UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO DE CIÊNCIAS, TECNOLOGIAS E SAÚDE DO CAMPUS ARARANGUÁ
CURSO DE GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO

Giulio Postingher Brugalli

Um Método Baseado em Predição de *Links* Voltado à Gestão de *Leads*

Araranguá
2022

Giulio Postingher Brugalli

Um Método Baseado em Predição de *Links* Voltado à Gestão de *Leads*

Trabalho de Conclusão do Curso de Graduação em Engenharia de Computação do Centro de Ciências, Tecnologias e Saúde do Campus Araranguá da Universidade Federal de Santa Catarina para a obtenção do título de Bacharel em Engenharia de Computação.

Orientador: Prof. Alexandre Leopoldo Gonçalves, Dr.

Araranguá
2022

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Brugalli, Giulio Postingher
Um método baseado em predição de links voltado à gestão de leads / Giulio Postingher Brugalli ; orientador, Alexandre Leopoldo Gonçalves, 2022.
32 p.

Trabalho de Conclusão de Curso (graduação) - Universidade Federal de Santa Catarina, Campus Araranguá, Graduação em Engenharia de Computação, Araranguá, 2022.

Inclui referências.

1. Engenharia de Computação. 2. Gestão de Leads. 3. Aprendizado de Representação de Rede. 4. Predição de Links. I. Gonçalves, Alexandre Leopoldo. II. Universidade Federal de Santa Catarina. Graduação em Engenharia de Computação. III. Título.

Giulio Postingher Brugalli

Um Método Baseado em Predição de *Links* Voltado à Gestão de *Leads*

Este Trabalho de Conclusão de Curso foi julgado adequado para obtenção do Título de Bacharel em Engenharia de Computação e aprovado em sua forma final pelo Curso de Graduação em Engenharia de Computação.

Araranguá, 21 de julho de 2022.

Documento assinado digitalmente
Analucia Schiaffino Morales
Data: 03/08/2022 09:24:53-0300
CPF: 622.256.420-87
Verifique as assinaturas em https://v.ufsc.br

_____
Profª. Analúcia Schiaffino Morales, Dra .
Coordenadora do Curso

Banca Examinadora:

Documento assinado digitalmente
Alexandre Leopoldo Goncalves
Data: 03/08/2022 16:57:04-0300
CPF: 786.740.349-34
Verifique as assinaturas em https://v.ufsc.br

_____
Prof. Alexandre Leopoldo Gonçalves, Dr.
Orientador
Universidade Federal de Santa Catarina

Documento assinado digitalmente
Andrea Sabedra Bordin
Data: 03/08/2022 08:07:45-0300
CPF: 626.193.220-49
Verifique as assinaturas em https://v.ufsc.br

_____
Profª. Andréa Sabedra Bordin, Drª.
Avaliadora
Universidade Federal de Santa Catarina

Documento assinado digitalmente
Leticia Silveira Artese
Data: 02/08/2022 11:30:00-0300
CPF: 383.125.998-43
Verifique as assinaturas em https://v.ufsc.br

_____
Letícia Silveira Artese, M.Sc.
Avaliadora
Universidade Federal de Santa Catarina

Um Método Baseado em Predição de *Links* Voltado à Gestão de *Leads*

A Link Prediction-Based Method Towards Lead Management

Giulio Postingher Brugalli[1] Alexandre Leopoldo Gonçalves[2]

2022, Julho

Resumo

A análise de dados tem se tornado crucial para estratégias de sucesso nas organizações, principalmente quando se pensa nas etapas de aquisição e retenção de clientes. Para tais etapas, o acompanhamento e gestão dos *leads* é uma parte essencial. Todavia, à medida que o número de *leads* aumenta, a gestão se torna complexa e pouco eficiente, resultando em *leads* desqualificados e perda de tempo para o time de vendas. Desta forma, uma gestão de *leads* automatizada e baseada em dados é fundamental para otimizar a aquisição e retenção de clientes. Neste contexto, o presente trabalho propõe um método de apoio à gestão de *leads* para identificar e recomendar, para o time de vendas, futuros interesses de *leads* já existentes na base de dados de uma organização com intuito de adquirir ou reter clientes. Para cumprir este objetivo, explora-se o aprendizado de representação de redes através dos algoritmos *Node2Vec* e *Metapath2Vec* e modelos de predição de *links* para identificar possíveis tendências de conexões entre *leads* e produtos. Um estudo de caso utilizando dados de uma organização é apresentado para demonstrar a efetividade do método proposto. Para tal, foram realizadas análises de predições com diferentes estruturas topológicas de redes e calculado um coeficiente de generalização γ para qualificar os melhores modelos. Todos os modelos gerados atingiram um valor entre 0,873 e 0,998 considerando a métrica ROC-AUC, e os 3 melhores modelos apresentaram os valores de γ de 0,062, 0,018, 0,011, respectivamente. Diante dos resultados, os modelos de predição apresentaram baixos valores do coeficiente γ, muito distantes de 1, valor ideal. Porém, o método se mostra promissor para ser investigado na prática, ou seja, ativar os *leads* recomendados para convertê-los em clientes. Para trabalhos futuros é sugerido um aprofundamento em capacidades técnicas de aprendizado de redes para obter melhores resultados dos modelos de predição de *links*.

Palavras-chave: Gestão de leads. Aprendizado de representação de redes. Predição de *links*.

[1] giuliobrugalli@gmail.com
[2] a.l.goncalves@ufsc.br

Um Método Baseado em Predição de *Links* Voltado à Gestão de *Leads*

A Link Prediction-Based Method Towards Lead Management

Giulio Postingher Brugalli[3] Alexandre Leopoldo Gonçalves[4]

2022, Julho

Abstract

Data analysis has become crucial for successful strategies in organizations, especially when considering the stages of customer acquisition and retention. For such steps, tracking and managing leads is an essential part. However, as the number of leads increases, management becomes complex and inefficient, resulting in disqualified leads and lost time for the sales team. In this way, automated and data-based lead management is critical to optimizing customer acquisition and retention. In this context, the present work proposes a method to support lead management to identify and recommend, to the sales team, future interests of leads that already exist in an organization's database in order to acquire or retain customers. To fulfill this objective, we explore the learning of network representation through Node2Vec and Metapath2Vec algorithms and link prediction models to identify possible trends in connections between leads and products. A case study using data from an organization is presented to demonstrate the effectiveness of the proposed method. To this end, prediction analyzes were performed with different topological structures of networks and a generalization coefficient was calculated to qualify the best models. All generated models reached a value between 0.873 and 0.998 considering the ROC-AUC metric, and the 3 best models presented values of 0.062, 0.018, 0.011, respectively. In view of the results, the prediction models showed low coefficient values, very far from 1, the ideal value. However, the method shows promise to be investigated in practice, that is, to activate recommended leads to convert them into customers. For future work, a deepening of technical capabilities of network learning is suggested to obtain better results from link prediction models.

Keywords**:** *Lead management*. *Network representation learning. Link prediction*.

[3] giuliobrugalli@gmail.com
[4] a.l.goncalves@ufsc.br

1 INTRODUÇÃO

Segundo Saura, Ribeiro-Soriano e Palacios-Marqués (2021), os novos negócios têm o desafio de empreender a partir de um ecossistema conectado, isto é, canais de comunicação, *marketing* e vendas integrados aos sistemas internos das organizações, onde a análise de dados tem se tornado crucial para estratégias de sucesso, principalmente quando se consideram estratégias de aquisição e retenção de clientes. O acompanhamento e gestão dos *leads* é uma parte essencial para a aquisição de novos clientes. Existe um investimento substancial na geração de *leads*, mas estatísticas mostram que a maioria deles são ignorados e nunca contatados (OHIOMAH; BENYOUCEF; ANDREEV, 2016).

O conceito de *lead* se define como um registro de interesse, por alguma pessoa, em algum produto ou serviço de uma organização, independentemente se essa pessoa já foi um cliente ou não. Esse registro normalmente possui informações básicas para que um vendedor entre em contato com esse potencial cliente. Ou seja, enquanto uma pessoa possui interesse em algum produto ou serviço, mas ainda não o adquiriu, ela é um *lead* (OHIOMAH; BENYOUCEF; ANDREEV, 2016).

Porém, a aquisição de novos clientes é um processo constituído de várias etapas e, à medida que o número de *leads* aumenta, esse processo se torna complexo e pouco eficiente (YU; CAI, 2007; D'HAEN; VAN DEN POEL; THORLEUCHTER, 2013). Para um representante de vendas que geralmente possui poucos recursos para selecionar os melhores *leads* de forma inteligente e racional, a prospecção de clientes começa a ser ditada por regras baseadas na intuição, acarretando perda de tempo e dinheiro com contatos muitas vezes irrelevantes. Por isso, como D'haen *et al.* (2016) afirmam, um sistema automatizado de gestão de *leads* é imprescindível para qualificar e otimizar a aquisição e retenção de clientes. Segundo Gebert (2002), a gestão de *leads* é a consolidação, qualificação e priorização de contatos em potenciais clientes, provendo à equipe de vendas uma lista de *leads* qualificada e priorizada.

Normalmente, a gestão de *leads* é um dos processos que constituem um sistema mais completo, chamado de Gestão de Relacionamento com o Cliente (do inglês *Customer Relationship Manager* - CRM). Embora não haja uma definição comum para o conceito de CRM, para este trabalho compreende-se como um conjunto de ferramentas de *software* desenvolvidas para administrar vendas, *marketing* e serviços, que são áreas associadas ao relacionamento empresa-cliente (LAMRHARI *et al*., 2022; GIL-GOMEZ *et al*., 2020). De forma mais ampla, Lamrhari (2022) define CRM como uma estratégia de negócios, que conta com a tecnologia e métodos científicos, para identificar possíveis clientes, construir relacionamento e promover a retenção dos mesmos, minimizando assim os custos de *marketing*.

Uma vez que as tomadas de decisões baseadas em dados são cada vez mais comuns, a implementação e utilização de um CRM se torna imprescindível para as organizações, justamente devido a sua característica de facilitar a coleta, análise e exploração de dados relacionados às necessidades e preferências do cliente, obtendo conhecimentos e informações comerciais orientadas para o sucesso (GIL-GOMEZ *et al*., 2020; GUEROLA-NAVARRO *et al*., 2021a; SAURA; RIBEIRO-SORIANO; PALACIOS-MARQUÉS, 2021). Tais características, concedem ao CRM a capacidade de inovação, considerada um fator chave de sucesso para o desempenho organizacional das empresas, motivando-as a desenvolver produtos, serviços e práticas úteis e inovadoras, gerando mais valor para todos os *stakeholders* (GUEROLA-NAVARRO *et al*., 2021b).

