UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

MARKOS FLÁVIO BOCK GAU DE OLIVEIRA

FILTRAGEM COLABORATIVA EM PESQUISAS DE CLIMA ORGANIZACIONAL: PREDIÇÃO DE ÍNDICE DE FAVORABILIDADE E DE OCORRÊNCIA DE COMENTÁRIOS

CURITIBA

2022

MARKOS FLÁVIO BOCK GAU DE OLIVEIRA

FILTRAGEM COLABORATIVA EM PESQUISAS DE CLIMA ORGANIZACIONAL: PREDIÇÃO DE ÍNDICE DE FAVORABILIDADE E DE OCORRÊNCIA DE COMENTÁRIOS

Collaborative filtering in surveys of organizational climate: Prediction of favorability index and occurrence of comments

Dissertação apresentada como requisito para obtenção do título de Mestre em Ciências do Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Dr. Ricardo Lüders

Coorientador: Profa. Dra. Myriam Regattieri de Biase da Silva Delgado

CURITIBA

2022

Ministério da Educação
Universidade Tecnológica Federal do Paraná
Campus Curitiba

MARKOS FLAVIO BOCK GAU DE OLIVEIRA

FILTRAGEM COLABORATIVA EM PESQUISAS DE CLIMA ORGANIZACIONAL: PREDIÇÃO DE ÍNDICE DE FAVORABILIDADE E DE OCORRÊNCIA DE COMENTÁRIOS

Trabalho de pesquisa de mestrado apresentado como requisito para obtenção do título de Mestre Em Ciências da Universidade Tecnológica Federal do Paraná (UTFPR). Área de concentração: Engenharia De Computação.

Data de aprovação: 05 de Outubro de 2022

Dr. Ricardo Luders, Doutorado - Universidade Tecnológica Federal do Paraná

Dr. Cesar Augusto Tacla, Doutorado - Universidade Tecnológica Federal do Paraná

Dr. Cristiano Roberto Dos Santos, Doutorado - Pin People S. A.

Documento gerado pelo Sistema Acadêmico da UTFPR a partir dos dados da Ata de Defesa em 05/10/2022.

À minha esposa Marta e ao meu filho Samuel.

AGRADECIMENTOS

Sou muito grato a Deus pelas pessoas que Ele colocou em minha vida. Sem o auxílio de minha esposa Marta, finalizar esse trabalho não seria possível. Obrigado meu amor pela demonstração quase diária de preocupação com a finalização dessa dissertação e na demonstração de fé na minha capacidade em concluí-la.

Sem o carinho, persistência e dedicação do professor Ricardo e da professora Myriam durante todo o meu mestrado eu também não teria finalizado esse trabalho. E se o meu colega de trabalho e amigo Cristiano não acreditasse no meu potencial, eu não poderia tê-lo iniciado.

Todos vocês são parte dessa conquista e de qualquer outra que eu venha a conquistar em minha vida, pois vocês não só ajudaram a construir este trabalho mas me ajudaram a crescer como pessoa. Sempre irei me espelhar em vocês em muitas coisas; apenas pra falar algumas, cito: a sabedoria do Ricardo, a brandura da Myriam, a humanidade do Cristiano e o amor da minha esposa Marta. Vou levar todos vocês para toda minha vida no meu coração e na minha memória.

Peço perdão pelos vários momentos que eu não agi e não correspondi como deveria.

Muito obrigado.

Porque todos nós temos recebido da sua
plenitude e graça sobre graça. – João 1:16

RESUMO

A Filtragem Colaborativa (FC) pode ser entendida como o processo de predizer preferências de usuários e derivar padrões úteis por meio do estudo de suas atividades. No contexto deste trabalho, FC é usada para predizer o nível de favorabilidade e a ocorrência de comentários em respostas a perguntas presentes em questionários extensos de clima organizacional de uma empresa. O objetivo deste trabalho é comparar o desempenho de quatro algoritmos baseados em FC (item-item, fatoração de matriz, fatoração de matriz logística e filtragem colaborativa neural) e uma abordagem de referência baseada em média simples. Os algoritmos são utilizados para estimar as respostas de baixa favorabilidade, ou seja, aquelas que um respondente não concorda com uma afirmação positiva sobre a empresa. Além disso, os algoritmos também são usados para estimar a emissão de comentários opcionais por respondentes. Para os dois problemas, foram utilizados dados de diferentes pontos de verificação (“checkpoints”) de pesquisas de clima organizacional, compostos ao todo por mais de 1,25 milhão de respostas de funcionários. Esses dados foram coletados entre 2019 e 2021 por uma grande empresa brasileira de tecnologia com mais de 10.000 funcionários. Os resultados mostram que as abordagens de filtragem colaborativa fornecem alternativas relevantes tanto para discriminar respostas de baixa favorabilidade na escala Likert quanto para discriminar a ocorrência de comentários, com estimativas de boa qualidade em ambos os casos. Esses resultados podem ser explorados para eventualmente reduzir o tamanho dos questionários, evitando fenômenos de sobrecarga enfrentados pelos respondentes em pesquisas extensas.

