Detecção e Avaliação de Cola em Provas Escolares Utilizando Mineração de Texto: um Estudo de Caso

Elmano Ramalho Cavalcanti

UFCG / DSC – Rua Aprígio Veloso, 882 – Bairro Universitário – 58.429-140 – Campina Grande – PB - Brasil
elmano@copin.ufcg.edu.br

Elmano Pontes Cavalcanti

UFCG / UAAC – Rua Aprígio Veloso, 882 – Bairro Universitário – 58.429-140 – Campina Grande – PB - Brasil
elmanopc@gmail.com

Carlos Eduardo Pires

UFCG / DSC – Rua Aprígio Veloso, 882 – Bairro Universitário – 58.429-140 – Campina Grande – PB - Brasil
cesp@dsc.ufcg.edu.br

Rodrigo Alves Costa

UFPE / CIn – Av. Jornalista Aníbal Fernandes, s/n – Cidade Universitária – 50.740-560 – Recife – PE - Brasil
rac2@cin.ufpe.br

Caroline Ramalho Cavalcanti

FSM – Rodovia BR, 230 – 58.900-000 – Cajazeiras PB – Brasil
carolineracavalcanti@yahoo.com.br

Resumo

A mineração de texto tem sido utilizada para diversos propósitos, como na extração, classificação e categorização de documentos. Dentro desse contexto, este trabalho apresenta um estudo de caso realizado na Universidade Federal de Campina Grande, em que se utilizou do modelo de vetor de espaços e da métrica de similaridade por cosseno na construção de um modelo classificador capaz de inferir o nível de cola existente entre provas escolares. A inferência do modelo foi comparada à detecção tradicional (humana) de cola e quantificada estatisticamente pelas seguintes métricas: acurácia, variação da acurácia, precisão, recall, F-measure e índice Kappa. Os resultados apontam que o modelo criado apresentou uma boa qualidade na detecção do nível de cola entre provas, tornando possível sua utilização em situações de ensino reais.

Palavras-Chave: Educação Mediada por Tecnologia, Mineração de Texto, Modelo Vetor de Espaço, Detecção de Cola, Avaliação de Cola, Classificação de documentos, UFCG

Abstract

Text mining has been used for various purposes, such as extraction, classification and categorization of documents. In this sense, this paper presents a case study accomplished at Federal University of Campina Grande that used the vector space model and the cosine similarity metric to build a classifier model able to infer the level of cheating between exams. The inference of the model was compared to traditional detection (human) of cheating and quantified statistically by the metrics accuracy, variation of the accuracy, precision, recall, F-measure, and Kappa index. The results point out that the model achieved a good quality in detecting the level of cheating between exams, making possible its use in real teaching situations.

Keywords: Technology-mediated Education, Text Mining, Vector Space Model, Cheating Detection, Cheating Evaluation, Documents classification, UFCG

1 Introdução

Em um contexto em que grande parte dos dados corporativos encontra-se disponível em forma textual, o processo de mineração de textos surgiu como uma poderosa ferramenta de apoio à gestão de conhecimento. Nesse sentido, o objetivo da prática de mineração de textos centra-se na busca por padrões, tendências e regularidades em documentos escritos em linguagem natural [12]. Extração e classificação de documentos, filtro de emails, máquinas de busca e categorização de documentos são alguns exemplos de aplicações de mineração de texto [8].

Embora as aplicações de mineração de texto sejam comumente utilizadas para fins industriais e comerciais, é apresentada neste artigo uma utilização prática no setor de educação, mais exatamente na detecção e classificação (avaliação) automática de cola em provas escolares presenciais.

É notório que cola de alunos em provas escolares é um problema antigo e, até então, de difícil solução [25]. Não há uma definição precisa de cola, mas supõe-se que haja cola toda vez que duas provas tenham um grau suspeito de semelhança. A dimensão de uma cola é variável: pode ser uma parte de uma questão, toda a questão, algumas questões, ou a prova inteira.

Alguns estudos exploratórios mostram a situação atual da prática da cola tanto em escolas de ensino médio [1] como em faculdades e universidades [2] brasileiras. Dependendo do tamanho da turma e também da prova aplicada, torna-se uma tarefa não trivial para o professor detectar cola em provas subjetivas.

Motivado por essas circunstâncias, apresenta-se neste trabalho um estudo de caso da aplicação dos métodos, técnicas e algoritmos de mineração de texto na construção de um detector de cola de provas escolares no contexto universitário.

Os conceitos sobre mineração de texto são apresentados na Seção 2. Na Seção 3 é apresentado o estudo de caso realizado na Universidade Federal de Campina Grande – PB, onde são descritas as etapas do processo de mineração que culmina com a criação de um detector de colas de provas escolares. A análise dos resultados é apresentada na Seção 4. A validação do modelo é apresentada na Seção 5, em que se compara o modelo detector de colas com a detecção de cola feita por um especialista humano. Por fim, as considerações finais, limitações e trabalhos futuros são listados na Seção 6.

2 Trabalhos Relacionados

Um problema pedagogicamente semelhante ao da cola em provas escolares é o plágio, sendo este mais co-

mun em trabalhos acadêmicos devido, principalmente, ao acesso à informação através da Internet. Na literatura encontram-se muitas pesquisas que tratam desse problema. Lukashenko et al. [26] apresentam um *survey* sobre métodos e ferramentas para detecção de plágio. Outros trabalhos mais recentes [27,28] propõem novas soluções para lidar com esse problema.

Atualmente existem tanto ferramentas proprietárias como de código livre para detecção de plágio em documentos. Dentre as ferramentas proprietárias, pode-se destacar duas: Ephorus¹ e Plagium². A primeira é uma ferramenta Web utilizada para prevenir e detectar plágios em trabalhos escolares. O usuário seleciona um documento que pode ser comparado com outros documentos armazenados no mesmo computador, ou com documentos disponíveis na Web. Os resultados são mostrados em um relatório contendo uma descrição resumida das semelhanças entre o documento informado e os documentos encontrados.

Plagium é uma ferramenta Web que funciona de modo similar a uma máquina de busca. A principal diferença é que, para realizar uma busca, o usuário pode informar um bloco de texto completo, ao invés de um número reduzido de palavras-chave. A ferramenta divide o bloco de texto em fragmentos que são comparados com documentos disponíveis na web. O resultado das comparações é um conjunto de links para documentos que são pontuados para determinar quais documentos se assemelham ao bloco de texto fornecido como entrada.