A partir dessa perspectiva, entende-se que a gestão de *leads* é uma etapa que deve se beneficiar dos dados gerenciados pelo CRM. Isso fica mais evidente quando D'Haen *et al.* (2016) definem os dois critérios que um sistema precisa para qualificar um *lead*: dados de

qualidade e um modelo de aprendizado de máquina (do inglês *Machine Learning* - ML) para descobrir as relações escondidas nesses dados. Neste sentido, diversos algoritmos de ML já foram aplicados em várias dimensões de um CRM para resolver diferentes problemas, tais como segmentação de clientes, ciclo de vida do cliente, venda cruzada e análise do cliente alvo. Mais especificamente, na área de gestão de *leads*, existem aplicações de ML para pontuação de *leads* e redução do tempo para qualificação de *leads* (CHAGAS *et al*., 2018).

Porém, nas pesquisas para esse trabalho não foram encontrados modelos de ML que utilizassem da ciência de redes para entender o comportamento do *lead* e auxiliar na sua gestão. A ciência de redes é considerada um campo de estudo recente e com potencial para compreender comportamentos e padrões de problemas complexos e de naturezas multidisciplinares. Entre as possíveis tarefas a serem elaboradas a partir de uma rede encontra-se a predição de *link*.. Pode-se entender a predição de *link* como o processo de prever futuras conexões, entre pares de vértices de uma rede, baseadas em conexões já existentes (BARABÁSI, PÓSFAI; 2016; DAUD *et al*., 2020).

Sendo assim, este trabalho tem sua relevância ao apontar um possível método de gestão de *leads* a partir do conceito de predição de *links*, para contribuir com a identificação e recomendação, para o time de vendas, de futuros interesses de *leads* já existentes na base de dados de uma organização, com intuito de adquirir ou reter clientes. Ademais, espera-se que os resultados obtidos possam ajudar na promoção de novas pesquisas em temas correlatos.

Além desta seção, este trabalho é composto por cinco outras. A segunda seção apresenta a fundamentação teórica deste trabalho abrangendo conceitos essenciais para a compreensão do tema em sua totalidade. Na terceira seção, discussões acerca dos trabalhos relacionados à gestão de *leads* e predição de *links* são realizadas. O método proposto é descrito na quarta seção e então seus resultados são analisados e discutidos na quinta seção. Por fim, na sexta seção são apresentadas as conclusões e elencadas sugestões de trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 GESTÃO DE *LEADS*

As tecnologias de vendas permitem que organizações gerenciem melhor o relacionamento com seus clientes através da automação de tarefas rotineiras, permitindo que mais tempo seja direcionado para melhor atender as necessidades dos clientes (BRADFORD; JOHNSTON; BELLENGER, 2016). O CRM pode ser compreendido como um aglomerado de tecnologias de vendas, pois ele envolve etapas que agilizam e organizam o processo de negócio: gerenciamento de campanhas, gestão de *leads*, gerenciamento de ofertas, gerenciamento de contratos, gerenciamento de reclamações e gerenciamento de serviços. Além de facilitar o contato direto com o cliente, essas etapas viabilizam o processamento de informações com intuito de gerar *insights* voltados ao *marketing* de relacionamento (GEBERT *et al*., 2002). Estudos mostram uma relação positiva entre as práticas de CRM e o desempenho da empresa (AHEARNE; HUGHES; SCHILLEWAERT, 2007).

A etapa de gestão de *leads*, que é responsável por consolidar, qualificar e priorizar os *leads* para identificar potenciais clientes, apoia a equipe de vendas principalmente quando o volume de *leads* é muito grande e quando a qualificação se torna complexa. Além de auxiliar nas tarefas com clientes em potencial, um sistema de gestão de *leads*, como uma ferramenta de CRM, utiliza da base de clientes de uma organização para apoiar os objetivos de gerenciamento de relacionamentos com clientes, interpretando essas informações e tornando as tarefas de vendas mais eficientes através da qualificação e priorização de clientes que

podem consumir novamente (MERO; TARKIAINEN; TOBON, 2020; OHIOMAH; BENYOUCEF; ANDREEV, 2016).

Apesar da extensa literatura sobre comportamentos de clientes, existem poucos artigos com foco na caracterização e qualificação de *leads*. Não existe de fato uma definição universal aceita sobre como qualificar um *lead*, pois as características levadas em consideração vão depender da necessidade e do modelo do negócio (MONAT, 2011; GIACOSA, CULASSO, CROCCO; 2022). Em um dos primeiros estudos realizados sobre a modelagem de *leads*, Kestnbaum e Hsieh (1983) consideraram quais características seriam importantes para identificar potenciais clientes, entre elas: tempo, número de compras anteriores, histórico de consulta anterior, experiência anterior com a empresa e os produtos da empresa, experiência anterior de compra com concorrentes, tamanho da empresa, adequação do produto para a aplicação pretendida, importância do cliente potencial e se o cliente potencial solicita ou não uma demonstração. Desde então, alguns autores se aventuraram nos conceitos de modelagem de *leads* trazendo interpretações e características diferentes do que eles consideram primordial no momento de qualificar um *lead* em potencial cliente (MONAT, 2011).

A complexidade da qualificação dos *leads* não fica somente na área acadêmica. Também existem divergências entre a área de *marketing* e vendas das organizações. A equipe de vendas reclama da baixa qualidade dos *leads*, e o *marketing* reclama do acompanhamento ruim (ou falta de acompanhamento) da área de vendas com esses *leads* (SABNIS *et al*., 2013). Como resultado, a equipe de vendas se concentra apenas nos *leads* que são considerados mais promissores, através de interpretações baseadas em experiência e intuição (D'HAEN *et al*., 2016). Estudos mostram que em empresas *business-to-business*, representantes de vendas não entram em contato com cerca de 70% dos *leads* gerados pelo *marketing* (MICHIELS, 2009). Esse problema é conhecido como "buraco negro de *leads* de vendas", representando a enorme quantidade de *leads* que ficam no limbo, sem serem contatados (SABNIS, 2013).

Como solução, a literatura mostra que o uso adequado de tecnologia na área de *marketing* e vendas pode aumentar significativamente a eficiência da gestão de *leads* (JÄRVINEN; TAIMINEN, 2016). Bucklin, Lehmann e Little (1998), previram que em 2020, uma proporção das decisões de *marketing* seria automatizada devido às demandas de customização em massa, melhor tomada de decisão e maior produtividade. Em relação ao gerenciamento de *leads*, os resultados sugerem que isso de fato pode ocorrer quando um sistema de gestão de *leads* é adequadamente implementado por uma infraestrutura de Tecnologia da Informação (TI) robusta e confiável (GIACOSA; CULASSO; CROCCO, 2022). Por fim, segundo Ohiomah *et al*. (2019), vendedores que usam sistema de gestão de *leads* de forma efetiva tendem a aumentar seu desempenho nas vendas visto que conseguem realizar as tarefas envolvidas na aquisição de potenciais clientes de maneira mais eficiente.

2.2 PREDIÇÃO DE *LINKS*

Atualmente, as pessoas estão cercadas por sistemas considerados complexos, desde a sociedade que requer a cooperação de bilhões de indivíduos, nossa infraestrutura de comunicação que integra bilhões de dispositivos, até nosso cérebro que é formado por bilhões de neurônios interconectados. Esses e tantos outros sistemas são considerados complexos devido à dificuldade de deduzir seu comportamento coletivo a partir do conhecimento dos componentes do sistema (BARABÁSI, PÓSFAI; 2016). Em outras palavras, sistemas complexos são sistemas em que o coletivo possui propriedades que não podem ser derivadas pela agregação dos constituintes (KASTHURIRATHNA, 2015).

Sistemas complexos têm atraído grande interesse de pesquisa nas últimas duas décadas. Assim, nasce no século XXI um novo campo de estudo: a ciência das redes, que

modela esses sistemas como redes complexas e promove uma linguagem em que diferentes disciplinas podem interagir, abrindo possibilidade de investigar questões multidisciplinares a partir dessas redes (BARABÁSI, PÓSFAI; 2016; MOLONTAY, NAGY; 2019).

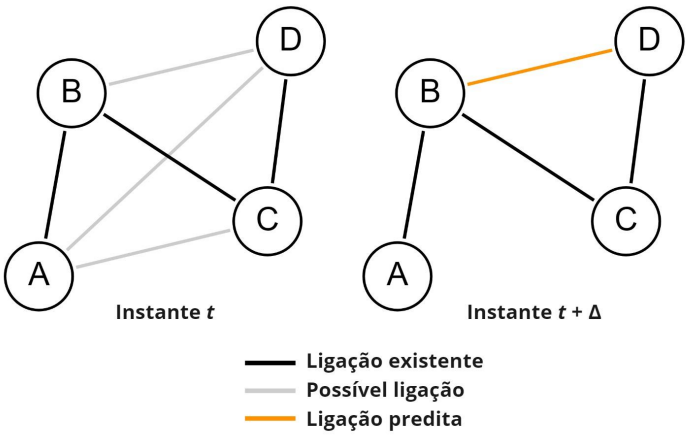
O formalismo matemático da ciência das redes é baseado na teoria de grafo, um subcampo da matemática que surgiu no século XVIII (BARABÁSI, PÓSFAI; 2016). Segundo Caldarelli (2007), um grafo é um modo de codificar uma relação entre elementos de um sistema. Os elementos formam um conjunto V (conjunto de vértices) e as relações entre eles o conjunto A (conjuntos das arestas). Um grafo indicado como G(V, A) pode ser desenhado traçando os vértices como pontos e as arestas como linhas entre eles. Um grafo pode ser classificado em homogêneo ou heterogêneo. Será homogêneo quando tiver apenas um tipo de elemento, ou seja, todos os vértices desempenham a mesma função no grafo. Quando possuir mais que um tipo de elemento, com vértices desempenhando diferentes funções, será considerado heterogêneo.

A topologia de um grafo é a estrutura espacial dos vértices e suas interconexões (arestas). Ou seja, é como o grafo apresenta a estrutura da sua organização. Ela carrega informações implícitas que podem ser específicas de um vértice ou globalmente relevantes para todo o grafo (KASTHURIRATHNA, 2015).

A predição de *links* é uma tarefa aplicada em redes complexas e que utiliza diretamente os conceitos da teoria do grafo. Pode também ser entendida como um processo de previsão de conexões futuras entre pares de vértices desconectados com base em conexões existentes (DAUD *et al.*, 2020). Em uma definição mais formal, Martínez, Berzal e Cubero (2016), explicam que dado um instante t no tempo, de uma rede não direcionada (em que as arestas não possuem direção), o problema de predição de *links* deduz um subconjunto de conexões ausentes no instante atual (conexões existentes, mas não observadas) ou que serão formadas no tempo $t + \Delta$, sendo Δ (delta) uma variação da unidade de tempo.

A tarefa de predição de *links* pode ser vista como um problema de classificação binária onde são consideradas duas classes: positiva ou existência de conexão e negativa ou ausência de conexão (MARTÍNEZ; BERZAL; CUBERO, 2016). A Figura 1 ilustra a tarefa, onde, para fins de exemplo, dentre três possíveis conexões apenas uma foi predita como a mais provável de ocorrer.