Palavras-chave: filtragem colaborativa; aplicação de questionário; pesquisa de clima organizacional; escala likert; predição de resposta.

ABSTRACT

Collaborative Filtering (CF) can be summarized as the process of predicting users preferences and deriving useful patterns by studying their activities. In this work, CF is used to predict the level of favorability and the occurrence of comments in answers to questions of large organizational climate surveys of a company. We aim to compare the performance of four algorithms based on CF (item-item, matrix factorization, logistic matrix factorization and neural collaborative filtering) and a baseline approach represented by a simple average of scores. The algorithms are used to estimate responses of low favorability, i.e., those that a respondent does not agree with a positive statement about the company. In addition, the algorithms are also used to estimate the registration of optional comments of respondents. For both problems, data from different checkpoints are used, comprising altogether more than 1.25 million employees' responses. The data was collected from 2019 to 2021 by a large Brazilian company of technology with more than 10,000 employees. The results show that collaborative filtering approaches provide relevant alternatives for discriminating low favorability answers in the Likert scale as well as the occurrence of comments, with good quality estimates in both cases. These results can be further explored to eventually reduce the size of the questionnaires, avoiding burden phenomena faced by respondents when dealing with large surveys.

Keywords: collaborative filtering; questionnaire application; organizational climate survey; likert scale; prediction of response.

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

Startups dedicadas à coleta e análise de dados de funcionários de empresas para melhor entendimento do ambiente de trabalho, as chamadas HR Techs (do inglês *Human Resource Techs*), estão ganhando visibilidade no mercado. Avaliar periodicamente as percepções e sentimentos dos funcionários sobre vários aspectos da organização está ajudando as empresas a compreender os comportamentos destes e os padrões (ocultos) do clima organizacional. Além disso, esse processo orientado a dados permite que os departamentos de recursos humanos tomem decisões de forma rápida e confiável. Esses processos tornam-se ainda mais críticos quando ocorrem eventos abruptos na dinâmica das organizações que acabam por alterar comportamentos e percepções dos funcionários. A pandemia da COVID-19 é um exemplo desse tipo de evento, que exigiu de várias organizações ao redor do mundo a migração para o trabalho remoto.

No entanto, uma das formas mais sistemáticas de obter informações sobre as pessoas dentro das empresas ainda é a aplicação de questionários. Uma metodologia de pesquisa típica é aplicar periodicamente (por exemplo, a cada dois trimestres) uma pesquisa de clima organizacional que tenta capturar a percepção dos funcionários em uma ampla gama de aspectos, desde saúde mental até trabalho em equipe. Essas pesquisas tendem a ser grandes (com até 300 perguntas) e repetitivas (em termos de conteúdo, sequência e quantidade de perguntas) para todos os entrevistados.

Levantamentos baseados em questões previamente fixadas apresentam diversos problemas, apesar da vantagem desse tipo de metodologia, principalmente no que diz respeito ao projeto, aplicação e controle de amostragem. Sabe-se, por exemplo, que existe uma relação inversamente proporcional entre o tamanho do questionário e a taxa de resposta. Isso ocorre porque os respondentes são mais propensos a não abrir questionários extensos. Se o fizerem, é mais provável que abandonem o questionário rapidamente, segundo um fenômeno conhecido como *survey breakoff* (EARLY; MANKOFF; FIENBERG, 2017; ZHANG *et al.*, 2020).

Além disso, os respondentes podem dar respostas para terminar o questionário mais rapidamente a fim de reduzir seu esforço cognitivo. Eles selecionam respostas arbitrárias (como a pontuação máxima para todas as respostas) ou escolhem respostas que permitem encurtar o questionário (usando “não” como resposta, por exemplo). Esse comportamento é chamado de *satisficing* (LAVRAKAS, 2008), que é mais frequente em questionários longos e ocorre principalmente nas últimas perguntas (EARLY; MANKOFF; FIENBERG, 2017; ZHANG *et al.*, 2020; GONZALEZ; ELTINGE, 2008).