Por outro lado, uma ferramenta de detecção de plágio de código livre é a Sherlock [38,39], um programa voltado à detecção de similaridade entre documentos de texto. Para tal, são utilizadas assinaturas digitais, que são associadas a cada documento. A saída do programa indica as porcentagens de similaridade entre cada par do conjunto de documentos considerado.

Uma outra ferramenta aberta é a URKUND [40], que busca por similaridades entre um documento e outros disponíveis em três tipos de fontes: Internet, documentos impressos (ex: livros, revistas, etc.) e documentos outrora enviados pelos estudantes. O URKUND é um sistema Web transparente que oferece um serviço automatizado para detecção de plágio através de envio de documentos por e-mail. Além disso, o sistema suporta vários formatos de arquivos (.doc(x), .odt, .html, .pdf, entre outros). Entretanto, a ferramenta funciona apenas de forma online, não havendo opção de utilizá-la localmente em uma máquina.

Outras ferramentas de detecção de plágio também são apenas voltadas à Internet/Web, ou seja, verificar se de-

¹ <http://www.ephorus.pt/home>

² <http://www.plagium.com/>

terminado documento é similar a outras fontes que estejam disponíveis online. Essa realidade, entretanto, diverge da tarefa de detectar cola entre provas escolares presenciais, dado que, nesse caso, o plágio é apenas local.

O estudo de caso descrito neste trabalho é baseado em técnicas de mineração de dados bastante difundidas na literatura. Além disso, durante o desenvolvimento do estudo de caso foram usadas apenas ferramentas de código livre. Até onde se sabe este é o primeiro trabalho que demonstra como utilizar a tecnologia de mineração de texto no desenvolvimento de uma solução de detecção e avaliação de cola em provas escolares.

Em relação à prática da cola, estudos apontam que a mesma está presente em todo o mundo, em todos os segmentos da educação, de ensino fundamental até a pós-graduação [29,30]. Esforços têm sido feitos para encontrar maneiras de impedir os estudantes de conseguir colar [30, 31] ou mesmo prever quando um estudante provavelmente irá colar [32,33].

Além de técnicas de prevenção e previsão, também é possível usar programas de computador para detectar cola em provas ou trabalhos escolares. Nesse sentido, a maioria dos trabalhos propõem técnicas estatísticas para detectar cola em provas de múltipla escolha [34-37]. Entretanto, neste artigo foi demonstrado como algoritmos de mineração de texto podem ser utilizados para detecção e avaliação de cola em provas subjetivas. Novamente, a aplicação apresentada nesse artigo, até onde se sabe, é inédita.

3 Mineração de Texto

A mineração de texto é um ramo da mineração de dados. Esta última trata, em sua maioria, com dados estruturados, ou seja, aqueles que apresentem uma forma bem definida (e.g., planilha de dados) [13]. Por outro lado, a mineração de texto trabalha com dados não-estruturados [14]. Recuperação da informação, classificação supervisionada e não-supervisionada (i.e., *clustering*) de documentos estão entre as áreas de pesquisa da mineração de texto [8].

Uma das mais propagadas e utilizadas técnicas de mineração de texto é a modelagem vetor de espaço (do inglês, *Vector Space Modelling – VSM*), introduzida por Salton et al. [5]. Ela é um modelo algébrico utilizado para representar documentos textuais como vetores de termos. Uma das grandes vantagens desse método é a possibilidade de automatizar o processo de mineração. Essa técnica tem sido muito utilizada em sistemas de recuperação de informação (RI) [9].

3.1 Representação e Similaridade de Documentos

Devido à natureza não-estruturada de documentos textuais, uma atividade essencial na etapa de pré-processamento do processo de mineração de textos se refere à estruturação desses. Isso permite que os documentos sejam manipuláveis por algoritmos de extração de conhecimento [12].

Embora alguns algoritmos exijam informações mais sofisticadas, como os baseados em conhecimentos linguísticos, a grande maioria dos algoritmos de extração de padrões necessita apenas que os documentos estejam em um formato de planilha, no qual as linhas correspondam aos documentos e as colunas aos termos presentes na coleção [13]. Esse formato é denominado “bag of words”, na qual os termos são considerados independentes, formando um conjunto desordenado em que a ordem de ocorrência das palavras não importa. Uma forma de representar a *bag of words* é por meio de tabelas atributo-valor [9]. Um exemplo dessa representação pode ser observado na Tabela 1, na qual d_i corresponde ao i -ésimo documento, t_j representa o j -ésimo atributo (ou termo), a_{ij} é a medida que relaciona o i -ésimo documento com o j -ésimo atributo e y_i representa a classe (rótulo) do i -ésimo documento. A classe representa uma classificação do documento. Por exemplo, se a mineração de texto estiver sendo utilizada para classificar um conjunto de documentos científicos, as classes poderiam representar os tópicos ou categorias da área de um documento (e.g., medicina, engenharia, direito).

Dessa forma, cada documento pode ser representado como um vetor $\vec{d}_i = (\vec{a}_i, y_i)$, no qual $\vec{a}_i = (a_{i1}, a_{i2}, \dots, a_{iM})$. Segundo Salton e Buckley [4], o cálculo do valor de a_{ij} depende de três componentes:

- O componente do documento, relacionado à estatística de um determinado termo estar presente em um dado documento;
- O componente da coleção, relacionado à distribuição da presença do termo ao longo da coleção de documentos;
- O componente da normalização, o qual possibilita que documentos de diferentes tamanhos possam ser comparados segundo uma mesma escala.

Diversas medidas foram propostas para calcular os valores dos pesos a_{ij} [12]. Essas medidas podem ser classificadas em dois tipos: binárias e baseadas em frequência. Pesos binários indicam a ocorrência ou não de um termo em um determinado documento. Pesos baseados em frequência visam contabilizar o número de ocorrências de um determinado termo em um dado documen-

to, servindo como base para a extração de diversas medidas estatísticas na extração de padrões, sendo esses os mais comumente aplicados. Dentre os pesos baseados em medidas de frequência, pode-se distinguir duas classes [12]: pesos supervisionados, que dependem de dados com valor de classe conhecido, medindo a importância de um dado atributo na determinação do valor de classe; e pesos não-supervisionados, aplicáveis a dados não-rotulados.

	t1	t2	...	tj	...	tM	C
d1	a11	a12	...	a1j	...	a1M	y1
d2	a21	a22	...	a2j	...	a2M	y1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	y3
di	ai1	ai2	...	aij	...	aiM	y2
⋮	⋮	⋮	⋮	⋮	⋮	⋮	y2
dN	aN1	aN2	...	aNj	...	aNM	y3

Tabela 1: Representação de documentos por vetor.