Figura 1 - Tarefa de predição de *links*



Fonte: Elaborado pelo autor (2022)

Existem diversos algoritmos que propõem, de maneiras diferentes, executar a tarefa de predição de *links*. Os primeiros algoritmos elaborados consideram heurísticas que tentam capturar e explorar algumas interações estruturais da rede (MALEK et al., 2021). Eles podem

ser considerados métodos baseados em similaridade, que geram uma pontuação para cada par de vértices não conectados. O par de vértices com maior pontuação representa a conexão com mais probabilidade de acontecer. Cada algoritmo possui uma função própria para determinar essa pontuação. Entre eles estão o *Common Neighbours*, *Jaccard Coefficient* e *Adamic/Adar Index*. Existem outros métodos que se baseiam na similaridade global, ou seja, usam informações da topologia inteira da rede para pontuar cada possível conexão. Os métodos de índices quase locais entram como substitutos aos métodos locais e globais, pois levam em consideração tanto atributos locais quanto informações adicionais da topologia de toda a rede (KUMAR *et al.*, 2020; MARTÍNEZ; BERZAL; CUBERO, 2016).

Nos últimos anos, algoritmos baseados em aprendizado de incorporação (do inglês, *learning embeddings*) têm ganhado destaque pela sua eficiência. O objetivo é mapear cada vértice para um vetor de baixa dimensão, de modo que esses vetores representem um resumo da estrutura da rede. Existem duas categorias de algoritmos baseados em aprendizado de incorporação: métodos rasos, como por exemplo, *DeepWalk* e *Node2Vec*, e os métodos profundos, como *GraphSage* e *Graph Attention Networks*. A principal diferença entre essas duas categorias é a eficiência computacional. Embora os métodos rasos possam ser considerados eficientes, os métodos profundos fornecem melhores resultados mas com um custo computacional maior. No entanto, para algumas tarefas, como predição de *links*, os métodos rasos produzem resultados muito bons, evitando assim, necessidade de um poder computacional elevado (MALEK *et al.*, 2021).

A predição de *links* é baseada na evidência empírica de que duas entidades são mais propensas a interagir se forem semelhantes. Porém, a similaridade nas redes deve ser entendida como um conceito abstrato e pode variar entre cada rede. Por isso, um dos desafios dessa tarefa é entender o domínio que a rede representa para definir o conceito de similaridade. Entre outros desafios, estão a necessidade de encontrar um equilíbrio entre a quantidade de informação considerada para realizar a predição e a complexidade do algoritmo utilizado para realizá-la (MARTÍNEZ; BERZAL; CUBERO, 2016).

2.3 REDES NEURAIS

Nos últimos anos, as técnicas de Inteligência Artificial (IA) vêm evoluindo rapidamente, sendo aplicadas nos mais variados cenários reproduzindo características humanas através de computadores, como resolução de problemas, aprendizado, percepção, compreensão e raciocínio (XU *et al.*, 2021). A IA tem como objeto de estudo o conhecimento, e utiliza de técnicas e métodos de diferentes áreas, entre elas, ciência da computação, lógica, biologia, psicologia, filosofia e tantas outras disciplinas. Possui resultados notáveis em aplicações como reconhecimento de fala, processamento de imagem, processamento de linguagem natural e robôs inteligentes (ZHANG; LU, 2021).

Apesar do termo “inteligência artificial” ter sido proposto somente em 1956, por estudiosos da Universidade de Dartmouth, as raízes da IA provavelmente remontam à década de 1940, quando Alan Turing desenvolveu uma máquina para decodificação de mensagens criptografadas para o governo britânico, chamada *The Bombe*. Ela foi considerada o primeiro computador eletromecânico funcional e tinha como objetivo decifrar o código da *Enigma,* máquina de criptografia usada pelo exército alemão na Segunda Guerra Mundial (HAENLEIN; KAPLAN, 2019; ZHANG; LU, 2021).

O desenvolvimento da IA vem promovendo grandes benefícios para a humanidade e conduzindo o desenvolvimento social e econômico para uma nova era, além de transformar fundamentalmente a forma como as empresas tomam decisões (HAENLEIN; KAPLAN, 2019; ZHANG; LU, 2021).

Entre as técnicas típicas e mais relevanntes de IA, encontram-se as Redes Neurais Artificiais (do inglês *Artificial Neural Networks* - ANNs). Esta tem atraído grande atenção devido à sua capacidade de lidar com grandes volumes de dados, mapear relações não lineares e fornecer previsão de resultados (XU *et al.*, 2021). Uma ANN é um sistema computacional não linear inspirado na estrutura, comportamento e habilidades de aprendizado de um cérebro biológico. É uma abstração da rede neural do cérebro humano, que procura simular seu processamento de informações (DHARWAL; KAUR, 2016; WU; FENG, 2018).

A teoria de ANN foi proposta pela primeira vez em 1943, por McCulloch e Pitts. Porém, foi em 1949 que o psicólogo Donald Hebb publicou "A Organização do Comportamento", no qual propôs a hipótese de que a intensidade das conexões sinápticas é variável. Essa hipótese evoluiu para uma lei, chamada de lei de Hebb, afirmando que a força das conexões sinápticas entre os neurônios é variável e que a variabilidade é a base do aprendizado e da memória. Essa lei estabeleceu as bases para a construção de um modelo de rede neural com função de aprendizado. A evolução na área continuou, com destaque para o modelo *Perceptron* proposto por Rosenblatt em 1957, e em 1972 o professor Teuvo Kohonen propôs os Mapas Auto-Organizáveis (WU; FENG, 2018; XU *et al.*, 2021).

As ANNs têm proporcionado um avanço significativo em vários domínios, como o reconhecimento de fala e padrões, previsão climática e diagnóstico de doenças (KAVIANI; SOHN, 2021). Elas também têm sido amplamente aplicadas com o intuito de resolver problemas em diferentes áreas como a agricultura, medicina, finanças, comércio de *commodities*, engenharia, etapas de fabricação e transporte (ABIODUN *et al.*, 2018; XU *et al.*, 2021).

Basicamente, uma ANN consiste em um conjunto de neurônios conectados entre si, de forma unidirecional. Associado a cada neurônio existe uma função de ativação e, cada conexão entre dois neurônios tem um peso atribuído que controla a influência do primeiro para o segundo. Enquanto os neurônios representam as unidades computacionais básicas de uma ANN, as conexões ponderadas entre eles permitem a modelagem de relacionamentos complexos (HAGENAUER; HELBICH, 2022).

Tipicamente, as ANNs são compostas por diferentes camadas de neurônios, onde os neurônios em cada camada estão totalmente conectados aos neurônios da próxima camada. Cada neurônio recebe, como entrada, a saída de todos os neurônios da camada anterior, processa esses valores através de uma função de ativação e envia uma saída para os neurônios da próxima camada. Entre dois neurônios sempre existe um peso que altera o valor transmitido de um para o outro. Este processo acontece em toda a rede até a camada de saída fornecer um resultado (WU; FENG, 2018; KAVIANI; SOHN, 2021; HAGENAUER; HELBICH, 2022).

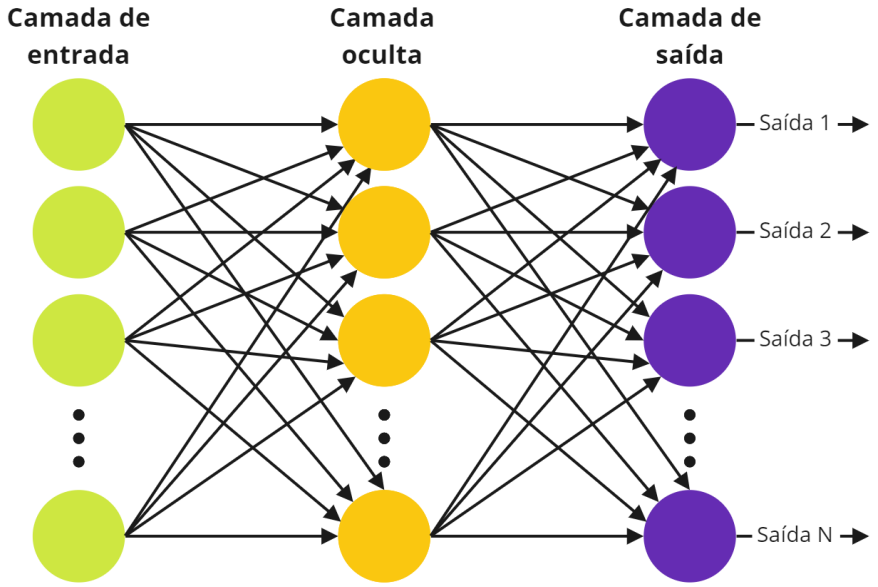
Uma rede neural simples consiste de três camadas, como consta na Figura 2, que são interconectadas para formar a rede (DHARWAL; KAUR, 2016):

1. Camada de entrada: o vetor de recursos do problema é passado para a camada de entrada da rede neural e deve ter os valores na forma numérica. O vetor precisa descrever as informações corretamente e de forma não redundante. Em suma, a camada de entrada recebe informações externas e passa para o processamento posterior.
2. Camada oculta: recebe informações da camada de entrada e realiza seu processamento. O número de camadas ocultas varia de acordo com a natureza e características do problema.
3. Camada de saída: recebe as informações processadas na camada oculta e fornece resultados para serem utilizados em aplicações diversas.

Na etapa de aprendizado, a ANN visa desenvolver a relação que melhor se ajusta à função geral entre os parâmetros de entrada e saída (FURRER; THALER, 2005). Existem

quatro principais tipos de aprendizado. No aprendizado supervisionado, são fornecidos dados de entrada e saída; o objetivo é minimizar a diferença entre as saídas esperadas e as saídas reais previstas pela rede. Para o aprendizado não supervisionado, somente as entradas são fornecidas; a função objetivo é definida apenas com as entradas e os parâmetros da rede. O aprendizado semi-supervisionado pode ser entendido como uma composição do aprendizado supervisionado e não supervisionado. Dados rotulados e dados não rotulados são misturados no processo de aprendizagem. Normalmente a quantidade de dados sem rótulos é muito maior do que a quantidade de dados com rótulos. Por fim, a aprendizagem por reforço é um método voltado à obtenção de recompensas interagindo com o ambiente, julgando a qualidade das ações por níveis de recompensa e, em seguida, treinando o modelo (YANG; WANG, 2020; ZHANG; LU, 2021).

Figura 2 - Representação gráfica de uma RNN simples



Fonte: Elaborado pelo autor (2022)

Nos últimos 10 anos houve um grande progresso no trabalho de pesquisa em redes neurais artificiais. As ANNs possuem capacidade de autoaprendizagem e precisão na modelagem de relações complexas entre dados de entrada e saída sem precisar de fórmulas matemáticas complexas. Seu grande potencial reside no processamento em alta velocidade, sendo este processamento proporcionado por uma implementação paralela massiva. De modo geral, as ANNs potencializam a resolução de problemas práticos que, sem elas, computadores modernos não poderiam resolver (WU; FENG, 2018; ABIODUN et al., 2018; XU et al., 2021).

Para se realizar a tarefa de predição de *links* a partir de um grafo, torna-se necessário extrair informações relevantes contidas nesta estrutura e, então, utilizá-las para modelar uma possível solução de aprendizado. O *Node2Vec* e o *Metapath2Vec,* utilizados neste trabalho, são algoritmos que utilizam ANNs para transformar as informações extraídas de um grafo em vetores que alimentam modelos de aprendizado para predição de *links*.

