Análise da evasão de alunos da área de tecnologia da informação por meio de um banco de dados orientado a grafos
Analysis of students evasion in information technology courses through a graph database

Kelvyn Yago da Silva Zanato[1], **Thiago Meirelles Ventura**[1], **Jivago Medeiros Ribeiro**[1]

[1] Instituto de Computação – Universidade Federal de Mato Grosso (UFMT)
Campus Cuiabá – MT – Brasil

kelvynzanatoo@gmail.com, thiago@ic.ufmt.br, jivago@ic.ufmt.br

Abstract. *The evasion of students in college is a problem reported in several papers. Regarding this problem, this work presents perspectives based on graph-oriented database to understand and identify students with high risk of evasion. Our approach uses exclusively the students' academic history, something easy to get to apply the proposed method. In this sense, it was calculated the similarity between current students and evaded students from previous classes. The results indicated patterns and showed that is possible to accurately identify 73% the final situation of the student. The proposed method may assist decision making to help students at risk, as well as the planning future actions, to reduce the amount of evaded students.*

Keywords*: data analysis, Neo4j, dropout.*

Resumo. *A evasão de estudantes no nível superior é um problema relatado em diversos trabalhos, e que precisa de estudos aprofundados. Neste contexto, são apresentadas perspectivas baseadas em banco de dados orientado a grafos com o intuito de entender e identificar estudantes com potencial de desistência. Foi utilizado exclusivamente o histórico escolar dos alunos, algo facilmente obtido para a aplicação da metodologia proposta. Cálculos de similaridade entre alunos atuais com alunos evadidos de turmas anteriores foram realizados. Os resultados indicam padrões entre os alunos evadidos e mostram que é possível identificar com precisão média de 73% a situação final do aluno. A metodologia proposta pode auxiliar na tomada de ações para ajudar alunos em risco, bem como no planejamento de ações futuras, a fim de diminuir a evasão dos estudantes.*

Palavras-chave*: análise de dados, Neo4j, evasão escolar.*

1. Introdução

A evasão de estudantes nas universidades brasileiras é um problema que atinge praticamente todos os cursos de nível superior, tendo como algumas consequências a formação de profissionais abaixo da capacidade desejada, frustração dos alunos que não conseguem concluir sua graduação e significativo desperdício de recursos [Hipólito 2011].

A evasão está relacionada a vários fatores, que podem ser divididos entre internos e externos. Os fatores internos são ligados ao curso, e podem ser classificados em: infraestrutura, corpo docente e a assistência sócio-educacional. Os fatores externos relacionam-se ao aluno, tais como: vocação, aspectos socioeconômicos e problemas pessoais [Paredes 1994].

É preciso fazer esforços que evitem ou diminuam a evasão em cursos superiores, em especial as instituições públicas, como forma de evitar a perda de recursos investidos. Esses esforços devem passar por processos de observação e compreensão dos aspectos, tanto gerais quanto específicos, levando em consideração as peculiaridades de cada curso e outros contextos, como os sociais, de ensino e administrativo de alunos e profissionais envolvidos na educação superior [Lobo 2012, Manhães et al. 2011].

Com o intuito de elucidar parte das situações que configuram essa problemática, acredita-se que a construção de instrumentos que auxiliem no acompanhamento da trajetória dos estudantes seja da grande relevância nesse contexto. Esses instrumentos auxiliariam, por exemplo, desde o planejamento para oferta de turmas extras até na identificação de padrões no histórico escolar de alunos evadidos contribuindo assim na predição de desistências.

Face ao exposto, este trabalho tem como objetivo analisar de diferentes perspectivas os dados relacionados à evasão, por meio da utilização de um banco de dados orientado a grafos. Essas análises podem possibilitar a criação de instrumentos para acompanhamento da trajetória do aluno, auxiliando os gestores acadêmicos.

2. Trabalhos Correlatos

O estudo do problema de evasão escolar possibilita identificar a sua relação com uma demanda importante na sociedade, no qual tanto universidades públicas como universidades particulares apresentam índices considerados altos. Diversos trabalhos vêm sendo realizados com o objetivo de identificação de tendências de evasão [Rigo et al. 2012]. Em [Manhães et al. 2014] é apresentado uma arquitetura que utiliza técnicas de mineração de dados para monitorar o progresso acadêmico dos estudantes e prever a aprovação dos mesmos nas disciplinas. Os alunos que não serão aprovados possuem um risco maior de evasão. Algoritmos de classificação foram utilizados para obtenção dos resultados.

Em outro trabalho de [Manhães et al. 2011] foi utilizado técnicas de mineração de dados para prever a evasão de estudantes em cursos presenciais da Escola Politécnica da Universidade do Rio de Janeiro. Dez modelos diferentes foram testados, com acurácia média variando entre 75% e 80%, com Perceptron de Múltiplas Camadas e Florestas Aleatórias apresentando os melhores desempenhos.

Em [Rodrigues et al. 2013] foi apresentado um estudo da viabilidade do uso de um modelo de regressão linear para também prever o desempenho dos alunos. Este trabalho também utilizou dados de um ambiente virtual de aprendizagem, além da utilização de séries temporais. Diversos atributos foram utilizados, incluindo a quantidade de interações ao longo das semanas no ambiente.

Em [Santos et al. 2015] foi realizado um estudo utilizando a análise do motivo da evasão e a aplicação de um questionário para alunos evadidos. Neste questionário havia diversas questões referentes a aptidões, anseios, repercussão da evasão no aluno e

possíveis intervenções no curso. Para o curso identificou-se como principais causas de evasão a insuficiência da infraestrutura de apoio ao ensino de graduação e a exigência de dedicação exclusiva ao curso. Já em [Fernandes e Junior 2016] foi feito uma estudo analisando documentos produzidos pelos professores com o intuito de comprovar a relação entre evasão e reprovação de disciplinas como lógica e programação.