Dentre os pesos supervisionados, pode-se citar medidas como a ConfWeight [5] e a Informação Mútua [6]. Como exemplos de pesos não-supervisionados, pode-se citar a TF (*term frequency*), a qual considera a frequência absoluta dos termos nos documentos [7]; a IDF (*inverse document frequency*) [5], que calcula a frequência inversa do termo, favorecendo termos que aparecem em poucos documentos da coleção; e a TF-IDF (Salton e Buckley [4]), que consiste em uma combinação das duas anteriores (TF e IDF).

Quando se pretende verificar se dois documentos são parecidos, é intuitivo inicialmente analisar os termos (palavras) que estão presentes em ambos os documentos. Mais do que isso, é preciso verificar o número de vezes que cada termo aparece em cada documento. A isso se chama frequência do termo (*term frequency* - TF). Entretanto, visto que alguns termos são comuns (e.g., artigos), utiliza-se o fator de frequência inversa de documento (*inverse document frequency* - IDF), de forma a ponderar a frequência dos termos. Dessa forma, termos comuns terão um peso menor que termos incomuns. Esse método, denominado TF-IDF (*term frequency – inverse document frequency*) e proposto por Salton e Buckley [4], é muito utilizado na área de Recuperação de Informação [5].

Formalmente, tem-se que o valor da frequência de um termo i que aparece em um documento d_j é dado por:

$$TF_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \quad (1)$$

em que $n_{i,j}$ é o número de ocorrências do termo i no documento j e o denominador é a soma do número de

No contexto de mineração de texto, a *recall* de uma classe é definida como a porcentagem de documentos

ocorrências de todos os termos no documento d_j . Seja $|D|$ a quantidade de documentos, tem-se que a fórmula da frequência inversa de documento (IDF) é dada por:

$$IDF_i = \log \frac{|M|}{|\{d : t_i \in d\}|} \quad (2)$$

em que $|\{d : t_i \in d\}|$ representa o total de documentos em que o termo t_i aparece. Dessa forma, o valor TF-IDF para um termo i em um documento j é dado por:

$$TFIDF_{i,j} = TF_{i,j} * IDF_i \quad (3)$$

O custo computacional do método TF-IDF é da ordem $O(|N||M|)$, em que $|N|$ é o quantidade de documentos e $|M|$ a quantidade de termos utilizados (vide Tabela 1).

A similaridade entre dois documentos $\vec{D}_1$ e $\vec{D}_2$ pode ser calculada pelo cosseno do ângulo formado entre seus vetores, ou seja,

$$Sim(\vec{D}_1, \vec{D}_2) = \cos(\theta) = \frac{\vec{D}_1 \cdot \vec{D}_2}{\|\vec{D}_1\| \|\vec{D}_2\|} \quad (4)$$

em que $\vec{D}_1 \cdot \vec{D}_2$ representa o produto escalar dos vetores enquanto que $\|\vec{D}_1\|$ e $\|\vec{D}_2\|$ representam os módulos dos vetores.

O valor da similaridade é sempre positivo, variando entre 0 (zero), nenhuma similaridade, a 1 (um), similaridade total. O método de similaridade por cosseno é considerado o padrão nas pesquisas de mineração de texto [8,9,12]. Alguns exemplos de bibliotecas de código livre contendo algoritmos de similaridade de texto são o Simmetrics [10] e o SecondString [11].

3.2 Métricas de Qualidade

Na literatura de mineração de dados, existem métricas que quantificam e qualificam os modelos preditivos (e.g., classificação supervisionada, regressão). Considere a Tabela 2 que mostra a quantidade de classificações corretas em oposição às classificações preditas para as classes “+” e “-” de um modelo binário. Essa tabela é denominada matriz de confusão, e através dela as métricas *Acurácia*, *Precisão*, *Recall* e *F-measure* são calculadas.

	+	-	Precisão	Recall	F-measure
+	T_p	F_p	$\frac{T_p}{T_p + F_p}$	$\frac{T_p}{T_p + F_n}$	$\frac{2T_p}{2T_p + F_p + F_n}$
-	F_n	T_n	$\frac{F_n}{F_n + T_n}$	$\frac{F_n}{T_p + F_n}$	$\frac{2F_n}{2F_n + T_p + T_n}$

T_p : Verdadeiro Positivo, F_p : Falso Positivo.

T_n : Verdadeiro Negativo, F_n : Falso Negativo.

Tabela 2: Exemplo de matriz de confusão binária.

classificados corretamente entre todos os documentos pertencentes à classe. A *precisão* é a porcentagem de

documentos classificados corretamente entre todos os documentos que foram considerados pelo modelo como pertencentes à classe [12]. A métrica *F-measure* combina as duas anteriores, sendo definida pela expressão:

$$F\text{-measure} = \frac{2}{1/\text{Precisão} + 1/\text{Recall}} \quad (5)$$

Enquanto as métricas anteriores são calculadas para cada classe do modelo, a *Acurácia* é uma métrica global do modelo. Ela reflete a taxa de acerto, ou seja, o número de classificações que o modelo inferiu corretamente. Considerando o exemplo da Tabela 1, tem-se que:

$$\text{Acurácia} = (T_p + T_n) / (T_p + F_p + F_n + T_n) \quad (6)$$

Além dessas métricas, existe um coeficiente estatístico denominado índice *Kappa* ou Estatística K, que é uma medida de concordância em escalas nominais, muito utilizado em estudos que envolvam diagnóstico médico [6]. Aplicado ao contexto de classificação em mineração de texto, o índice *Kappa* indica o nível de concordância entre a classificação do modelo e a classificação de referência, ou seja, o quão os dois estão de acordo quanto à classificação. Como exemplo, considere a Figura 1, que ilustra uma matriz de confusão genérica de um problema de n casos (documentos).

