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance. *Climate research*, 30(1):79–82.

- Witten, I. and Milne, D. (2008). An effective, low-cost measure of semantic relatedness obtained from wikipedia links. In *Proceeding of AAAI Workshop on Wikipedia and Artificial Intelligence: an Evolving Synergy*, AAAI Press, Chicago, USA, pages 25–30.
- Yujian, L. and Bo, L. (2007). A normalized levenshtein distance metric. *IEEE transactions on pattern analysis and machine intelligence*, 29(6):1091–1095.