Como pode ser visto, o problema de evasão está presente em várias instituições, e os pesquisadores estão utilizando diversos métodos para solucionar o problema. Os trabalhos citados auxiliam a coordenação de curso no contexto de evasão. Entretanto, ainda é necessário um método para, de forma mais específica, identificar os alunos que tendem a evadir.

Além disso, baseado nos trabalhos citados, é possível perceber que há uma dificuldade em descobrir quais são os principais atributos a serem analisados para realizar a previsão da evasão, sendo utilizado principalmente as notas dos alunos, mas sem considerar o histórico temporal dos mesmos. Este trabalho se diferencia por analisar diversas perspectivas, tendo como foco as relações entre os dados. Na Seção 4 são apresentadas tais perspectivas.

3. Materiais e Métodos

O tipo de banco de dados selecionado para este trabalho foi o orientado a grafos, a fim de permitir uma melhor identificação de padrões por meio das relações entre os dados. Especificamente, foi utilizado o banco de dados Neo4j [Neo4j 2018]. Este banco tem uma estrutura diferenciada dos bancos de dados relacionais no que se refere às relações dos dados. Ele guarda seus dados em forma de grafos, uma forma otimizada de armazenar e relacionar qualquer tipo de informação por mais abstrata que ela seja de uma forma acessível através das relações entre os nós. A tecnologia de banco de dados em grafos é uma ferramenta eficaz para a modelagem de dados quando o foco no relacionamento entre as entidades é uma força motriz na concepção dos dados [Miller 2013].

A vantagem de utilização do modelo baseado em grafos fica clara quando consultas complexas são exigidas pelo usuário. Comparado ao modelo relacional, estas situações podem ser muito custosas. O modelo orientado a grafos tem um ganho de performance, permitindo um melhor desempenho das aplicações [Lóscio et al. 2011]. Para tanto, é necessário uma linguagem específica para poder interagir com o banco. De acordo com [Neo4j 2018], *Cypher* é uma linguagem para executar as *queries* no Neo4j. É semelhante com a *Structured Query Language* (SQL), porém com propriedades únicas referentes ao banco de dados orientado a grafos. Em banco de dados relacionais existem tabelas e registros, já no *Cypher* as tabelas são identificadas como *labels* e registros como *nodes*. As relações entre as *labels* se chamam *relationship* que podem ou não ter propriedades e uma tabela em banco de dados relacionais existem campos que representam valores dos registros, como nome, idade e sexo. Já no *Cypher* existe um node no formato *key-value*, como {nome:'valor', idade:valor}. Como exemplo, pode ser feito uma instrução que lista os funcionários que trabalham no departamento de Tecnologia da Informação. Exemplos podem ser vistos em SQL (Código 1) e em *Cypher* (Código 2).

No exemplo, a consulta *Cypher* é mais simples e clara comparada à instrução em SQL. Não só a consulta *Cypher* será mais rápida para criar e executar, mas também reduz as chances de obter resultados indesejados [Michael Hunger 2016].

Código 1. Exemplo em SQL

```
SELECT nome FROM Pessoa
LEFT JOIN Pessoa_Departamento
   ON Pessoa.Id = Pessoa_Departamento.PessoaId
LEFT JOIN Departamento
   ON Departamento.Id = Pessoa_Departamento.DepartamentoId
WHERE Departamento.nome = "Tecnologia da Informacao"
```

Código 2. Exemplo em Cypher

```
MATCH (p:Pessoa)-[:TRABALHA]->(d:Departamento)
WHERE d.nome = "Tecnologia da Informacao"
RETURN p.nome
```

3.1. Obtenção e Descrição dos Dados Brutos

A base de dados utilizada neste trabalho foi elaborada a partir de dados extraídos do Sistema de Informações de Gestão Acadêmica (SIGA) da universidade para um arquivo de texto no formato *Comma-separated values* (CSV). A Tabela 1 contém um exemplo de como os dados estão distribuídos no arquivo, apresentando respectivamente informações do registro geral do aluno (RGA), tipo de ingresso, período que o aluno realizou a disciplina, semestre que a disciplina é ofertada, nome da disciplina, a nota do aluno, sua situação de aprovação e se o mesmo foi evadido.

Tabela 1. Distribuição dos dados no arquivo CSV

RGA	TIPO_INGRESSO	SEMESTRE	SEM_COUNT	NOME_DISCIPLINA	MEDIA	SITUACAO	EVADIDO
200000000001	1	2000/2	2	VETORES_E_GEOMETRIA_ANALITICA	8.25	AP	SIM
200100000002	3	2001/1	1	FUNDAMENTOS_DA_COMPUTACAO	6.00	AP	SIM
200200000003	2	2002/2	2	ALGEBRA_LINEAR	7.00	AP	SIM
200300000004	5	2003/1	1	FILOSOFIA_DA_CIENCIA	9.50	AP	SIM
201200000005	6	2013/2	4	LABORATORIO_DE_PROGRAMACAO	0.00	AE	SIM

Foram obtidos um total de 31332 registros de históricos escolares dos alunos e ex-alunos dos cursos de Ciência da Computação e Sistemas de Informação do Instituto de Computação da Universidade Federal de Mato Grosso no período entre 2000 e 2015.

3.2. Organização e Importação dos Dados

Com os dados obtidos por meio do método descrito em 3.1, a modelagem em grafo foi realizada, relacionando os alunos às suas disciplinas. Nesse tipo de banco as relações também contém informações. Para este trabalho, a relação de cursar possui informações sobre período, nota e situação que esse relacionamento aconteceu.

Além disso, foi feito um tratamento nos dados, transformando a coluna SEMESTRE em ANO e PERIODO (separando-os através da barra) e removendo os caracteres especiais da coluna NOME_DISCIPLINA, substituindo-os por espaços. Após estes passos os dados foram importados utilizando a própria ferramenta disponibiliza pelo Neo4j, tendo como exemplo um comando de importação como mostrado no Código 3.