2.3.1 *Node2Vec*

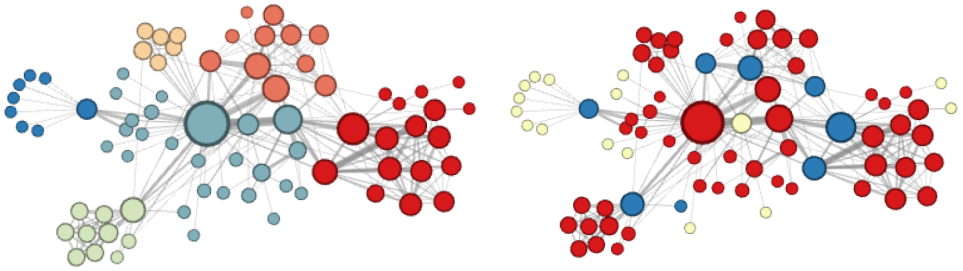
O algoritmo *Node2Vec* aprende as características topológicas de cada vértice a partir de um processo conhecido como *node embedding* (NE), baseado no aprendizado de

incorporação. Esse processo mapeia cada vértice do grafo para um espaço vetorial, distribuído e contínuo, de baixa dimensão, onde cada vértice é transformado em um vetor representativo, ou seja, um vetor que contém as características daquele vértice. As relações geométricas no espaço vetorial refletem as propriedades e estrutura do grafo original. Os vetores são então usados como entrada para algoritmos de ML para extrair informações úteis para a construção de classificadores ou preditores. É importante destacar que o *Node2Vec* não distingue grafos homogêneos e heterogêneos (PENG *et al*., 2020; AMARA; TAIEB; AOUICHA, 2021).

Para gerar os vetores de cada vértice, o algoritmo do *Node2Vec* foi inspirado no modelo de ANN *Skip-gram*, que procura aprender vetores para palavras com o intuito de preservar seu contexto. O *Skip-gram* varre as palavras de um documento e, para cada palavra, gera um vetor de modo que ele possa prever palavras próximas. O objetivo do *Skip-gram* é baseado na hipótese distribucional que afirma que palavras semelhantes tendem a aparecer em grupos semelhantes. Inspirado neste modelo, o *Node2Vec* estabelece uma analogia ao representar um grafo como um "documento". Da mesma forma que um documento textual é uma sequência ordenada de palavras, pode-se gerar um documento de um grafo ordenando sequências de vértices. Ao contrário de um documento textual, a natureza de um grafo não é linear, portanto, existem muitas estratégias de ordenamento para vértices que resultam em diferentes representações vetoriais (GROVER; LESKOVEC; 2016).

De modo geral, para gerar um "documento" de um grafo, o *Node2Vec* propõe um procedimento de percurso aleatório tendencioso de 2ª ordem, que garante flexibilidade na estratégia de ordenamento dos vértices. Existem duas principais estratégias de ordenamento dos vértices: amostragem em largura (do inglês *Breadth-first Sampling* - BFS) e a amostragem em profundidade (do inglês *Depth-first Sampling* - DFS). A estratégia BFS percorre vértices próximos ao vértice fonte, mapeando uma amostragem de similaridade de equivalência estrutural, ou seja, vértices que possuem um papel estrutural semelhante estarão localmente próximos no espaço vetorial. Por outro lado, a estratégia DFS percorre vértices sequencialmente com distâncias crescentes do vértice fonte, mapeando uma amostragem de similaridade de homofilia, ou seja, vértices que possuem alto grau de conexão e pertencem ao mesmo grupo ou comunidade dentro do grafo, estarão localmente próximos no espaço vetorial (GROVER; LESKOVEC; 2016).

Figura 3 - Estratégia DFS e BFS



(a) (b)

Fonte: Grover e Leskovec (2016)

O grafo da Figura 3a, ilustra a estratégia DFS, no qual os grupos/comunidades identificados pelo algoritmo *Node2Vec* são distintos pelas cores, apresentando uma similaridade de homofilia. Os vértices da mesma cor possuem alto grau de conexão se comparado ao resto do grafo. Já o grafo da Figura 3b ilustra a estratégia BFS, onde os vértices da mesma cor possuem papéis estruturais parecidos, ou seja, similaridade de equivalência estrutural. Nota-se que, na Figura 3b, os vértices azuis têm um papel de ponte entre

subgrupos, enquanto os vértices amarelos estão na periferia com uma função mais limitada no grafo (GROVER; LESKOVEC; 2016).

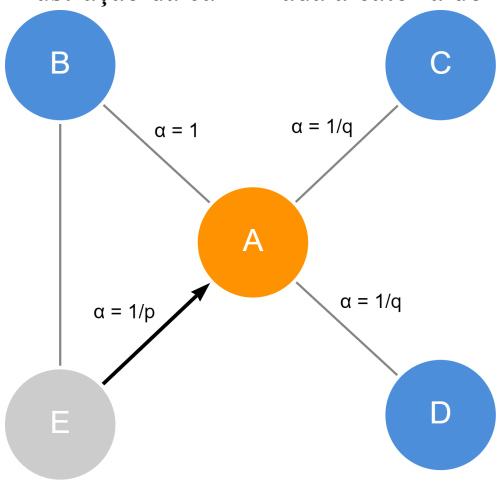
Um percurso aleatório de 1ª ordem leva em consideração apenas o estado atual, gerando uma probabilidade igualmente distribuída entre todos os vértices conectados ao vértice atual do percurso. A única influência que pode alterar essa probabilidade são os pesos atrelados às arestas conectadas ao vértice atual. Já o percurso aleatório tendencioso de 2ª ordem do *Node2Vec,* leva em consideração o estado atual e o anterior, ou seja, o algoritmo também sofre influência da etapa anterior do percurso na decisão de qual será o próximo vértice da amostragem (BRATANIC, 2021).

Segundo Grover e Leskovec (2016), o percurso aleatório tendencioso de 2ª ordem depende de dois parâmetros p e q que são responsáveis por guiar o percurso. Considere a Figura 4 e que o algoritmo recém atravessou a aresta (E, A) e agora se encontra no vértice A. O vértice X a ser selecionado para a próxima etapa do percurso é determinado a partir do maior valor de α, definido pela Equação 1, sendo $d_{E,X}$ a distância entre os vértices E e X.

$$\alpha(E,X) = \begin{cases} \frac{1}{p} \; se \; d_{E,X} = 0 \\ 1 \; se \; d_{E,X} = 1 \\ \frac{1}{q} \; se \; d_{E,X} = 2 \end{cases} \tag{1}$$

A probabilidade de retornar ao vértice E é controlada pelo parâmetro de retorno p. Um valor alto para p garante menores chances de revisitar o vértice anterior e incentiva uma exploração moderada do grafo, enquanto que um valor baixo para p levaria o percurso a retroceder uma etapa mantendo a exploração mais próxima ao vértice fonte. Quando o parâmetro $q > 1$ o percurso é enviesado para os vértices próximos ao vértice fonte, gerando um comportamento de BFS. Em contraste, se $q < 1$, a caminhada é mais inclinada a um comportamento DFS, pois se distancia do vértice fonte (GROVER; LESKOVEC; 2016). Dessa forma, o comportamento do percurso pode variar entre as estratégias BFS e DFS de acordo com os valores estipulados nos parâmetros p e q.

Figura 4 - Ilustração da caminhada aleatória do *Node2Vec*



Fonte: Elaborado pelo autor (2022)

Para realizar a tarefa de predição de *links*, é preciso gerar vetores que correspondem às conexões, ou seja, vetores das arestas. Para isso, se utiliza um operador binário entre vetores de dois vértices, resultando no vetor da aresta correspondente. Grover e Leskovec (2016) consideram quatro possíveis escolhas de operadores binários: *Average* definido como $\frac{u+v}{2}$,

Hadamard definido como $u * v$, *Weighted-L1* definido como $u - v$ e *Weighted-L2* definido como $(u - v)^2$, sendo u e v vetores representativos de vértices pertencentes ao grafo. O resultado é um vetor representativo da aresta entre *u* e *v*.

2.3.2 *Metapath2Vec*

O algoritmo *Metapath2Vec*, similar ao *Node2Vec*, também realiza o processo de NE para mapear um vetor representativo para cada vértice de um grafo em um espaço vetorial. Sua principal distinção é o percurso aleatório, que busca maximizar a probabilidade de preservação tanto da estrutura topológica quanto da semântica de um grafo com mais de um tipo de vértice, ou seja, um grafo heterogêneo. Outros algoritmos de NE, como *Node2Vec*, tratam vértices diferentes da mesma forma, gerando representações vetoriais que não distinguem vértices heterogêneos.

O *Metapath2Vec* utiliza o mesmo modelo de ANN *skip-gram* que o *Node2Vec*. Para incorporar a estrutura heterogênea no *skip-gram*, é proposto um percurso aleatório baseado em *meta-path*. Formalmente, Dong, Chawla e Swami (2017), estabeleceram *meta-path* como sendo um esquema P definido como um caminho denotado na forma $V_1 \xrightarrow{R_1} V_2 \xrightarrow{R_2} \cdots \xrightarrow{R_{l-1}} V_l$, onde *l* é o tamanho do esquema e *R* define a relação composta entre os vértices V_1 e V_l, sendo $R = R_1 \circ R_2 \circ \cdots \circ R_{l-1}$. Assim, o fluxo do percurso aleatório está pré-definido a partir do esquema *P*, que restringe as possibilidades de escolha do próximo vértice do percurso. A probabilidade dessa escolha está proporcionalmente distribuída apenas entre os vértices vizinhos do vértice atual do percurso, que possuem o tipo de conexão denotado em *P* para a etapa subsequente do percurso. Para exemplificar, considerando V_2 o vértice atual do percurso, apenas os vértices que possuem a relação R_3 com V_2, têm uma probabilidade de serem escolhidos como o vértice V_3 do percurso.

No *Metapath2Vec* é possível utilizar os mesmos operadores binários do *Node2Vec* entre vetores dos vértices para gerar os vetores das arestas. Com isso, pode-se alimentar um modelo de aprendizado para realizar a tarefa de predição de *links*.

3 TRABALHOS CORRELATOS

Através da busca na literatura científica foram selecionados trabalhos com a proposta de otimização da gestão de *leads* com o uso de IA, especialmente técnicas de classificação de *leads* e previsão de *leads* e/ou vendas. A pesquisa foi realizada nas bases de artigos acadêmicos Scopus®, Web of Science®, ScienceDirect®, IEEE Xplore® ACM Digital Library® e SpringerLink®. A chave de pesquisa utilizada foi: "lead management" AND "customer*" AND ("prediction" OR "link prediction" OR "neural network*" OR "forecasting" OR "recommendation*" OR "recommender"), em que o "*" indica as variações do termo.