		Classificação				
		1	2	...	c	Total
V e r d a d e	1	x_{11}	x_{12}	...	x_{1c}	x_{1+}
	2	x_{21}	x_{22}	...	x_{2c}	x_{2+}
	:	:	:	...	:	:
	c	x_{c1}	x_{c2}	...	x_{cc}	x_{c+}
	Total	x_{+1}	x_{+2}	...	x_{+c}	n

Figura 1: Matriz de confusão genérica.

O índice Kappa é calculado de acordo com a Equação 7. Assim como a métrica de similaridade por cosseno, $\hat{k}$ também varia entre zero (mínimo) e um (máximo).

$$\hat{k} = \frac{\theta_1 - \theta_2}{1 - \theta_2} \quad (7)$$

Para que se possa estimar a significância estatística e o intervalo de confiança de $\hat{k}$, é preciso obter sua variância, que é dada pela Equação 8 [19]. Para um número de amostras suficientemente grande ($n > 30$), tem-se que a $(\hat{k} - k) / \sqrt{\text{Var}(\hat{k})}$ segue uma distribuição Normal padrão (i.e., $N(0,1)$).

$$\text{Var}(\hat{k}) = \frac{1}{n} \left[\frac{\theta_1(1-\theta_1)}{(1-\theta_2)^2} + \frac{2(1-\theta_1)(2\theta_1\theta_2-\theta_3)}{(1-\theta_2)^3} + \right.$$

$$\left. \frac{(1-\theta_1)^2(\theta_4-4\theta_2^2)}{(1-\theta_2)^4} \right] \quad (8)$$

Os valores de θ_1 , θ_2 , θ_3 e θ_4 são calculados de acordo com os valores apresentados na Figura 1 e as equações de 8 a 11.

$$\theta_1 = \frac{\sum_{i=1}^c x_{ii}}{n} \quad (8)$$

$$\theta_2 = \frac{\sum_{i=1}^c x_{i+} x_{+i}}{n^2} \quad (9)$$

$$\theta_3 = \frac{\sum_{i=1}^c x_{ii} (x_{i+} + x_{+i})}{n^2} \quad (10)$$

$$\theta_4 = \frac{\sum_{i=1}^c \sum_{j=1}^c x_{ij} (x_{j+} + x_{+i})}{n^3} \quad (11)$$

Utilizando essas métricas de qualidade é possível avaliar adequadamente o modelo classificador de cola em provas apresentado na próxima seção.

4 Estudo de Caso

Realizou-se um estudo de caso no intuito de analisar até que ponto a metodologia e técnicas de mineração de texto podem ser satisfatoriamente aplicadas na detecção de cola em provas escolares. O estudo de caso foi efetuado na Universidade Federal de Campina Grande, em um projeto envolvendo as unidades acadêmicas de Administração e Ciência da Computação. Uma versão preliminar e resumida deste trabalho pode ser verificada em [22].

As etapas aplicadas neste estudo de caso seguem a metodologia do processo de mineração de dados apresentada por Tan et al. [13], a qual é composta pelas seguintes etapas: seleção, pré-processamento, transformação, garimpagem, análise e assimilação (vide Figura 2)

4.1 Seleção, Limpeza e Amostragem dos Dados

Para compor o estudo de caso, foram selecionadas 30 provas, contendo cada uma quatro questões subjetivas na área de administração e subárea marketing. As questões estão apresentadas no Apêndice deste trabalho.

As provas foram respondidas em formato eletrônico e armazenadas em formato de texto plano (e.g., .txt), não sendo necessário aplicar nenhuma atividade de limpeza nos dados.

A amostragem é mais comum em operações que envolvam grande quantidade de dados armazenados, por exemplo, em um banco de dados. Como isso não se aplica a este trabalho, não foi realizada amostragem, mas foi considerado todo o universo das 30 provas no estudo.

Figura 2: Processo de mineração de dados (traduzido e adaptado de Tan et al.[15])

Visto que na vida real um professor detecta cola ao comparar a resposta de uma questão pelo aluno A com a resposta da mesma questão pelo aluno B, a prova de cada aluno foi fragmentada em quatro partes, que foi o número de questões da prova. O texto de cada questão passou a ser considerado como o documento da mineração de texto.

Foram criados automaticamente dicionários para cada questão da prova, ou seja, um conjunto de palavras específicas que faz parte do contexto de cada questão. Além disso, quando se fez necessário, o professor acrescentou algumas palavras referentes ao contexto de cada questão. Dessa forma, sempre que duas respostas de alunos apresentarem muitas palavras idênticas desse dicionário, será considerado, com forte indício, a existência de cola. Os dicionários foram especificados em documentos de texto convencionais, com cada linha contendo uma palavra.

4.2 Transformação de Dados

Em seguida, como requisito para o correto funcionamento dos algoritmos de mineração, foram removidas a pontuação e acentuação das respostas de cada questão. Isso é necessário tanto por diminuir o tamanho dos vetores de documentos (vide Tabela 1), como por evitar que o algoritmo de mineração diferencie palavras que na realidade são as mesmas (e.g., ‘elétrico’ x ‘elettrico’). Embora essa remoção ajude na maioria dos casos, pode igualar palavras diferentes, no caso de palavras que se diferenciam justamente pelo acento (e.g., pelo/pélo/pêlo).¹ Pelo fato desses casos serem incomuns, acredita-se que essa etapa traga mais benefícios do que prejuízos. A implementação desta etapa foi realizada utilizando a API Java 1.5 e a versão 3.3.1 da IDE Eclipse [14].

Após a remoção da pontuação e acentuação, iniciou-se o processo de *tokenização*, que separou cada documento em palavras ou *tokens*. Além disso, foram filtrados os *tokens* de tamanho menor ou igual a três caracteres, removendo assim elementos gramaticais comuns, tais como preposições, artigos e conjunções, diminuindo o tamanho dos vetores e otimizando o algoritmo de mineração.

A etapa seguinte consistiu em remover as palavras irrelevantes (*stopwords*). Para isso, foi utilizada uma adaptação do dicionário de palavras irrelevantes da língua

portuguesa do projeto Snowball [15]. A adaptação foi feita acrescentando-se termos específicos e comuns ao domínio de cada questão da prova.

Em seguida, realizou-se a normalização morfológica (*stemming*), que consiste em transformar as palavras em seus termos primitivos. A Figura 3 ilustra essa etapa através de um fluxograma. As palavras que estão no plural, superlativo, ou que apresentem prefixo e sufixo são normalizadas, ou seja, reduzidas ao seu radical.