As Figuras 1 e 2 mostram, respectivamente, um exemplo das relações das disciplinas cursadas de um único aluno e relações entre diferentes alunos, no qual uma mesma disciplina pode ter sido feita uma única vez por um aluno e mais de uma vez por outros alunos. Com os dados importados é possível criar consultas e realizar análises sobre as evasões de diferentes perspectivas.

Código 3. Exemplo de comando de importação com o Neo4j

```
LOAD CSV WITH HEADERS FROM
"file:///csv/aluno.csv"
AS csvLine FIELDTERMINATOR  ';'
CREATE (a:Aluno {id: toInt(csvLine.id), rga: toInt(csvLine.rga),
   tipo_ingresso:csvLine.tipo_ingresso, evadiu: (case csvLine.evadiu
   when 'SIM' then true else false end), curso: toInt(csvLine.curso)
})
```

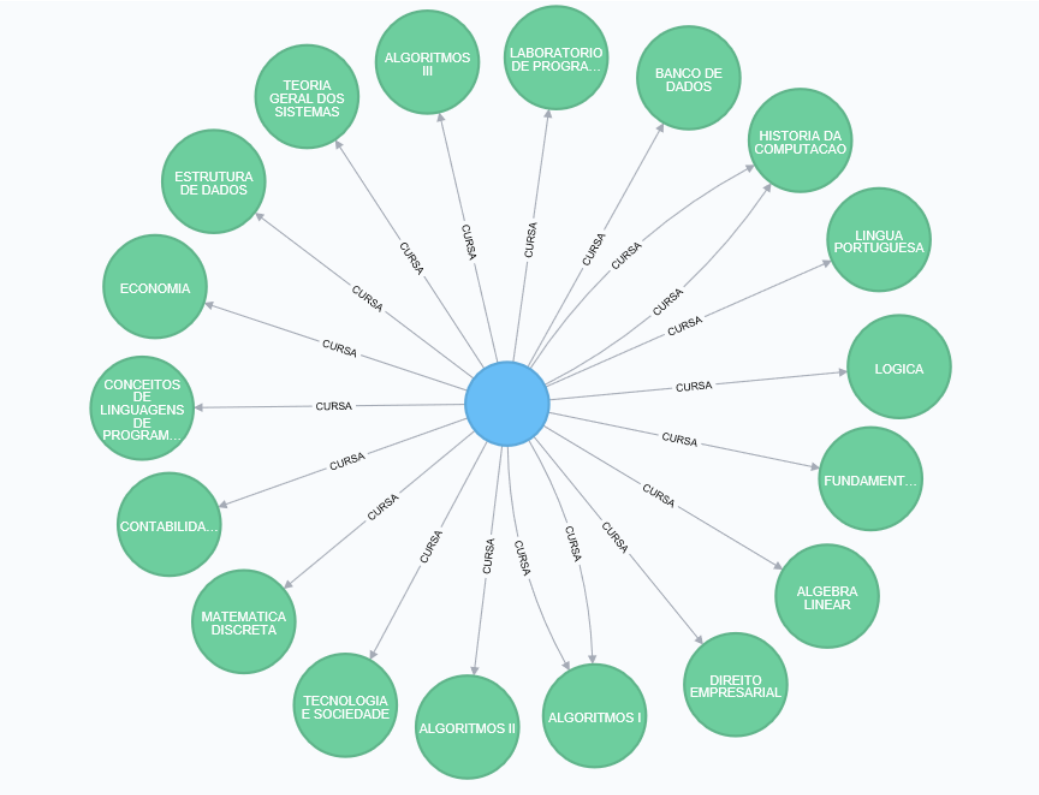

Figura 1. Representação em grafo de disciplinas cursadas por um aluno.

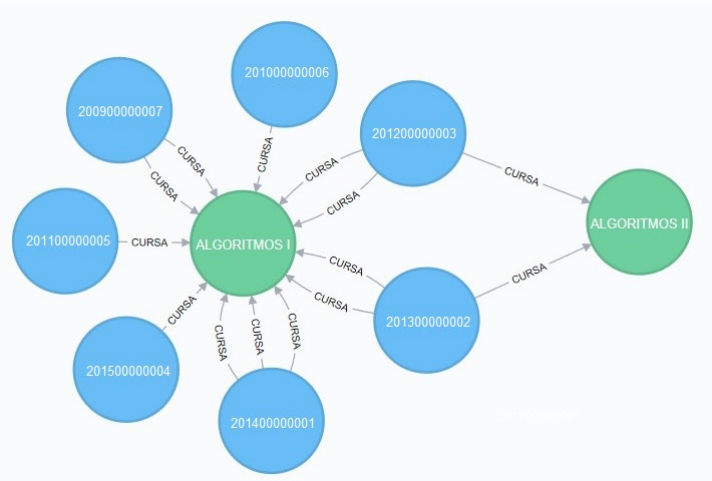

Figura 2. Representação em grafo de disciplinas cursadas por diferentes alunos.

3.3. Sobre os cursos de Ciência da Computação e Sistemas de Informação

O curso de Bacharelado em Ciência da Computação (CC) foi criando em 1990 e sua última atualização na matriz curricular do curso aconteceu em 2004. O curso funciona em período integral (vespertino e noturno), com carga horária de 3340 horas e integralização mínima de 8 semestres e máxima de 16 semestres. Além das disciplinas gerais da área de computação, o curso de CC possui disciplinas como: Cálculo I, II e III, Filosofia da Ciência e Metodologia Científica, Teoria dos Grafos, Teoria das Linguagens Formais e Autômatos, Compiladores I e II. Como trabalho de conclusão de curso o aluno pode optar por fazer uma Monografia ou um Estágio Obrigatório.

O curso de Bacharelado em Sistemas de Informação (SI) foi criado em 2008 e possui funcionamento noturno, em regime de crédito semestral, com uma carga horária total de 3120 horas com integralização mínima de 8 semestres e máxima de 12 semestres. O curso de SI possui disciplinas gerais da área de computação e disciplinas mais específicas, como Interface Humano-Computador, Administração de Redes e Sistemas de Apoio a Decisão, além de disciplinas nas grandes áreas de Português, Administração e Ética. Como trabalho de conclusão o aluno deve realizar um Estágio Obrigatório de 300 horas (já incluídas na carga horária total).