Quadro 1 - Resultado da revisão da literatura

Base de artigos acadêmicos	Quantidade de trabalhos resultantes
Scopus®	2
Web of Science®	1
ScienceDirect®	68
IEEE Xplore®	0
ACM Digital Library®	5
SpringerLink®	155
Total de resultados	231

Fonte: Elaborado pelo autor (2022).

O Quadro 1 apresenta a síntese dos resultados obtidos pela expressão de busca. Do total de 231 trabalhos publicados em periódicos ou congressos, 30 apresentaram título adequado ao objetivo da pesquisa. Seus resumos foram então lidos na íntegra e 11 artigos foram selecionados para a leitura da introdução. Por fim, foram selecionados 4 trabalhos que possuem relação com o tema desta pesquisa e que passaram por todos os critérios de exclusão. Devido a quantidade de trabalhos encontrados dentro do escopo proposto, foi realizada uma busca na literatura cinzenta através do Google Scholar® com a mesma expressão de busca. Cerca de 15 artigos foram selecionados para a leitura do resumo e, destes, 2 foram selecionados, totalizando 6 trabalhos.

No trabalho de Espadinha-Cruz, Fernandes e Grilo (2021) é proposta uma metodologia que visa melhorar a eficiência nos diferentes estágios de gestão de *leads* no setor de telecomunicação, entre eles: estimar a probabilidade da conversão do *lead*, monitorar o gerenciamento de *leads* ao longo do seu ciclo de vida e apoiar a tomada de decisão na segmentação de *leads*. Para isso, foram utilizadas técnicas de mineração de dados (do inglês, *Data Mining* - DM), aplicando a metodologia SEMMA que propõem as etapas de exploração de dados, processamento de dados, modelagem, comparação e avaliação dos resultados. Foi concluído que a metodologia proposta permite aumentar significativamente a eficiência e rentabilidade das vendas por *telemarketing*.

Li e Xu (2022) propuseram um CRM orientado por IA, como uma solução técnica que pode auxiliar empresas a maximizar seu desenvolvimento econômico. O artigo contribui, segundo os autores, na previsão da participação de mercado e previsão do planejamento de metas e precificação de produtos. Também foi utilizada uma ANN para antecipar o volume de vendas maximizando o lucro do cliente do CRM. A partir deste estudo, foi descoberto que as tecnologias baseadas em IA, incorporadas a um CRM, proporcionam uma experiência agradável ao usuário. Como resultado, os clientes do CRM permaneceram fiéis à utilização do mesmo, inclusive recomendando a outras pessoas.

Para capturar o impacto de sistemas de gestão de *leads*, Ohiomah *et al.* (2019) validaram empiricamente um modelo conceitual baseado na teoria *Technology-Task-Fit* utilizando os seguintes mediadores: características da tarefa (quantidade de chamadas e intensidade de acompanhamento de *leads*), comportamento de venda (venda adaptativa) e características do vendedor (habilidades técnicas e de vendas). Comparam também sistemas que organizam os *leads* em fila (utilizam regras de negócio pré-definidas) ou em listas (que não fornecem nenhum tipo de filtro). Foi descoberto que o uso efetivo de um sistema de gestão de *leads* aumenta o desempenho de acompanhamento de *leads* e consequentemente das vendas. Além disso, sistemas que organizam os *leads* em fila obtêm melhores resultados em vendas comparado aos baseados em lista.

No trabalho de Munoz *et al.* (2021), é abordado o desafio de identificar clientes ideais para prospecção de empréstimos. Usando técnicas de aprendizado de máquina e aprendizado profundo, classificadores são construídos para resolver duas tarefas de previsão: a intenção de empréstimo e a aprovação de empréstimo. A probabilidade de aprovação do empréstimo é modelada para melhorar a qualidade do *lead*. No geral, para ambas as abordagens, o desempenho da classificação para prever futuros solicitantes de empréstimos dentro do conjunto de testes foi reduzido. O estudo foi limitado a dados relativos ao processo de solicitação de reembolso. A sugestão é que em pesquisas futuras também sejam utilizados dados históricos de reembolso de clientes.

O objetivo da pesquisa de Ogwueleka *et al.* (2015) é estabelecer um modelo de ANN de classificação e agrupamento, a partir de dados do setor bancário, para identificar padrões anormais (comportamento de clientes que vão sair) ou normais (comportamento de clientes lucrativos que ficarão). O modelo foi aplicado no *Intercontinental Bank Plc*® para apoiar o CRM e o planejamento. Os resultados mostram que melhorias significativas foram obtidas na retenção e satisfação do cliente, redução de custos, saldo do cliente, reavivamento do cliente e descoberta de novos clientes.

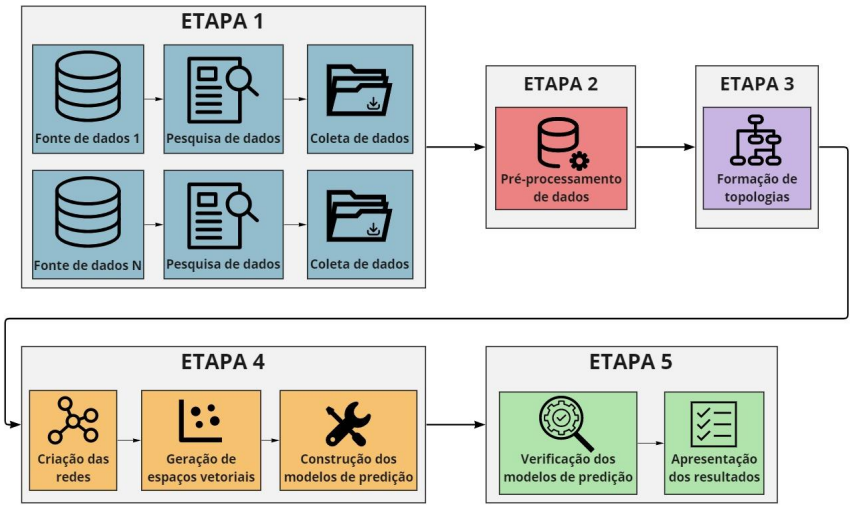
O trabalho de pesquisa elaborado por Eitle e Buxmann (2019) visa analisar a previsão dos cenários de *pipeline* de vendas: conversão de *lead* para oportunidades e oportunidades para venda. Para tal, foram usadas técnicas de classificação de ML como Random Forest, Support Vector Machine, XGBoost e CatBoost. Os resultados considerando precisão e AUC (Area Under The Curve) mostram que o *CatBoost* supera claramente os outros algoritmos. Mas a previsão da probabilidade de vendas no estágio inicial de *lead* é melhor realizada pela *Random Forest*, cujos resultados superam significativamente *Support Vector Machine*, *XGBoost*, *CatBoost*.

A busca na literatura acadêmica retornou poucos trabalhos que refletem o tema dessa pesquisa. Aqueles apresentados acima, revelam que, além de metodologias de otimização, técnicas de ML vêm sendo frequentemente utilizadas para aprimorar a gestão de *leads*. Nenhum trabalho, que trate especificamente sobre estrutura de redes para a tarefa de predição de *links* dentro da área de gestão de *leads*, foi encontrado. Pode-se concluir que, apesar de já existir uma quantidade significativa de pesquisa sobre IA dentro das áreas que compõem o CRM, o uso de grafos ainda não foi explorado, demonstrando que tarefas promissoras, como a predição de *links*, possuem potencial de inovação na área de gestão de *leads*.

4 MÉTODO PROPOSTO

Este trabalho apresenta um método de análise da gestão de *leads* voltado à previsão de interesse de *leads* existentes por novas edições de produtos de uma determinada organização. O intuito é apoiar a otimização de ativação de *leads* pela equipe de vendas. Para alcançar este objetivo, utilizam-se técnicas de aprendizado de representação de espaços vetoriais de redes heterogêneas e predição de *links* no contexto da gestão de *leads*. A Figura 5 fornece uma visão geral do método e das 5 etapas que o compõem: 1) Pesquisa e coleta de dados; 2) Pré-processamento de dados; 3) Formação de topologias; 4) Construção dos modelos de predição; e 5) Verificação e apresentação dos resultados. As primeiras três etapas são responsáveis pela preparação dos dados, enquanto as demais são responsáveis pela elaboração e análise do modelo de predição. A seguir, cada uma das etapas é descrita em detalhes.

Figura 5 - Etapas gerais do método proposto



Fonte: Elaborado pelo autor (2022)

4.1 ETAPA 1: PESQUISA E COLETA DOS DADOS

Na primeira etapa, é realizada uma pesquisa em todas as bases de dados de uma determinada organização sobre o histórico referente aos *leads* (como o *e-mail*, valor gasto com a organização e edição de produtos com registro de interesse) e informações dos produtos (como categoria do produto, número de edições e ano de cada edição). Sendo assim, os dados disponíveis na organização a partir de múltiplas fontes são coletados e armazenados.

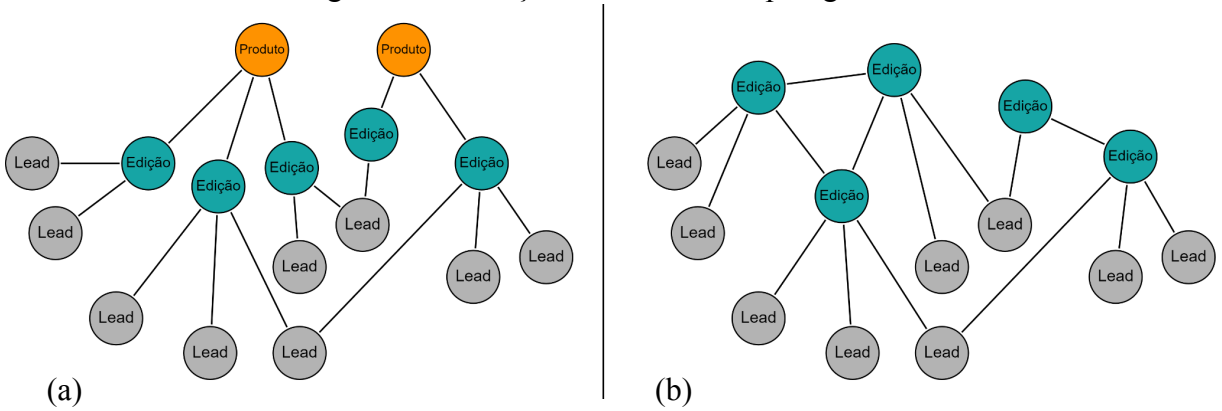
4.2 ETAPA 2: PRÉ-PROCESSAMENTO DOS DADOS

Na segunda etapa, todos os dados coletados são analisados e pré-processados para que apenas as informações úteis à formação de topologias sejam mantidas. O resultado é um conjunto de dados pré-processados voltados à formação de topologias, como a identificação do *lead*, as edições de produtos que o *lead* se interessou, o ano de cada edição e a categoria dos produtos. Vale mencionar que os dados a serem utilizados na etapa seguinte podem variar dependendo das características e do objetivo de determinado estudo.