Para exemplificar o processo da normalização morfológica, considere que na resposta a uma questão o aluno A escreve a frase “...processa-se o produto para então...”, enquanto que o aluno B consegue olhar a prova de A e escreve: “...o produto é processado para então...”. Vemos que as palavras ‘processa-se’ e ‘processado’ possuem o mesmo radical ‘process’. A etapa de normalização morfológica serve para garantir a remoção dos caracteres referentes ao plural, feminino, aumentativo, diminutivo, etc., mantendo-se apenas o radical da palavra. A execução desta etapa foi realizada utilizando-se o algoritmo de *stemming* do projeto Snowball [15].

Após normalizar morfológicamente os documentos, é também importante normalizá-los semanticamente. Isso é feito através de um mapeamento de todos os sinônimos de uma palavra para uma única palavra-base. Para se conseguir isso, é preciso utilizar uma base léxica da língua em que o documento foi escrito. No caso da língua inglesa, existe a WordNet [16]. Já para a língua portuguesa existe a WordNet.PT [17]. Entretanto, não foi possível realizar a normalização semântica visto que a WordNet.PT encontrava-se em fase de manutenção no período de execução deste trabalho de pesquisa.

Finalmente, após essas etapas, cada questão de cada aluno (documento) é transformada em um vetor de palavras (Seção 2.1), de acordo com o método TF-IDF (Seção 2.2). Como a dimensão das respostas das questões era pequena (poucos parágrafos) não houve necessidade de utilizar técnicas de compressão de vetores. O tamanho do vetor ficou na faixa de 500 colunas. Não foi necessária a utilização de *pruning* (podagem), visto que seu uso piorou os resultados do cálculo de similaridade.

Todas as tarefas de transformação de dados foram realizadas utilizando a ferramenta RapidMiner [18], software de código aberto para descoberta de conheci

¹ Com o Novo Acordo Ortográfico da língua portuguesa [23], algumas palavras que antes eram diferenciadas pelo acento são agora escritas da mesma forma (ex: pelo/pélo/pêlo serão todas escritas sem acento).

Figura 3: Normalização morfológica (adaptado de Moraes [21]).

mento, aprendizagem de máquina e mineração de dados. A Figura 4 ilustra a cadeia de passos da etapa de transformação, que inicia-se no operador *ToLowerCaseConverter* e termina no *SnowballStemmer*.

O texto de cada questão é inicialmente transformado em letras minúsculas. Em seguida, realiza-se a *tokenização*. Depois, são retiradas as palavras de menos de 3 caracteres e as que são consideradas irrelevantes (*stopwords*). Por fim, realiza-se a normalização morfológica. Após a etapa de transformação, tem-se a garimpagem, a qual está destacada na mesma figura pelo operador *ExampleSet2Similarity*.

4.3 Garimpagem

A garimpagem (mineração) foi realizada através do modelo de vetor de espaços e da técnica de agrupamento por similaridade por função cosseno (Seção 2.3), a qual retorna um valor real no intervalo [0,1] para cada par de provas, em que 0 (zero) significa ausência total de cola e 1 (um) provas idênticas. A justificativa para a escolha do modelo vetor de espaços e da métrica de similaridade por cosseno deu-se pelo fato deles serem considerados, respectivamente, o modelo e a métrica padrões utilizados em aplicações de mineração de texto [8,9].

Visto que foram utilizadas trinta provas com quatro questões cada, o algoritmo de similaridade por cosseno foi executado $4 \times \binom{30}{2} = 4 \times 435 = 1740$ vezes. Visto que o tamanho dos vetores foi reduzido pela remoção de palavras pequenas (com menos de três caracteres) e das palavras contidas na lista de palavras irrelevantes, o tempo de execução total do algoritmo foi considerado pequeno, da ordem de poucos segundos.

Figura 4: Etapas do processo de mineração realizadas no RapidMiner.

5 Análise dos Resultados

Como detalhado na Seção 3, foram utilizadas 30 amostras de provas, entre as quais algumas contêm colas de tamanhos diferentes: pequena, razoável e grande. Lembrando que $\text{Sim}(\vec{D}_x, \vec{D}_y) \in [0,1]$, em que $\vec{D}_x$ e $\vec{D}_y$ representam as respostas de uma certa questão Z da prova pelos alunos X e Y , o modelo utilizará as seguintes regras para determinar qual a classe associada à tupla $\langle \text{Questão } Z, \text{Aluno } X, \text{Aluno } Y \rangle$:

- $\text{Sim}(\vec{D}_x, \vec{D}_y) \geq 0,70 \rightarrow$ Cola grande;
- $0,40 \leq \text{Sim}(\vec{D}_x, \vec{D}_y) < 0,70 \rightarrow$ Cola razoável;
- $0,20 \leq \text{Sim}(\vec{D}_x, \vec{D}_y) < 0,40 \rightarrow$ Cola pequena;
- $\text{Sim}(\vec{D}_x, \vec{D}_y) < 0,20 \rightarrow$ Ausência de cola. Considera-se que a similaridade ocorreu pelo acaso.

A Figura 5 apresenta os histogramas percentuais da similaridade, $\text{Sim}(\vec{D}_x, \vec{D}_y)$, entre todos os pares de resposta que possuíram uma similaridade acima de 10%¹.

Figura 5: Histogramas da similaridade entre as questões da prova.

Percebe-se uma distribuição de similaridade semelhante entre as quatro questões: a maior parte das questões apresentou baixo nível de semelhança com as demais ($< 0,20$), enquanto que um pequeno grupo apresentou níveis variados de semelhança, constituindo as colas pequenas, razoáveis e grandes. Isso mostra que em todas as questões da prova houve indícios da existência de cola de diversas dimensões.

Seja o grafo $G = \langle V, A \rangle$ em que V é o conjunto de provas (código do aluno) e A o conjunto de arestas que ligam as questões cuja similaridade é maior que certo limiar θ . Dessa forma, para cada valor de θ existe um único grafo de similaridade. Ajustando-o apropriadamente, é possível obter os grafos de similaridade para cada nível de cola.

A Figura 6 ilustra a visão do grafo circular mostrando a incidência das colas grandes e razoáveis ocorridas na primeira questão da prova. Os vértices do grafo representam os códigos atribuídos a cada aluno; as arestas representam o grau de similaridade entre as respostas a uma mesma questão feita por dois alunos. A posição dos vérti-

ces no grafo foi gerada aleatoriamente pela ferramenta, não tendo nenhuma relação à posição dos alunos na sala de aula. Os grafos circulares de similaridade das outras questões da prova estão ilustrados nas Figuras 7 a 9.