Outra informação importante é que ambos os cursos possuem um número considerável de pré-requisitos, como a necessidade do aluno ter cursado e ter sido aprovado em Algoritmos I para poder se matricular em Algoritmos II. Outro exemplo pode ser citado onde para matricular em Estrutura de Dados é preciso ter cursado e ter sido aprovado na disciplina de Algoritmos II e Laboratório de Programação, entre outros pré-requisitos. O curso de CC possui ainda alguns co-requisitos (disciplinas que devem ser cursadas concomitantemente).

4. Resultados e Discussões

Nesta seção são apresentadas as consultas criadas na linguagem *Cypher* e os respectivos dados retornados. As consultas foram elaboradas de modo a permitir a discussão e análise de perspectivas diversas, como: relação de disciplinas com evasão, padrão de histórico de alunos evadidos, impacto das reprovações no início do curso, dentre outras. Para tanto, serão apresentadas e descritas as consultas utilizadas para retornarem os dados necessários, analisando cada perspectiva.

A primeira questão discutida nesta seção refere-se a possível relação entre a reprovação em disciplinas específicas e a evasão de alunos. De modo a permitir a análise e discussão dessa questão, criou-se uma consulta envolvendo a quantidade de reprovações por disciplinas, representada no Código 4.

Código 4. Quantidade de reprovações por disciplina

```
MATCH (a:Aluno)-[c]->(d:Disciplina)
WHERE a.evadiu = TRUE
AND c.situacao IN ["RM","RMF","RF"]
RETURN (d.nome), count (c) AS Total
ORDER BY Total DESC
```

O Código 4 faz a consulta de todos os alunos evadidos e que foram reprovados por média (RM), por falta (RF) ou por ambos (RMF) em alguma disciplina. Os resultados são

apresentados na Tabela 2, que de acordo com a mesma, os maiores índices estão relacionados com as disciplinas iniciais (primeiro e segundo semestre) dos cursos, sendo que quatro das disciplinas fazem parte da matriz curricular de ambos os cursos (Algoritmos I, Lógica, Álgebra Linear e Fundamentos da Computação). Apenas Cálculo I faz parte exclusivamente da matriz do curso de Ciência da Computação. Mais informações sobre as disciplinas citadas neste trabalho podem ser encontradas no Anexo I.

Tabela 2. Disciplinas com maiores índices de reprovações

Disciplina	Total de reprovações
ALGORITMOS I	362
LOGICA	308
ALGEBRA LINEAR	264
CALCULO I	233
FUNDAMENTOS DA COMPUTACAO	231

Sabendo-se que a disciplina de Algoritmos I tem o maior índice de reprovação dos alunos evadidos, pode ser criado uma consulta para analisar quantos alunos evadidos reprovaram nesta disciplina.

Código 5. Total de alunos evadidos em uma disciplina específica

```
MATCH (a:Aluno)-[c]->(d:Disciplina{nome:'ALGORITMOS I'})
WHERE c.situacao IN ["RM","RMF","RF"]
WITH COUNT(DISTINCT a) AS TotalAlunosRep
MATCH (a:Aluno)-[c]->(d:Disciplina{nome:'ALGORITMOS I'})
WHERE c.situacao IN ["RM","RMF","RF"]
AND a.evadiu = TRUE
RETURN TotalAlunosRep,
  COUNT(DISTINCT a) AS QtdAlunosEvadidos
```

O Código 5 retorna o total de alunos distintos que foram reprovados na disciplina de Algoritmos I (um mesmo aluno pode ter mais de uma reprovação na mesma disciplina) e destes alunos, quantos foram evadidos. O resultado mostra que 388 alunos foram reprovados, sendo 247 evadidos. Portanto, 63,6% dos alunos que reprovaram em Algoritmos I evadiram.

Outra perspectiva é analisar a quantidade de reprovações de um aluno em um mesmo semestre, antes de evadir do curso. A análise desses dados pode ajudar a elucidar se a evasão acontece devido a reprovações em disciplinas isoladas ou não. Assim, foi analisada a quantidade de reprovações que os alunos recebem no mesmo semestre antes de evadir. Com as mesmas condições utilizadas pela consulta anterior e adicionando a condição para retornar o último semestre cursado pelos alunos, foi criado o Código 6 e obtidos os resultados apresentados na Figura 3.

Não há uma grande diferença na quantidade de reprovações em um único semestre para os alunos evadidos ao se analisar a Figura 3. No último semestre matriculado há várias ocorrências de evasão tanto quando houve poucas reprovações quanto quando houve muitas reprovações. As exceções estão na quantidade de 7 e 8 reprovações, mas isso se deve ao fato de acontecer poucos casos em que um aluno possui tantas disciplinas matriculadas em um mesmo semestre.

Ainda sobre a Figura 3, pode ser observado que os maiores índices de evasão ocorrem quando os alunos reprovam de 6 disciplinas no primeiro semestre (54 alunos), representando aproximadamente 14% do total de evasões (considerando 374 alunos evadidos com reprovações em seus históricos). Analisando apenas o primeiro semestre, dos 135 alunos evadidos que tiveram reprovações neste semestre, 81,5% possuem 4 ou mais reprovações.