4.3 ETAPA 3: FORMAÇÃO DE TOPOLOGIAS

Nesta etapa, diversas formações de relações entre os dados pré-processados são elaboradas, resultando em várias possíveis estruturas topológicas. Cada estrutura topológica gera uma rede distinta sobre os mesmos dados. Para a elaboração das topologias não foi utilizado nenhum método específico, mas um processo de abstração e entendimento dos dados coletados e como eles podem se relacionar. Para exemplificar, a Figura 6 ilustra duas topologias diferentes. A Figura 6a relaciona produto com edição do produto e interesse do *lead* com alguma edição. Por outro lado, a topologia da Figura 6b estabelece relações entre as próprias edições de um produto e o interesse de um *lead* com alguma edição. De modo geral, a caracterização de cada vértice depende do estudo que será realizado no contexto da gestão de *leads*.

Figura 6 - Ilustração de diferentes topologias



Fonte: Elaborado pelo autor (2022).

4.4 ETAPA 4: CONSTRUÇÃO DOS MODELOS DE PREDIÇÃO

Na sequência, a etapa quatro prepara uma rede para cada estrutura topológica. Então é selecionado a edição mais recente, entre todos os produtos presentes em cada rede, e removido todas suas conexões com qualquer *lead*. Os *leads* que foram desvinculados da edição mais recente e permanecerem na rede devido à conexão com alguma outra edição, devem ser identificados para serem utilizados na etapa de verificação.

A partir de algoritmos de NE, são mapeadas diferentes representações vetoriais para cada rede. Por último, os modelos de predição de *links* são gerados com algoritmos de ML alimentados pelos vetores de representação das conexões (arestas), obtidos através das representações vetoriais dos vértices..

4.5 ETAPA 5: VERIFICAÇÃO E APRESENTAÇÃO DOS RESULTADOS

Por fim, nesta última etapa os vetores que representam conexões entre cada *lead* e a edição mais recente selecionada na etapa 4, são passados para os modelos de ML treinados, com o intuito de promover um entendimento dessas conexões. Como resultado, obtém-se as probabilidades das conexões entre cada *lead* e a edição selecionada. Então, é verificada as conexões entre a edição e seus verdadeiros *leads* identificados na etapa 4, para obter-se uma acurácia de cada modelo de predição.

5 RESULTADOS EXPERIMENTAIS

5.1 APRESENTAÇÃO DO CENÁRIO DE ESTUDO

Para demonstrar a viabilidade do método proposto, um cenário de estudo foi elaborado. O conjunto de dados foi provido pela Base Colaborativa®, uma organização sem fins lucrativos que arrecada recursos vendendo determinados produtos, neste caso, cursos e viagens de desenvolvimento pessoal. Esse conjunto conta com 5.046 *leads*, 3 cursos que juntos somam 50 edições realizadas entre 2016 e 2021.

Em média, a frequência de interesse de um *lead* por alguma edição é de 1,33. Essa frequência demonstra que existe um comportamento de interesse que se repete através das 50 edições dos cursos, possibilitando a predição de *link* entre algum *lead* já existente no conjunto de dados e uma nova edição de algum curso.

Para a análise do método proposto, na etapa de treinamento dos modelos, foram removidos todos os registros de interesses pela edição mais recente das 50 existentes. Assim, esta última edição foi utilizada para verificar a capacidade dos modelos na realização de predições corretas sobre quais *leads* poderiam ter interesse ou não.

5.2 INSTANCIAÇÃO DO MÉTODO PROPOSTO

Esta seção objetiva detalhar o método proposto apresentando os componentes tecnológicos utilizados e como estes se interconectam nas etapas, de maneira que ao final seja possível realizar a predição de *links*.

5.2.1 Etapa 1

Para constituir o conjunto de dados utilizado no cenário de estudo, foi necessário realizar uma busca por duas fontes de dados que a organização utiliza: Google Drive® e RD Station Marketing®. Os dados foram encontrados sem uma forma padronizada de estrutura e armazenamento. Caso tivessem relação com as edições dos cursos e seus respectivos *leads*, os dados eram coletados e armazenados.

5.2.2 Etapa 2

Os dados coletados foram analisados, filtrados e organizados, mantendo apenas os que continham informações úteis para a geração de grafos com diferentes estruturas topológicas. O resultado produziu uma planilha única composta por diferentes campos, entre eles *e-mail* do *lead*, edição de interesse do *lead*, ano da edição e curso da edição. Ao todo, foram obtidas 6731 linhas, onde cada linha representa o registro do interesse de algum *lead* por alguma edição de algum curso.

5.2.3 Etapa 3

A partir dos dados pré-processados, foram criadas 5 topologias diferentes, conforme consta no Quadro 2. As colunas representam os tipos de relações estipuladas para cada topologia, entre os tipos de dados coletados. No total foram utilizados 4 diferentes tipos de relações, sendo que cada topologia possui no máximo 3. Os 4 tipos de relações são:

1. *lead*-edição: representa o interesse de um *lead* por alguma edição de algum curso;
2. edição-curso: representa qual curso aquela edição pertence;
3. edição-edição: todas as edições do mesmo curso estão conectadas entre si;
4. edição-ano: representa qual ano aquela edição foi realizada.

Quadro 2 - Relações das topologias

Tipologia	Relação 1	Relação 2	Relação 3
Topologia 1	*lead*-edição		
Topologia 2	*lead*-edição	edição-curso	
Topologia 3	*lead*-edição	edição-edição	
Topologia 4	*lead*-edição	edição-edição	edição-ano
Topologia 5	*lead*-edição	edição-ano	

Fonte: Elaborado pelo autor (2022).

5.2.4 Etapa 4

Para a construção da rede de cada topologia, utilizou-se a biblioteca NetworkX® da linguagem de programação Python®. A edição mais recente entre os cursos foi identificada, e todas as 49 relações do tipo *lead*-edição referentes a edição em questão foram removidas. Os *leads* que não possuíam nenhuma relação do tipo *lead*-edição com qualquer outra edição foram removidos da rede, restando 5018 *leads* do total de 5046. Devido à conexão com outras edições, se mantiveram 21 *leads* dos 49. Estes, foram identificados para serem utilizados na etapa de verificação (etapa 5). Formou-se então, a partir da rede em questão, o conjunto de amostras positivas e negativas. Todas as conexões presentes na rede foram utilizadas no conjunto de amostras positivas, enquanto que, para o conjunto de amostras negativas foram selecionadas, de forma aleatória, conexões não existentes da rede. Nenhuma amostra negativa continha conexão com a edição mais recente, para não enviesar os modelos.

Para realizar o aprendizado de NE, ou seja, as representações vetoriais da rede de cada topologia, foi aplicado o algoritmo *Node2Vec* e o algoritmo *Metapath2Vec*, com a finalidade de comparação entre os mesmos. Para a aplicar *Node2Vec* foi utilizada a biblioteca de mesmo nome do algoritmo, enquanto que para aplicação do *Metapath2Vec* foi utilizado a biblioteca StellarGraph®. Conforme consta no Quadro 3, foram elaboradas três representações vetoriais para a rede de cada topologia. A coluna "Parâmetros" apresenta os valores de p e q para o algoritmo *Node2Vec*, assim como os percursos definidos para o algoritmo *Metapath2Vec*.

Conforme mencionado por Grover e Leskovec (2016), é preciso utilizar um operador binário entre os vetores dos vértices para gerar as representações vetoriais das arestas dos grafos. Dessa forma, são gerados os vetores das conexões contidas na rede de cada topologia. Todos os operadores binários foram testados, porém, neste documento foi considerado apenas o operador *Weighted-L1* definido como $|u - v|$, sendo u e v representações vetoriais de vértices pertencentes ao grafo.

A partir disso, dois modelos de predição de *links* foram gerados para cada representação vetorial das redes utilizando os algoritmos de ML *Random Forest* e *Logistic Regression*, por meio da biblioteca Scikit-learn®. Na etapa de treinamento, como dados de entrada para os modelos, todas as conexões existentes na rede são utilizadas como amostras positivas. São escolhidas, de forma aleatória, amostras negativas na mesma quantidade que as positivas. Como mencionado anteriormente, nenhuma amostra negativa está relacionada à edição que será usada para a etapa de verificação.

Quadro 3 - Representações vetoriais

Topologia	**Representação vetorial**	**Algoritmo**	**Parâmetros**	
Topologia 1	Representação vetorial 1	*Node2Vec*	p=1	q=2
	Representação vetorial 2	*Node2Vec*	p=1	q=0,5
	Representação vetorial 3	*Metapath2Vec*	*lead*, edição, *lead*	
Topologia 2	Representação vetorial 1	*Node2Vec*	p=1	q=2
	Representação vetorial 2	*Node2Vec*	p=1	q=0,5
	Representação vetorial 3	*Metapath2Vec*	*lead*, edição, curso, edição, *lead*	
Topologia 3	Representação vetorial 1	*Node2Vec*	p=1	q=2
	Representação vetorial 2	*Node2Vec*	p=1	q=0,5
	Representação vetorial 3	*Metapath2Vec*	*lead*, edição, *lead* lead, edição, edição, lead	

Topologia 4	Representação vetorial 1	*Node2Vec*	*p*=1	*q*=2
	Representação vetorial 2	*Node2Vec*	*p*=1	*q*=0,5
	Representação vetorial 3	*Metapath2Vec*	*lead*, edição, ano, edição, *lead* *lead*, edição, edição, ano, edição, *lead*	
Topologia 5	Representação vetorial 1	*Node2Vec*	*p*=1	*q*=2
	Representação vetorial 2	*Node2Vec*	*p*=1	*q*=0,5
	Representação vetorial 3	*Metapath2Vec*	*lead*, edição, ano, edição, *lead*	

Fonte: Elaborado pelo autor (2022).

5.2.5 Etapa 5

Por fim, na etapa de verificação, para cada modelo de predição construído foi utilizado um conjunto de dados contendo os vetores de conexão entre todos os *leads* e a edição mais recente, retratando a relação do tipo *lead*-edição. A partir dos *leads* identificados na etapa 4, foi verificada a quantidade de *leads* indicados como prováveis conexões e a taxa de predições corretas.

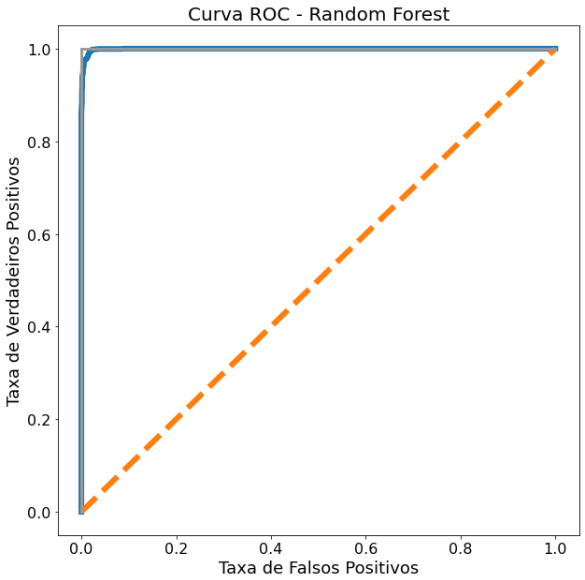
5.3 ANÁLISE E DISCUSSÃO DOS RESULTADOS

Esta seção foi elaborada com o objetivo de verificar os resultados obtidos através do método proposto considerando o cenário de estudo apresentado na subseção 5.1. Neste sentido, discute-se sobre o desempenho do modelo de predição e a viabilidade de aplicação do método proposto.