Um outro tipo de visualização das similaridades entre as questões da prova está ilustrado na Figura 10, onde as questões mais similares ficam mais próximas uma das outras², o que facilita ao professor descobrir rapidamente os grupos de alunos nos quais possivelmente houve cola. Na figura, percebe-se a ocorrência de oito incidências de possível cola, sendo que em dois casos houve a participação de um terceiro aluno: um grupo formado pelos alunos 5,21,28 e outro formado pelos alunos 2,27,29.

Figura 6: Grafo de similaridade para a questão 1

6 Avaliação do Modelo

Para validar o modelo de detecção e avaliação de cola foram utilizadas as correções reais das provas feita pelo professor da disciplina, de maneira tradicional. Após a correção, foi gerado um mapeamento das colas detectadas, o qual encontra-se apresentado na Tabela 3. Essa tabela mostra os pares de alunos em que foram detectados a cola e a dimensão da mesma. O professor foi instruído a classificar as colas nas mesmas três classes usadas pelo modelo de mineração, ou seja, colas de tamanho pequeno, razoável e grande. Somente após a correção das provas e geração do mapeamento de colas o professor teve acesso às avaliações feitas pelo modelo.

A partir do mapeamento de cola definido pelo profes-

¹ Omitiram-se os documentos em que a similaridade é menor que 10% para que as ocorrências sejam perceptíveis.

² O grafo foi desenhado de acordo com o método de Peter Eades para grafos não-direcionados [26].

sor pôde-se avaliar o desempenho do modelo de inferência, verificando se este detectou não apenas a existência de cola, mas se acertou também a dimensão da mesma.

Os percentuais de cola detectados pelo modelo de in-

ferência e pelo professor estão ilustrados nas Figuras 8 e 9. Percebe-se uma boa proximidade nas porcentagens dos tipos de cola *grande* e *razoável*, enquanto que uma razoável diferença na porcentagem de provas com *nenhuma*

Figura 7: Agrupamento das provas quanto à similaridade na resposta da 4ª questão.

Aluno	P	R	G
25	Q2-A2		Q1-A1
26		Q1-A1	Q2-A5 e Q3, 4-A7
27	Q1-A1	Q4-A2	Q2-A3
28	Q4-A5		
29		Todas-A2	
30			Todas-A1

* P,R,G: cola Pequena, Razoável e Grande.

** QX-AY: significa que o aluno colou a questão X do aluno Y.

Tabela 3: Mapeamento das colas feito pelo professor.*

cola. Uma diferença maior ocorreu nos valores de cola *pequena*.

Um exemplo em que o sistema detectou um cola grande é descrito na Tabela 4, onde encontram-se as respostas de dois alunos (códigos 1 e 30) em relação à terceira questão da prova. O modelo quantificou a similaridade entre essas respostas em 0,788, rotulando-a como uma grande cola. Ao realizar a correção, o professor também considerou que houve uma cola grande entre essas respostas.

O conjunto de previsão, utilizado para avaliar o modelo de detecção de cola, está descrito na Tabela 6. A primeira coluna especifica o número da questão da prova; as duas colunas seguintes indicam o código dos alunos envolvidos; a quarta contém a similaridade detectada pelo modelo; a quinta e sexta colunas contém, respectivamente, o tipo de cola inferido pelo modelo e o tipo verdadeiro (definido pelo professor); e a última coluna indica se o algoritmo acertou ou não em sua inferência. A tabela está em ordem crescente do número de questão da prova e em ordem decrescente de grau de similaridade.

Conforme pode ser verificado na Tabela 6, dos dez alunos taxados com alto índice de cola pelo modelo, nove deles também tiveram um alto índice de cola de acordo com o professor. A concordância entre a classificação de cola feita pelo modelo e pelo professor está apresentada na matriz de confusão (vide Tabela 4).

De acordo com a matriz de confusão, a acurácia do modelo foi de $35/44 = 79,55\%$. Os valores das métricas precisão, recall e *F-measure* para cada tipo de cola estão presentes na Tabela 5. Segundo a métrica harmônica *F-Measure*, o modelo apresentou excelente índice de acerto de cola dos tipos *grande* e *nenhuma*, um bom índice na

detecção de cola *pequena*, e um índice razoável na detecção de cola de tamanho *razoável*. Ressalta-se também, que o modelo não cometeu nenhum erro comprometedor, como confundir uma cola pequena com uma cola grande ou vice-versa.

“O marketing procura entender as necessidades dos consumidores e traduzir esse conhecimento para poder elaborar adequadamente os seus produtos e serviços. Considerando nesse caso, as características de todos os elementos do mix de produto. Por outro lado, o marketing também deve considerar a “arte” no que diz respeito ao lado do comportamento humano, no qual o consumidor toma decisões baseado em emoções e muitas vezes não tão necessariamente racional como talvez pudesse ser.”

“Os profissionais de marketing procuram entender as necessidades das pessoas e interpretar esse conhecimento para só assim poder elaborar os seus produtos e serviços. Considera-se para isso, as características de todos os elementos do mix de produto. O marketing também pode ser considerado como “arte” quando se considera o lado do comportamento humano, no qual o consumidor toma decisões baseado em emoções e muitas vezes não tão necessariamente racional como talvez pudesse ser.”

Tabela 4: Indício de cola entre duas respostas da 3ª questão da prova.

Figura 8: Percentuais de cola segundo o professor.

Figura 9: Percentuais de cola segundo o modelo.

	G	R	P	N	Total
G	9	1	0	0	10
R	2	4	0	0	6
P	0	1	6	4	11
N	0	1	0	16	17
Total	11	7	6	20	44

G: grande, R: razoável, P: pequena, N: nenhuma.

Tabela 5: Matriz de confusão do modelo classificador de cola.

Classe	Precisão	Recall	F-Measure
Cola grande	0,90	0,82	0,86
Cola razoável	0,67	0,57	0,62
Cola pequena	0,55	1,00	0,71
Nenhuma cola	0,94	0,80	0,86
Acurácia = 79,55%			

Tabela 6: Métricas de qualidade do modelo.