Código 6. Reprovações por semestre

```
MATCH (a)-[c]->(d)
WHERE c.situacao IN ["RM","RMF","RF"]
AND a.evadiu = TRUE
WITH a, MAX(c.semestre) AS sem
MATCH (a)-[c]->(d)
WHERE c.semestre = sem
AND c.situacao IN ["RM","RMF","RF"]
RETURN a.rga as Aluno, sem, COUNT(d.nome) AS QtdDiscipRep
ORDER BY Aluno, sem
```

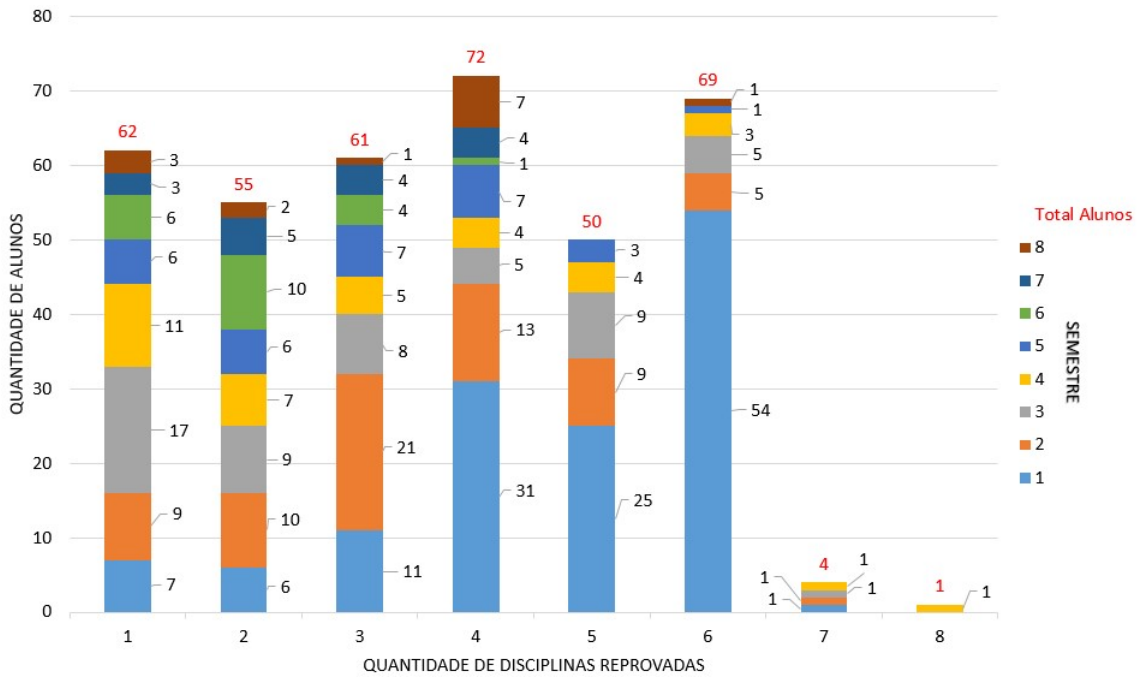

Figura 3. Quantidade de disciplinas reprovadas pelos alunos por semestre.

Os dados levantados referentes a evasão e reprovações em disciplinas do primeiro semestre abrem discussão para diferentes hipóteses: (i) os alunos estão tendo dificuldade nas disciplinas do primeiro semestre; (ii) a dificuldade pode ser em disciplinas específicas ou no conjunto de disciplinas referentes ao primeiro semestre; (iii) os alunos desistem do curso antes de terminar o primeiro semestre por diferentes motivos, principalmente pessoais. A análise de cada uma dessa hipóteses dependerá da realização de estudo qualitativo levantando os motivos de evasão junto aos alunos evadidos.

Acredita-se também que a evasão pode estar relacionada ao número de reprovações em sequência, ou seja, reprovações em semestres subsequentes antes do

aluno vir a evadir. O Código 7 recupera esta informação e o respectivo resultado é apresentado na Figura 4.

Código 7. Reprovações em semestres subsequentes

```
MATCH (a)-[c]->(d)
WHERE c.situacao IN ["RM","RMF","RF"]
AND a.evadiu = TRUE
RETURN DISTINCT a.rga AS Aluno, a.curso AS Curso,
  COUNT(DISTINCT (c.semestre)) AS QtdSem
ORDER BY Aluno
```

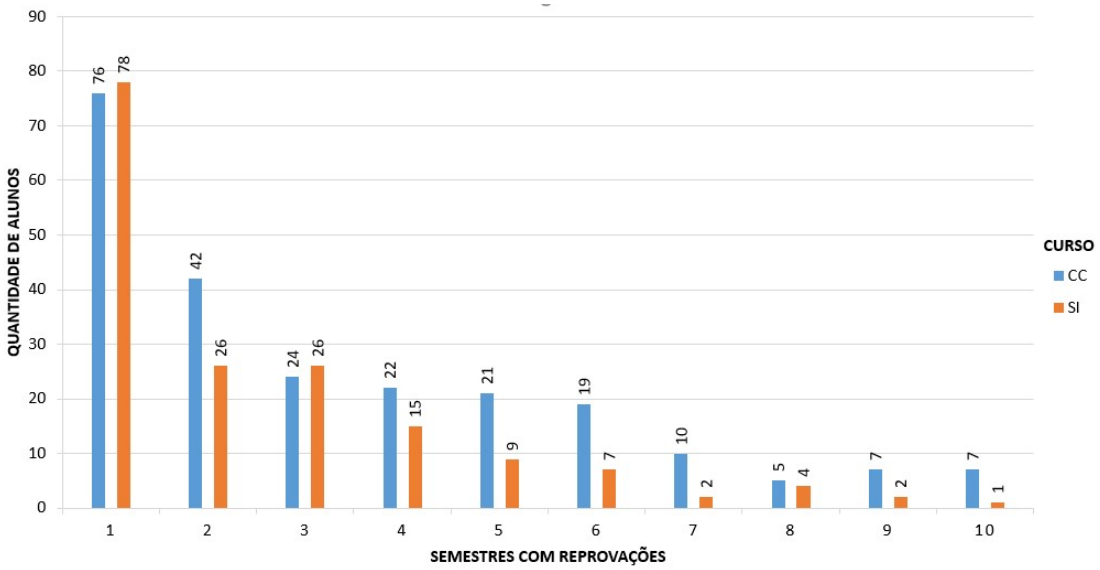

Figura 4. Reprovações em sequência de alunos por curso.