A métrica empregada para avaliar o modelo de predição foi a Área Sob a Curva de Característica de Operação do Receptor (do inglês *Area Under the Receiver Operating Characteristic Curve* - ROC-AUC). Esta métrica é amplamente utilizada como indicador de desempenho para problemas de predição de *links* e de classificação binária (LEE *et al.*, 2021). A métrica ROC-AUC calcula a área sob a curva ROC, uma curva de probabilidades que apresenta a taxa de verdadeiros positivos contra a taxa de falsos positivos em diferentes limiares de classificação. De modo geral, a ROC-AUC simplifica a análise da curva ROC ao agregar todos seus limiares e resumi-la a um único valor. Basicamente, quanto mais próximo de 1 a ROC-AUC estiver, melhor o desempenho do classificador ao distinguir entre as conexões que ocorreram e as que não.

O Quadro 5 (Apêndice A), contém o valor da ROC-AUC obtido para todos os modelos de predição treinados. Percebe-se que todos os modelos se comportam de maneira desejada, considerando que o menor valor de ROC-AUC obtido foi 0,873 e o maior foi 0,998. Para exemplificar graficamente, a Figura 7 exibe a curva ROC do modelo treinado a partir da Topologia 2, com a representação vetorial 3 utilizando o algoritmo de ML *Random Forest*, com uma ROC-AUC de 0,998. A linha azul se refere ao modelo de predição, enquanto que a linha laranja é a linha base que representa um classificador incapaz de distinguir as classes positivas e negativas, com uma ROC-AUC de 0,50.

Figura 7 - Curva ROC do modelo de predição treinado a partir da Topologia 2, com a representação vetorial 3 utilizando o algoritmo de ML *Random Forest*



Fonte: Elaborado pelo autor (2022).

Após a avaliação inicial dos modelos de predição, iniciou-se a avaliação do método proposto a partir do cenário de estudo, com o objetivo de determinar seu desempenho real. Para isto, os resultados foram sumarizados no Quadro 4. As últimas 4 colunas da tabela são explicadas a seguir:

- Predições corretas: considera os 21 *leads* identificados na etapa 4, como sendo as predições de conexões esperadas para a edição mais recente;
- Taxa de predições corretas: porcentagem de predições corretas referente aos 21 *leads*, ou seja, $\frac{Predições\ corretas}{21}$;
- Todas as predições: apresenta o número total de predições entre todos os *leads* e a edição mais recente;
- Taxa de todas as predições: porcentagem de todas as predições referente aos 5018 *leads* totais, ou seja, $\frac{Todas\ as\ predições}{5018}$.

Quadro 4 - Resultados obtidos a partir das predições realizadas no cenário de estudo

Topologia	**Representação vetorial**	**Algoritmo de ML**	**Predições corretas**	**Taxa de predições corretas**	**Todas as predições**	**Taxa de todas as predições**
Topologia 1	Representação vetorial 1	*Random Forest*	0	0,00%	12	0,24%
		Logistic Regression	0	0,00%	286	5,70%
	Representação vetorial 2	*Random Forest*	1	4,76%	16	0,32%
		Logistic Regression	2	9,52%	266	5,30%
	Representação vetorial 3	*Random Forest*	0	0,00%	0	0,00%
		Logistic Regression	0	0,00%	0	0,00%
Topologia 2	Representação vetorial 1	*Random Forest*	1	4,76%	1296	25,83%

		Logistic Regression	7	33,33%	3305	65,86%
	Representação vetorial 2	*Random Forest*	1	4,76%	489	9,74%
		Logistic Regression	18	85,71%	3998	79,67%
	Representação vetorial 3	*Random Forest*	1	4,76%	130	2,59%
		Logistic Regression	0	0,00%	93	1,85%
Topologia 3	Representação vetorial 1	*Random Forest*	20	95,24%	4978	99,20%
		Logistic Regression	21	100,00%	4901	97,67%
	Representação vetorial 2	*Random Forest*	21	100,00%	4878	97,21%
		Logistic Regression	20	95,24%	4846	96,57%
	Representação vetorial 3	*Random Forest*	2	9,52%	577	11,50%
		Logistic Regression	4	19,05%	733	14,61%
Topologia 4	Representação vetorial 1	*Random Forest*	20	95,24%	4993	99,50%
		Logistic Regression	20	95,24%	4989	99,42%
	Representação vetorial 2	*Random Forest*	18	85,71%	4360	86,89%
		Logistic Regression	21	100,00%	4920	98,05%
	Representação vetorial 3	*Random Forest*	0	0,00%	19	0,38%
		Logistic Regression	0	0,00%	119	2,37%
Topologia 5	Representação vetorial 1	*Random Forest*	0	0,00%	144	2,87%
		Logistic Regression	11	52,38%	2903	57,85%
	Representação vetorial 2	*Random Forest*	1	4,76%	122	2,43%
		Logistic Regression	2	9,52%	1495	29,79%
	Representação vetorial 3	*Random Forest*	6	28,57%	526	10,48%
		Logistic Regression	7	33,33%	383	7,63%

Fonte: Elaborado pelo autor (2022).

Analisando o Quadro 4 pode-se perceber que a topologia, a representação vetorial e o algoritmo de ML influenciam significativamente nos resultados dos modelos de predição. Isto fica evidente quando a representação vetorial 2, que utiliza o algoritmo *Node2Vec*, é isolada. As topologias 1 e 5 apresentaram uma baixa taxa de acerto nas predições corretas. Na topologia 2 houve uma variação significativa na taxa de predições corretas entre os algoritmos de ML. Enquanto que, nas topologias 3 e 4, as taxas de predições corretas ficaram próximas de 100%.

Porém, a taxa de todas as predições deve também ser considerada para qualificar os modelos. Numa análise ampla, percebe-se que, quanto maior a taxa de predições corretas, maior a taxa de todas as predições, sinalizando um comportamento generalista do modelo. Por isso, para qualificar os modelos, criou-se um coeficiente de generalização definido pela Equação 2, sendo que, quanto mais próximo de 1 for o valor do coeficiente γ, menos generalista e mais qualificado é o modelo.

$$\gamma = \frac{predições\ corretas}{todas\ as\ predições} \quad (2)$$

Dessa forma, os 3 melhores modelos classificados, a partir do coeficiente de generalização γ, constam na Tabela 1 em ordem decrescente.

Tabela 1 - Melhores modelos de predição

Topologia	Representação vetorial	Algoritmo de ML	Predições corretas	Taxa de predições corretas	Todas as predições	Taxa de todas as predições	γ
Topologia 1	Representação vetorial 2	Random Forest	1	4,76%	16	0,32%	0,062
Topologia 5	Representação vetorial 3	Logistic Regression	7	33,33%	383	7,63%	0,018
Topologia 5	Representação vetorial 3	Random Forest	6	28,57%	526	10,48%	0,011

Fonte: Elaborado pelo autor (2022).

Apesar do algoritmo de NE, *Node2Vec*, não considerar a heterogeneidade das redes, a representação vetorial do modelo melhor classificado foi gerada com o *Node2Vec*. No segundo e terceiro melhores modelos classificados foi utilizado o *Metapath2Vec*, que conta com um caminho pré-estabelecido a partir dos tipos de nodos existentes no grafo, ou seja, considera as características heterogêneas da rede na representação vetorial. Independente do algoritmo utilizado, nota-se que os valores de γ estão muito abaixo de 1, valor ideal para o coeficiente. Assim, considerando a aplicação do método com os dados disponibilizados no cenário de estudo, entende-se que os modelos gerados tiveram um comportamento muito generalista, com uma alta taxa de erro nas predições de *links*. Ainda assim, ao analisar individualmente as muitas predições, os resultados são consistentes e demonstram potencial para subsidiar um processo de gestão de *leads*.

Apesar do conjunto dos fatores (qualidade dos dados, topologias empregadas e algoritmos de ML utilizados) terem influenciado nos valores de γ, um fator que se destaca na discussão é a complexidade do aprendizado de representação vetorial das redes. Quando as redes possuem estruturas heterogêneas e comportamento dinâmico (quando a rede se modifica com o tempo), existe uma dificuldade maior para aprender representações vetoriais que considerem as informações topológicas e de semântica de forma adequada.

6 CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

Este trabalho apresentou um método baseado em predição de *links* voltado à gestão de *leads*, para desempenhar a tarefa de prever possíveis interesses de *leads* em novos produtos ou novas edições de produtos de uma organização. Com este objetivo, através de diferentes formações topológicas, o trabalho empregou aprendizado de representação de rede, baseado em redes neurais rasas e predições de *links*. Dessa forma, foi possível efetuar uma análise a partir de diversos modelos gerados com diferentes configurações e algoritmos.

Para a realização da pesquisa, foram utilizados dados de interesse de *leads* por edições de produtos da Base Colaborativa®, que vende cursos e viagens de desenvolvimento pessoal. A partir de 5048 *leads*, foram compilados 6731 registros de interesse de algum *lead* por alguma edição de algum curso. A avaliação do método levou em conta a estrutura topológica, a representação vetorial e o algoritmo de ML utilizados em cada modelo gerado. Os resultados demonstraram que todos os três elementos citados anteriormente influenciam na qualidade do modelo. Apesar dos ótimos valores de ROC-AUC dos modelos, grande parte

dos mesmos se mostraram muito generalistas nos resultados. Dessa forma, um coeficiente de generalização γ foi desenvolvido para auxiliar na qualificação dos modelos, e os três melhores foram explicitados com os seguintes valores de γ respectivamente: 0,062, 0,018 e 0,011.

Apesar das contribuições do método proposto, este estudo está sujeito a limitações que requerem trabalhos futuros. Primeiro, com o intuito de lidar com a complexidade de redes heterogêneas e dinâmicas, é imprescindível aprofundar a capacidade do aprendizado de representação vetorial de rede, para que, menos informações topológicas e de semântica sejam perdidas. Segundo, a falta de padronização e organização das bases de dados do cenário de estudo interferiu diretamente nos resultados do cenário de estudo, limitando a exploração de estruturas topológicas.

Pesquisas futuras podem explorar outros métodos de representação vetorial que levam em consideração as características heterogêneas e dinâmicas das redes. Também, utilizar dados padronizados e melhor organizados para que não interfiram negativamente na modelagem do método e possam proporcionar mais opções de estruturas topológicas.

Ademais, pretende-se investigar o método na prática, ou seja, realizar uma ativação com abordagem de vendas nos *leads* classificados pelos modelos e analisar a conversão de *leads* para clientes.

REFERÊNCIAS

ABIODUN, Oludare Isaac et al. State-of-the-art in artificial neural network applications: A survey. **Heliyon**, v. 4, n. 11, p. e00938, 2018.