O índice Kappa, $\hat{k}$, foi de 0,7126, enquanto que o desvio padrão de $\hat{k}$ foi de 0,0831, ambos calculados de acordo com o método de Fleiss et al. [19]. Com 95% de confiança, tem-se que o intervalo de confiança do valor populacional de $\hat{k}$ é $0,5496 \leq \hat{k} \leq 0,8756$. De acordo com a interpretação desse índice discutida em [20], tem-se que ele está situado na categoria *substantial agreement* (que vai de 0,6 a 0,8), a segunda melhor possível, o que sugere uma boa qualidade de inferência na detecção de colas.

Uma justificativa para alguns dos erros de inferência apresentados pode ser a ausência do uso de algoritmos de mapeamento de expressões sinônimas (e.g., “com certeza” = “sem dúvida”), de forma que provas similares semanticamente e não lexicamente não puderam ser plenamente captadas pelo processo de mineração utilizado. Salienta-se também que as questões subjetivas da prova (vide Apêndice) não pediam respostas memorizadas, mas de caráter totalmente subjetivo, evitando assim a ocorrência

cia de falso-positivos.

7 Considerações Finais

Este artigo apresenta um estudo de caso da utilização da metodologia e técnicas de mineração de texto no desenvolvimento de um modelo detector e avaliador de cola em provas escolares. Na implementação da solução foi utilizado o tradicional modelo de vetor de espaços e da

métrica de similaridade por cosseno. Utilizou-se o software RapidMiner [18] nas etapas de transformação e garimpagem dos dados. Foi elaborado um conjunto predição para avaliar a acurácia do modelo detector de colas. O resultado indicou uma acurácia de quase 80% e índice Kappa de 0,71, o que indica um bom resultado [19].

Questão	Aluno A	Aluno B	Similaridade	Modelo	Professor	Acertou?
1	01	25	0,812	G	G	sim
1	01	30	0,770	G	G	sim
1	02	29	0,721	G	R	não
1	25	30	0,687	R	G	não
1	01	27	0,357	P	P	sim
1	25	27	0,337	P	P	sim
1	13	08	0,286	P	N	não
1	27	30	0,258	P	P	sim
1	01	26	0,216	N	R	não
1	26	30	0,196	N	N	sim
1	11	17	0,191	N	N	sim
1	22	6	0,166	N	N	sim
1	25	26	0,158	N	N	sim
1	13	28	0,152	N	N	sim
2	26	05	0,725	G	G	sim
2	27	03	0,714	G	G	sim
2	01	30	0,710	G	G	sim
2	02	29	0,511	R	R	sim
2	02	25	0,286	P	P	sim
2	25	29	0,279	P	P	sim
2	10	12	0,203	N	N	sim
2	11	18	0,173	N	N	sim
2	11	25	0,171	N	N	sim
2	11	29	0,154	N	N	sim
3	25	26	0,821	G	G	sim
3	01	30	0,788	G	G	sim
3	25	07	0,712	G	G	sim
3	26	07	0,674	R	G	não
3	02	29	0,554	R	R	sim
3	27	05	0,230	P	N	não
3	19	22	0,216	P	N	não
3	22	30	0,184	N	N	sim
3	11	15	0,168	N	N	sim
3	1	22	0,165	N	N	sim
3	14	18	0,156	N	N	sim
4	26	07	0,856	G	G	sim
4	02	29	0,497	R	R	sim
4	27	29	0,404	R	R	sim
4	02	27	0,323	P	R	não
4	28	05	0,290	P	P	sim
4	21	05	0,220	P	N	não
4	13	20	0,181	N	N	sim
4	08	09	0,152	N	N	sim
4	17	03	0,150	N	N	sim

G: cola grande, R: cola razoável, P: cola pequena, N: nenhuma cola.

Tabela 7: Conjunto de previsão utilizado para avaliação do modelo de detecção de cola.

A solução apresentada neste artigo pode auxiliar ou mesmo substituir o professor na difícil e às vezes árdua tarefa de detecção e avaliação da cola em provas escolares.

Como trabalhos futuros pretende-se: experimentar o modelo em provas de outras áreas do conhecimento;

utilizar o mapeamento de expressões semânticas; variar os limiares que definem a dimensão de cola. Além disso, pretende-se considerar a distribuição dos alunos em sala de aula como entrada para o modelo de inferência e classificação de colas desenvolvido.