No total foram obtidos 403 alunos evadidos que tiveram reprovações em seus registros históricos, sendo 233 do curso de Ciência da Computação (CC) e 170 do curso de Sistemas de Informação (SI). Os resultados entre os cursos são semelhantes e corroboram com a hipótese de que reprovações em semestres iniciais podem estar ligadas a evasão, sugerindo ainda que a reprovação nos primeiros semestres possivelmente possui um peso maior na evasão dos alunos do que a reprovação sequencial. Juntamente com a relação anterior de sequência de reprovações, pode-se analisar a média de reprovações de uma mesma disciplina entre alunos evadidos, por meio do Código 8 e com resultados apresentados na Figura 5.

Código 8. Reprovações em uma mesma disciplina

```
MATCH (a)-[c]->(d)
WHERE c.situacao IN ["RM","RMF","RF"]
AND a.evadiu = TRUE
RETURN a.rga AS Aluno, COUNT(d.nome) AS QtdRep,
  d.nome AS NomeDisc
ORDER BY Aluno, NomeDisc
```

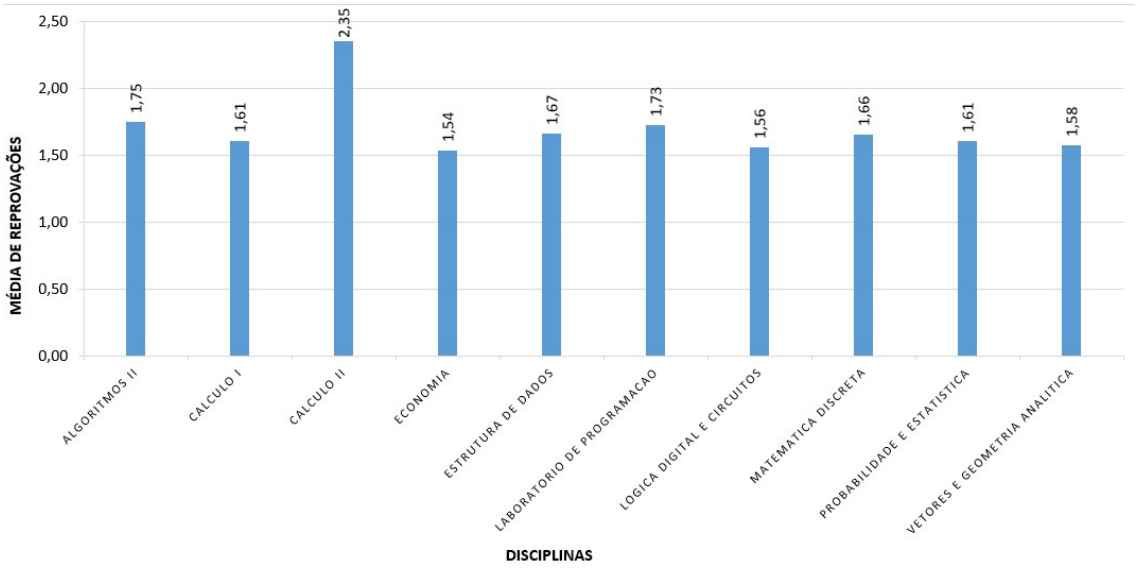

Figura 5. Disciplinas com maiores médias de reprovações de alunos evadidos.

Para obter os resultados da Figura 5 foi realizado um filtro removendo as disciplinas que possuem menos de dez alunos que foram reprovados e também as disciplinas que não fazem parte da matriz curricular atual dos cursos. Enquanto a Tabela 2 possui principalmente disciplinas do primeiro semestre, a Figura 5 traz, além de disciplinas do primeiro semestre (cálculo I e vetores e geometria analítica) disciplinas de segundo (algoritmos II e laboratório de programação), terceiro (estrutura de dados) e até quarto-semestre (estatística no caso de BCO). A análise desses dados pode indicar que até a integralização do quarto semestre, o aluno possui maior propensão a evadir caso reprove em disciplinas específicas. Entretanto, não é possível visualizar uma relação entre a quantidade de reprovações em uma mesma disciplina e a evasão.

Por fim, ainda testou-se a hipótese relacionada a previsão de evasão baseada na similaridade de padrões encontrados no histórico de alunos já evadidos. Para isso, a similaridade foi definida como o menor valor absoluto da diferença entre as médias do aluno analisado com alunos das turmas anteriores.

No estudo desta perspectiva foi utilizado como parâmetro o 4º semestre dos alunos e a turma de 2013 do curso de SI, devido ao curso possuir poucos registros. O Código 9 faz a busca destes alunos, trazendo todos os registros históricos até o 4º semestre e, em seguida, calcula a média das notas obtidas com a quantidade de disciplinas cursadas. São obtidos então os alunos de turmas anteriores e do mesmo curso que os alunos da turma de 2013, retornando as notas dos 4 primeiros semestres e também calculando a média de cada aluno. Por fim, a instrução é finalizada com um filtro onde verifica os alunos que realmente cursaram 4 semestres (pois no retorno anterior existem alunos que podem ter cursado apenas 1, 2 ou 3 semestres) e é feito o cálculo de similaridade.

Foram obtidos um total de 4480 comparações entre os alunos da turma de 2013 com os alunos de turmas anteriores. Dessas comparações, foi filtrada apenas a maior similaridade de cada aluno da turma de 2013 para realizar a análise, no qual um aluno é comparado com alunos de turmas anteriores que possuem o resultado de similaridade próximo ao deste aluno, obtendo-se no final 40 resultados, representados na Tabela 3.