AHEARNE, Michael; HUGHES, Douglas E.; SCHILLEWAERT, Niels. Why sales reps should welcome information technology: Measuring the impact of CRM-based IT on sales effectiveness. **International Journal of Research in Marketing**, v. 24, n. 4, p. 336-349, 2007.

AMARA, Amina; TAIEB, Mohamed Ali Hadj; AOUICHA, Mohamed Ben. Network representation learning systematic review: Ancestors and current development state. **Machine Learning with Applications**, v. 6, p. 100130, 2021.

BARABÁSI, A.-L.; PÓSFAI, M. Network science. Cambridge: Cambridge University Press, 2016. ISBN 9781107076266. Disponível em: http://networksciencebook.com/

BRADFORD, William; JOHNSTON, Wesley James; BELLENGER, Danny. The impact of sales effort on lead conversion cycle time in a business-to-business opportunity pipeline. In: **6th International Engaged Management Scholarship Conference**. 2016.

BRATANIC, Tomaz. Complete guide to understanding Node2Vec algorithm. Towards Data Science, 16, agosto, 2021. Disponível em: https://towardsdatascience.com/complete-guide-to-understanding-node2vec-algorithm-4e9a35e5d147

BUCKLIN, Randolph; LEHMANN, Donald; LITTLE, John. From decision support to decision automation: A 2020 vision. **Marketing Letters**, v. 9, n. 3, p. 235-246, 1998.

CALDARELLI, Guido. **Large scale structure and dynamics of complex networks: from information technology to finance and natural science**. World Scientific, 2007.

CHAGAS, Beatriz Nery Rodrigues et al. Current applications of machine learning techniques in CRM: a literature review and practical implications. In: **2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI)**. IEEE, 2018. p. 452-458.

D'HAEN, Jeroen *et al*. Integrating expert knowledge and multilingual web crawling data in a lead qualification system. **Decision Support Systems**, v. 82, p. 69-78, 2016.

D'HAEN, Jeroen; VAN DEN POEL, Dirk; THORLEUCHTER, Dirk. Predicting customer profitability during acquisition: Finding the optimal combination of data source and data mining technique. **Expert systems with applications**, v. 40, n. 6, p. 2007-2012, 2013.

DAUD, Nur Nasuha *et al*. Applications of link prediction in social networks: A review. **Journal of Network and Computer Applications**, v. 166, p. 102716, 2020.

DHARWAL, Rajan; KAUR, Loveneet. Applications of artificial neural networks: a review. **Indian J. Sci. Technol**, v. 9, n. 47, p. 1-8, 2016.

EITLE, Verena; BUXMANN, Peter. Business analytics for sales pipeline management in the software industry: A machine learning perspective. In: **Proceedings of the 52nd Hawaii International Conference on System Sciences**. 2019.

ESPADINHA-CRUZ, Pedro; FERNANDES, A.; GRILO, António. Lead management optimization using data mining: A case in the telecommunications sector. **Computers & Industrial Engineering**, v. 154, p. 107122, 2021.

FURRER, David; THALER, Stephen. Neural-network modeling: neural-network modeling tools enable the engineer to study and analyze the complex interactions between material and process inputs with the goal of predicting final component properties. **Advanced materials & processes**, v. 163, n. 11, p. 42-47, 2005.

GEBERT, Henning *et al*. Towards customer knowledge management: Integrating customer relationship management and knowledge management concepts. In: **The Second International Conference on Electronic Business (ICEB 2002)**. 2002. p. 296-298.

GIACOSA, Elisa; CULASSO, Francesca; CROCCO, Edoardo. Customer agility in the modern automotive sector: how lead management shapes agile digital companies. **Technological Forecasting and Social Change**, v. 175, p. 121362, 2022.

GIL-GOMEZ, Hermenegildo et al. Customer relationship management: digital transformation and sustainable business model innovation. **Economic research-Ekonomska istraživanja**, v. 33, n. 1, p. 2733-2750, 2020.

GROVER, Aditya; LESKOVEC, Jure. node2vec: Scalable feature learning for networks. In: **Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining**. 2016. p. 855-864.

GUEROLA-NAVARRO, Vicente et al. Customer relationship management and its impact on innovation: **A literature review. Journal of Business Research**, v. 129, p. 83-87, 2021a.

GUEROLA-NAVARRO, Vicente et al. Research model for measuring the impact of customer relationship management (CRM) on performance indicators. **Economic research-ekonomska istraživanja**, v. 34, n. 1, p. 2669-2691, 2021b.

HAENLEIN, Michael; KAPLAN, Andreas. A brief history of artificial intelligence: On the past, present, and future of artificial intelligence. **California management review**, v. 61, n. 4, p. 5-14, 2019.

HAGENAUER, Julian; HELBICH, Marco. A geographically weighted artificial neural network. **International Journal of Geographical Information Science**, v. 36, n. 2, p. 215-235, 2022.

JÄRVINEN, Joel; TAIMINEN, Heini. Harnessing marketing automation for B2B content marketing. **Industrial Marketing Management**, v. 54, p. 164-175, 2016.

KASTHURIRATHNA, Dharshana Mahesh. The influence of topology and information diffusion on networked game dynamics. 2015.

KAVIANI, Sara; SOHN, Insoo. Application of complex systems topologies in artificial neural networks optimization: An overview. **Expert Systems with Applications**, v. 180, p. 115073, 2021.

KESTNBAUM, R. D.; HSIEH, L. A yardstick to measure inquiry quality. **Business Marketing. August**, p. 70-71, 1983.

KUMAR, Ajay et al. Link prediction techniques, applications, and performance: A survey. **Physica A: Statistical Mechanics and its Applications**, v. 553, p. 124289, 2020.

LAMRHARI, Soumaya et al. A social CRM analytic framework for improving customer retention, acquisition, and conversion. **Technological Forecasting and Social Change**, v. 174, p. 121275, 2022.

LEE, Jiho *et al*. An approach for discovering firm-specific technology opportunities: Application of link prediction to F-term networks. **Technological Forecasting and Social Change**, v. 168, p. 120746, 2021.

LI, Fangyuan; XU, Guanghua. AI-driven customer relationship management for sustainable enterprise performance. **Sustainable Energy Technologies and Assessments**, v. 52, p. 102103, 2022.

MALEK, Masoud *et al*. Shallow Node Representation Learning using Centrality Indices. In: **2021 IEEE International Conference on Big Data (Big Data)**. IEEE, 2021. p. 5209-5214.

MARTÍNEZ, Víctor; BERZAL, Fernando; CUBERO, Juan-Carlos. A survey of link prediction in complex networks. **ACM computing surveys (CSUR)**, v. 49, n. 4, p. 1-33, 2016.

MERO, Joel; TARKIAINEN, Anssi; TOBON, Juliana. Effectual and causal reasoning in the adoption of marketing automation. **Industrial Marketing Management**, v. 86, p. 212-222, 2020.

MICHIELS, Ian. **Lead lifecycle management: Building a pipeline that never leaks**. research report, Aberdeen Group (July), 2009.

MOLONTAY, Roland; NAGY, Marcell. Two decades of network science: as seen through the co-authorship network of network scientists. In: **Proceedings of the 2019 IEEE/ACM international conference on advances in social networks analysis and mining**. 2019. p. 578-583.

MONAT, Jamie P. Industrial sales lead conversion modeling. **Marketing Intelligence & Planning**, 2011.

MUNOZ, Justin *et al*. Deep learning based bi-level approach for proactive loan prospecting. **Expert Systems with Applications**, v. 185, p. 115607, 2021.

OGWUELEKA, Francisca Nonyelum et al. Neural network and classification approach in identifying customer behavior in the banking sector: A case study of an international bank.

Human factors and ergonomics in manufacturing & service industries, v. 25, n. 1, p. 28-42, 2015.

OHIOMAH, Alhassan Abdullahi; BENYOUCEF, Morad; ANDREEV, Pavel. Driving inside sales performance with lead management systems: A conceptual model. **Journal of Information Systems Applied Research**, v. 9, n. 1, p. 4, 2016.

OHIOMAH, Alhassan *et al*. The role of lead management systems in inside sales performance. **Journal of Business Research**, v. 102, p. 163-177, 2019.

PENG, Hao *et al*. Dynamic network embedding via incremental skip-gram with negative sampling. **Science China Information Sciences**, v. 63, n. 10, p. 1-19, 2020.

SABNIS, Gaurav *et al*. The sales lead black hole: On sales reps' follow-up of marketing leads. **Journal of marketing**, v. 77, n. 1, p. 52-67, 2013.

SAURA, Jose Ramon; RIBEIRO-SORIANO, Domingo; PALACIOS-MARQUÉS, Daniel. Setting B2B digital marketing in artificial intelligence-based CRMs: A review and directions for future research. **Industrial Marketing Management**, v. 98, p. 161-178, 2021.

WU, Yu-chen; FENG, Jun-wen. Development and application of artificial neural network. **Wireless Personal Communications**, v. 102, n. 2, p. 1645-1656, 2018.

XU, Ankun et al. Applying artificial neural networks (ANNs) to solve solid waste-related issues: A critical review. **Waste Management**, v. 124, p. 385-402, 2021.

YANG, Guangyu Robert; WANG, Xiao-Jing. Artificial neural networks for neuroscientists: A primer. **Neuron**, v. 107, n. 6, p. 1048-1070, 2020.

YU, You‐Ping; CAI, Shu‐Qin. A new approach to customer targeting under conditions of information shortage. **Marketing intelligence & planning**, 2007.

ZHANG, Caiming; LU, Yang. Study on artificial intelligence: The state of the art and future prospects. **Journal of Industrial Information Integration**, v. 23, p. 100224, 2021.

APÊNDICE A

Quadro 5 - ROC-AUC de todos os modelos de predição treinados

Topologia	Representação vetorial	Algoritmo de ML	ROC-AUC
Topologia 1	Representação vetorial 1	Random Forest	0,980
		Logistic Regression	0,883
	Representação vetorial 2	Random Forest	0,982
		Logistic Regression	0,884
	Representação vetorial 3	Random Forest	0,988
		Logistic Regression	0,957
Topologia 2	Representação vetorial 1	Random Forest	0,979
		Logistic Regression	0,873
	Representação vetorial 2	Random Forest	0,981
		Logistic Regression	0,891
	Representação vetorial 3	Random Forest	0,998
		Logistic Regression	0,996
Topologia 3	Representação vetorial 1	Random Forest	0,988
		Logistic Regression	0,936
	Representação vetorial 2	Random Forest	0,983
		Logistic Regression	0,912
	Representação vetorial 3	Random Forest	0,977
		Logistic Regression	0,961
Topologia 4	Representação vetorial 1	Random Forest	0,990
		Logistic Regression	0,956
	Representação vetorial 2	Random Forest	0,984
		Logistic Regression	0,918
	Representação vetorial 3	Random Forest	0,998
		Logistic Regression	0,993
Topologia 5	Representação vetorial 1	Random Forest	0,982
		Logistic Regression	0,904
	Representação vetorial 2	Random Forest	0,981
		Logistic Regression	0,905
	Representação vetorial 3	Random Forest	0,997
		Logistic Regression	0,995

Fonte: Elaborado pelo autor (2022).