Referências

- [1] J. C. X. Silva, C. E. Leal, L. P. Brandão, S. M. Lanes, L. F. Barbosa, L. F. Santos, M. B. Corrêa, P. R. Pessanha, S. R. de Azeredo, T. Fejolo, W. J. Silva, A. Alves. O Uso da Cola como Fator que Prejudica a Relação Ensino-Aprendizagem. In *XVIII Simpósio Nacional de Ensino de Física*, Vitória, Espírito Santo, 2009.
- [2] G. A. da Silva, M. M. da Rocha, E. O., Y. L. Pereira, and V. S. R. Bussab. Um estudo sobre a prática da cola entre universitários. *Psicol. Refl. Crít.*, 19(1):18–24, 2006.
- [3] G. Salton, A. Wong, and A. C. S. Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18:229–237, 1975.
- [4] G. Salton, C. Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523, 1988.
- [5] P. Soucy, G. W. Mineau. Beyond tfidf weighting for text categorization in the vector space model. In *IJCAI '05: Proceedings of the XIX International Joint Conferences on Artificial Intelligence*, páginas 1130–1135, 2005.
- [6] Y. Y. Yao. Information-theoretic measures for knowledge discovery and data mining. Em *Entropy Measures, Maximum Entropy Principle and Emerging Applications*, páginas 115–136, 2003.
- [7] C. J. V. Rijsbergen. *Information Retrieval*. Butterworths, 1979.
- [8] M. W. Berry. *Survey of Text Mining – Clustering, Classification and Retrieval*, Springer, New York, 2003.
- [9] S. Weiss, N. Indurkha, T. Zhang, F. Damerau. *Text Mining – Predictive Methods for Analyzing Unstructured Information*. Springer, New York, 2005.
- [10] S. Chapman. Simmetrics: a java & c# .net library of similarity metrics. Disponível em: <http://sourceforge.net/projects/simmetrics/>, Dez. 2010.
- [11] W. W. Cohen, P. Ravikumar, and S. Fienberg. Secondstring: Open source java-based package of approximate string matching. <http://secondstring.sourceforge.net/>, Dez. 2010.
- [12] R. Feldman, J. Sanger. *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*. Cambridge University Press, 2007.
- [13] P.-N. Tan, M. Steinbach, V. Kumar. *Introduction to Data Mining*, 1st edition. Addison-Wesley, 2006.
- [14] Eclipse IDE. <http://www.eclipse.org>, Agosto 2010.
- [15] M. F. Porter and R. Boulton. Snowball: A language for stemming algorithms. <http://snowball.tartarus.org/>, 2002.
- [16] G. A. Miller. Wordnet: A lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.
- [17] M. Palmira, R. Amaro, R. P. Chaves, S. Lourosa, C. Martins, S. Mendes. Rede léxico-conceitual do português. <http://www.clul.ul.pt/clg/wordnetpt>, Nov. 2010.
- [18] Rapidminer - Open-source data mining with the java software rapidminer. <http://rapidi.com/>, Nov. 2010.
- [19] J. L. Fleiss, B. Levin, M. C. Paik, and J. Fleiss. *Statistical Methods for Rates and Proportions*. Wiley-Interscience, New York, 2003.
- [20] J. R. Landis and G. G. Koch. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174, March 1977.
- [21] E. A. M. Morais, Contextualização de Documentos em Domínios Representados por Ontologias Utilizando Mineração de Textos. Dissertação de Mestrado, Instituto de Informática - Universidade Federal de Goiânia, 2007.
- [22] E. R. Cavalcanti, J. S. Jackson. Aplicando classificação não-supervisionada para detecção de cola em provas escolares. In *Workshop Franco-Brasileiro sobre Mineração de Dados (WFB2009)*, Recife, Brasil, 2009.
- [23] A. G. Cunha, *Vocabulário Ortográfico da Língua Portuguesa*, Lexikon, 2ª edição, 2009.
- [24] P. Eades, C. Gutwenger, S-H Hong, P. Mutzel, *Graph Drawing Algorithms. Algorithms and theory of computation handbook: special topics and techniques*, 2ª edição, CRC Press, 2010.
- [25] M. Rangel. O problema da cola sob a ótica das representações. *Revista Brasileira de Estudos Pedagógicos*, 82(200/201/202):78–88, 2001.
- [26] R. Lukashenko, V. Gaudina, J. Grundspenkis. Computer-Based Plagiarism Detection Methods and Tools: An Overview. In *International Conference on Computer Systems and Technologies*,

- Rousse, Bulgaria, June 14-15, vol. 285, p. 40, ACM, 2007.
- [27] A. Barrón-Cedeño, P. Rosso. On Automatic Plagiarism Detection Based on n-Grams Comparison, *Lecture Notes in Computer Science*, Springer, 5478:686–700, 2009.
- [28] S. Butakov, V. Scherbinin. The toolbox for local and global plagiarism detection. *Computers & Education* 52 (4), 781–788.
- [29] S. F. Davis, P. F. Drinan, T. B. Gallant. *Cheating in school: What we know and what we can do*, 1st edition, Wiley-Blackwell, 2009.
- [30] C. Guthrie. *Plagiarism and cheating: A mixed methods study of student academic dishonesty*. Ph.D. thesis, University of Waikato, 2009.
- [31] M. A. Broeckelman-Post. Faculty and student classroom influences on academic dishonesty. *Informatics in Education* 51 (2), 206–211.
- [32] H. J. Passow, M. J. Mayhew, C. J. Finelli, T. S. Harding, D. D. Carpenter. Factors influencing engineering students' decisions to cheat by type of assessment. *Research in Higher Education* 47 (6):643–684, 2006.
- [33] M. L. Kremmer, M. Brimble, P. Stevenson-Clarke. Investigating the probability of student cheating: The relevance of student characteristics, assessment items, perceptions of prevalence and history of engagement. *International Journal for Educational Integrity* 3(2):3–17, 2007.
- [34] I. C. McManus, T. Lissauer, S. E. Williams. Detecting cheating in written medical examinations by statistical analysis of similarity of answers: pilot study. *British Medical Journal* 330 (7499):1064–1066, 2005.
- [35] L. S. Sotaridona, W. J. van der Linden, R. R. Meijer. Detecting answer copying using the kappa statistic. *Applied Psychological Measurement* 30(5):412–431, 2006.
- [36] L. A. van der Ark, W. H. M. Emons, K. Sijtsma. Detecting answer copying using alternate test forms and seat locations in small-scale examinations. *Journal of Educational Measurement* 45(2):99–117, 2008.
- [37] R. DiSario, A. Olinsky, J. Quinn, P. Schumacher. Applying Monte Carlo simulation to determine the likelihood of cheating on a multiple-choice professional exam. *CS-BIGS* 3 (1), 30–36, 2009.
- [38] D.R. White; M.S. Joy. Sentence-based natural language plagiarism detection. *Journal on Educational Resources in Computing* (JERIC) 4(4):1-20, 2004, ACM.
- [39] Sherlock Plagiarism Detector, disponível em: <http://sydney.edu.au/engineering/it/~scilect/sherlock>, Ago. 2011.
- [40] URKUND, disponível em: <http://www.urkund.com>, Ago. 2011.

Apêndices

A. Questões da prova

1) Conhecer a evolução de uma ciência pode ajudar a entender melhor as concepções atuais da sua teoria. Explique como as orientações da empresa para o mercado podem servir como argumentos para demonstrar como se processou a evolução do marketing.

2) A definição do negócio é uma das decisões que pode afetar consideravelmente as atividades de marketing de uma organização. Explique por que a definição de um negócio deve ser feita por mercado e não por produto.

3) O marketing tem sido descrito como ciência e também como arte. Explique como essas duas vertentes de pensamento podem ser consideradas verdadeiras.

4) Após vencer vários desafios de uma seleção profissional, restaram cinco candidatos disputando uma única vaga. A vaga em disputa era para a direção geral de uma unidade de negócio. Imagine que essa vaga pudesse ser ocupada por um profissional de administração de qualquer uma das especialidades: Marketing, Finanças, Recursos Humanos, Produção ou Informações. Coincidentemente essas eram as especialidades dos cinco candidatos restantes do concurso. Considerando que você era o candidato especialista em Marketing, explique resumidamente quais seriam as suas justificativas para ser o selecionado.

Figura 11: Grafo de similaridade para a questão 3

B. Grafos de similaridade das questões

Figura 10: Grafo de similaridade para a questão 2

Figura 12: Grafo de similaridade para a questão 1