Código 9. Consulta por similaridade de reprovações

```
MATCH (a)-[r1]->(d)
WHERE toInt(left(toString(a.rga), 4)) = 2013
AND a.curso = 316
AND r1.semestre <= 4
WITH a, SUM(ABS(r1.nota)) / COUNT(r1) as media,
  SUM(ABS(r1.nota)) as soma, COUNT(r1) as qtd
MATCH (a2)-[r2]->(d2)
WHERE toInt(left(toString(a.rga), 4)) >
  toInt(left(toString(a2.rga), 4))
AND a <> a2
AND a.curso = a2.curso
AND r2.semestre <= 4
WITH a, media, soma, qtd, a2,
  SUM(ABS(r2.nota)) / COUNT(r2) as media2,
  SUM(ABS(r2.nota)) as soma2, COUNT(r2) as qtd2,
  COLLECT(r2.semestre) as sem2
MATCH (a3)-[r3]->(d3)
WHERE 4 in sem2
RETURN DISTINCT a.rga, a2.rga, media, media2,
  a.evadiu, a2.evadiu, ABS(media-media2) as similaridade
ORDER BY similaridade, a.rga, a2.rga
```

Tabela 3. Análise de acerto do cálculo de similaridade entre alunos.

Situação	Total Alunos	Acertos Similaridade	%Acertos
Evadidos	22	15	68,2
Não Evadidos	18	14	77,8

De acordo com a Tabela 3, dos 40 alunos da turma de 2013, 22 realmente evadiram e 18 não evadiram (pelo menos até o período de 2015), sendo que dos 22 alunos evadidos, obteve-se 15 acertos. A mesma análise equivale aos alunos não evadidos, no qual dos 18 que ainda não evadiram, obtiveram similaridade com 14 alunos das turmas anteriores que não evadiram. Desta forma, é possível obter uma estimativa de alunos com risco de evasão após um determinado período de vida acadêmica, fornecendo uma informação importante aos coordenadores de cursos.

Algumas discussões podem ser analisadas sobre as limitações deste trabalho. Pode ser incluído como taxa de erro da análise os alunos que obtiveram aproveitamento de disciplinas, pois no sistema acadêmico algumas dessas disciplinas são lançadas com nota zero. Os resultados também são influenciados pelos alunos considerados *outliers* que evadem do curso mesmo com alto rendimento acadêmico. Como mencionado anteriormente, há os fatores externos relacionado ao aluno, como aspectos socioeconômicos e problemas pessoais. Nesses casos, a previsão é dificilmente bem sucedida. A falta de dados relacionados aos aspectos sociais dos alunos limitam as análises mas, por outro lado, a exigência de poucos tipos de dados para a realização desses experimentos facilita a aplicação do método por outras universidades.

5. Conclusão

Este trabalho teve como objetivo identificar padrões e fazer previsões de evasão nos cursos de Ciência da Computação e Sistemas de Informação da Universidade Federal de Mato

Grosso, analisando dados de turmas anteriores utilizando banco de dados orientado a grafos, o Neo4j. A aplicação desta metodologia nos cursos de ensino superior podem auxiliar na tomada de decisão de maneira antecipada por parte dos coordenadores e da própria instituição, a fim de diminuir os índices de evasão.

A identificação dos alunos que apresentam risco de evasão por meio do uso de grafos mostrou-se possível. Os resultados mostraram previsões de alunos com risco de evasão com base na função de similaridade com alunos de turmas anteriores de acordo com o respectivo curso. Os experimentos retornaram dados com precisão média de 73% para previsão de evasão.

Além disso, outras perspectivas foram analisadas. De diferentes formas os resultados mostraram que as disciplinas iniciais dos cursos tem uma importância maior no que se refere a evasão dos alunos. Portanto, uma atenção especial por parte dos gestores deve ser dada para os alunos que estão nos primeiros semestres.

Com relação ao motivo da evasão e reprovações nos primeiros semestres, três hipóteses foram levantadas. Como trabalhos futuros há a intenção de realizar pesquisas qualitativas para avaliar tais hipóteses, além de analisar dados de outras instituições e de outros cursos. Desta forma, espera-se beneficiar uma maior quantidade de gestores e, assim, auxiliar amplamente a comunidade acadêmica.

Referências

Fernandes, V. S. e Junior, V. F. (2016). Evasão e reprovação nas disciplinas de lógica e programação: Informações preliminares no campus sombrio, do instituto federal catarinense. In *Anais do 5º Simpósio de Integração Científica e Tecnológica do Sul Catarinense*, Araranguá-SC.

Hipólito, O. (2011). O gargalo do ensino superior brasileiro: Depoimento. `https://www.cartacapital.com.br/sociedade/o-gargalo-do-ensino-superior-brasileiro`. Entrevista concedida a Fernando Vives. Acesso em: 17/04/2017.

Lobo, M. B. C. M. (2012). Panorama da evasão no ensino superior brasileiro: aspectos gerais das causas e soluções. *Associação Brasileira de Mantenedoras de Ensino Superior. Cadernos*, (25).

Lóscio, B. F., OLIVEIRA, H. R., e PONTES, J. C. S. (2011). Nosql no desenvolvimento de aplicações web colaborativas. In *VIII Simpósio Brasileiro de Sistemas Colaborativos*, volume 10, page 11.

Manhães, L. M. B., Cruz, S. M. S., Costa, R. J. M., Zavaleta, J., e Zimbrão, G. (2011). Previsão de estudantes com risco de evasão utilizando técnicas de mineração de dados. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, volume 1.

Manhães, L. M. B., da Cruz, S. M. S., e Zimbrão, G. (2014). Wave: an architecture for predicting dropout in undergraduate courses using edm. In *Proceedings of the 29th Annual ACM Symposium on Applied Computing*, pages 243–247. ACM.

Michael Hunger, Ryan Boyd, W. L. (2016). Rdbms & graphs: Sql vs. cypher query languages. https://neo4j.com/blog/sql-vs-cypher-query-languages/. Acesso em: 11/05/2017.

Miller, J. J. (2013). Graph database applications and concepts with neo4j. In *Proceedings of the Southern Association for Information Systems Conference, Atlanta, GA, USA*, volume 2324, page 36.

Neo4j (2018). Intro to cypher. https://neo4j.com/developer/cypher-query-language/. Acesso em: 15/01/2018.

Paredes, A. S. (1994). *A evasão do terceiro grau em Curitiba*. Núcleo de Pesquisas sobre Ensino Superior, Universidade de São Paulo.

Rigo, S. J., Cazella, S. C., e Cambruzzi, W. (2012). Minerando dados educacionais com foco na evasão escolar: oportunidades, desafios e necessidades. In *Anais do Workshop de Desafios da Computação Aplicada à Educação*, pages 168–177.

Rodrigues, R. L., Medeiros, F. P., e Gomes, A. S. (2013). Modelo de regressão linear aplicado á previsão de desempenho de estudantes em ambiente de aprendizagem. In *Anais do XXIV SBIE*.

Santos, N. V. M., Junior, M. L., e Ribeiro, M. L. L. (2015). Evasão no curso de engenharia de produção da universidade federal de goiás-regional catalão. In *Anais do ENEGEP*, Fortaleza-CE.

Anexo I - Ementa das disciplinas

Álgebra linear: Matrizes. Sistemas de equações lineares. Vetores. Espaços vetoriais. Dependência e independência linear. Transformações lineares. Equações diferenciais lineares. Sistemas lineares. Autovalores e autovetores.

Algoritmos I: Características básicas de organização de um computador. Conceito de algoritmos e programação. Algoritmos: representação, técnicas e estruturas de elaboração. Tipos de dados: conceituação, representação e manipulação. Representação de dados. Solução de problemas numéricos e não-numéricos através de algoritmos.

Algoritmos II: Tipos de dados simples e estruturados. Refinamento de algoritmos. Modularização: Blocos e subprogramas. Parâmetros e formas de passagem. Escopo de identificadores: tempo de vida e visibilidade. Operações com arquivos. Recursividade.Variáveis dinâmicas. Abstração de dados. Estruturas de dados dinâmicas: listas lineares.

Cálculo I: Funções. Limites. Derivadas e Aplicações. Diferenciais e Aplicações. Integrais Definidas e Indefinidas.

Cálculo II: Técnicas de Integração. Aplicações do Cálculo Integral. Sequências e séries. Séries de Potência.

Economia: Conceito de Economia. Problemas econômicos. Noções de funcionamento de uma economia moderna do ponto de vista global. Sistemas econômicos. Noções de Macro e Microeconomia. Dificuldades estruturais de uma economia subdesenvolvida. O conceito de economia digital.

Estrutura de dados: Listas lineares e suas generalizações: listas ordenadas, listas encadeadas, pilhas e filas. Aplicações de listas. Árvores e suas generalizações: árvores binárias, árvores de busca, árvores balanceadas (AVL), árvores B e B+. Aplicações de árvores. Pesquisa e ordenação: algoritmos para pesquisa e ordenação em memória principal e secundária (listas, árvores, hashing, cadeias, etc).

Filosofia da ciência: O ser humano: finalidade, direito e função. O pensamento crítico: verdade e interpretação, conhecimento e ideologia. Totalidade da razão: o noético , o ético e o estético. O conhecimento científico. Eu: autoconsciência e autodeterminação. A dialética dos contrários e o jurídico. A importância da lógica utilizada pelo pesquisador para a construção da ciência.

Fundamentos da computação: Breve histórico dos computadores. Um modelo de computadores: memória, registradores, periféricos. Sistemas de Numeração. Linguagem de Programação de alto nível e de montagem (exemplos). O uso de computadores, impacto social. Áreas de aplicações de informática. Familiarização com o uso de sistemas e ambientes operacionais. Instalação e configuração de sistemas operacionais. Aspectos avançados de editores de texto e planilhas eletrônicas de cálculo.

Laboratório de programação: Estudo de construções sintáticas de duas linguagens de programação. Compiladores e/ou interpretadores de código. Compilação, montagem e ligação de código. Implementação de algoritmos em uma linguagem de programação. Codificação, compilação, edição e montagem via linha de comando. Uso de ambientes integrados de desenvolvimento. Teste e depuração de código. Metodologia de desenvol-

vimento de programas. Estilos de programação. Qualidade e documentação de código.

Lógica: Sentido lógico-matemático convencional dos conectivos. Argumentos. Lógica sentencial. Regras de formação de fórmulas. Sistemas dedutivos. Decidibilidade da lógica sentencial. A lógica de predicados de primeira ordem. Valores-verdade. Funções de avaliação.

Lógica digital e circuitos: Instrumentação eletrônica digital. Eletrônica analógica e digital básica. Circuitos elétricos e circuitos eletrônicos básicos. Implementação de portas lógicas com transistores e diodos. Famílias lógicas. Flip-flops, registradores, contadores e memórias. Osciladores e relógios. Circuitos combinacionais: análise e síntese. Dispositivos lógicos programáveis. Circuitos seqüenciais: análise e síntese. Introdução aos sistemas digitais. Laboratório de Circuitos Digitais: utilização de softwares de EDA e bancada de montagem de circuitos.

Matemática discreta: Conjuntos. Funções. Relações sobre conjuntos: relações de equivalência e de ordem. Indução matemática. Recursão. Sistemas algébricos. Grupos. Anéis e Corpos. Análise Combinatória: Distribuição. Permutação. Combinação. Enumeração por recursão. Cardinalidade de união de conjuntos. Enumeração de conjunto.

Probabilidade e estatística: Estatística Descritiva. Probabilidade. Probabilidade Condicional e independência. Funções de variáveis aleatórias. Momentos. Variáveis aleatórias bidimensionais. Confiabilidade. Amostragem. Distribuição amostrais. Estimação de parâmetros. Testes de hipóteses. Regressão e correlação.

Vetores e geometria analítica: Vetores no Rn. Operações com vetores no Rn. Independência Linear, Retas e Planos, Cônicas e Quádricas, Hiperplanos. Matrizes, Determinantes e Sistemas Lineares.