Em suma, as empresas podem perder dados valiosos devido à desistência dos respondentes ou perder a qualidade dos dados devido ao *satisficing*, já que questionários grandes são mais suscetíveis à sobrecarga do respondente. Para superar esses problemas, alguns trabalhos sugerem estruturas de pesquisa adaptativas que reduzem ou reordenam as perguntas do questionário. Por exemplo, Zhang *et al.* (2020) propõem uma estratégia de redução baseada

na aprendizagem ativa, selecionando questões que maximizam a precisão de um modelo de fatoração matricial.

A abordagem proposta por Boim *et al.* (2012) assume o problema de redução como um problema de otimização, tentando minimizar a incerteza presente nos sistemas devido à distribuição diferente das respostas de cada questão.

Como tentativa de aprender parte do comportamento do usuário ao responder perguntas de favorabilidade ou emitir comentários sobre as respostas, o presente trabalho tem como foco a construção de modelos preditivos usando técnicas de filtragem colaborativa. Este é um primeiro passo para a construção de sistemas de recomendação de questionários mais enxutos e personalizados, uma vez que estimativas precisas seriam necessárias para orientar uma futura etapa de recomendação. Mostra-se que, de fato, é possível aprender essas estimativas com dados suficientes, que poderiam ser usados como substrato de um sistema de recomendação capaz de projetar questionários menores para objetivos bem definidos.

Para treinamento e teste dos modelos preditivos, foram selecionados mais de 1.25 milhão de avaliações de funcionários. Esses dados foram coletados a partir de seis levantamentos, realizados com um intervalo mínimo entre eles, e neste trabalho são denominados *checkpoints*. Nos experimentos, esses *checkpoints* são associados a pesquisas de clima organizacional realizadas entre 2019 e 2021 em uma empresa brasileira de tecnologia com mais de 10.000 funcionários. Para a etapa de predição de respostas de favorabilidade (alta ou baixa) do respondente e ocorrência ou não de comentários associados a estas respostas, foram aplicados cinco algoritmos, quatro baseados em filtragem colaborativa (FC) e uma abordagem *baseline*.

1.1 Objetivos

1.1.1 Objetivo geral

O objetivo geral desta dissertação é a aplicação de modelos de filtragem colaborativa para predição de eventos construídos a partir de respostas registradas em questionários de clima organizacional de uma grande empresa brasileira.

1.1.2 Objetivos específicos

Os objetivos específicos são:

- coletar e preparar os dados de seis diferentes *checkpoints* realizados por meio de questionários respondidos por funcionários da empresa;

- identificar algoritmos ou modelos de filtragem colaborativa para discriminar escores altos ou baixos das questões, assim como possíveis ocorrências nos questionários considerados de comentários associados a estas quetões;
- avaliar um mecanismo de configuração automática desses algoritmos utilizando uma pequena parcela dos dados;
- comparar o desempenho dos diferentes algoritmos ou modelos nos diferentes *checkpoints*;
- discutir a possibilidade do uso de sistemas de recomendação para a criação de questionários personalizados com base nos resultados obtidos.

1.2 Contribuições do trabalho

A principal contribuição desta dissertação é a ampliação da área de aplicação das técnicas de filtragem colaborativa para o contexto de pesquisas de funcionários, envolvendo tanto a predição da *favorabilidade* (alta ou baixa) do respondente a um aspecto abordado em uma questão; quanto a ocorrência ou ausência de comentários escritos opcionalmente pelo respondente. Uma questão cuja predição de favorabilidade é alta, pode então ser eliminada do questionário quando o objetivo é focar nos aspectos de necessidade de melhoria na empresa sendo avaliada.

Vale destacar que o uso de FC no contexto de pesquisas em geral, e de clima organizacional em particular, permite identificar padrões de comportamento sobre como os respondentes, funcionários neste caso, respondem a grandes pesquisas. Os resultados obtidos mostram ser possível explorar futuramente modelos construídos sobre esses padrões para criar questionários dinâmicos e personalizados para os respondentes. A seguinte publicação em conferência nacional é resultado desta dissertação (OLIVEIRA; DELGADO; LÜDERS, 2021):

OLIVEIRA, M. F. B. G.; DELGADO, M.; LÜDERS, R. Comparative analysis of collaborative filtering-based predictors of scores in surveys of a large company. In: SBC. Anais do XVIII Encontro Nacional de Inteligência Artificial e Computacional (ENIAC). [S.l.], 2021. p. 739–750.

1.3 Organização da dissertação

O Capítulo 2 apresenta a revisão da literatura com a formulação do problema e os trabalhos relacionados. Na sequência, o Capítulo 3 descreve os modelos de predição usados para a estimativa da favorabilidade das respostas e ocorrências de comentários. O Capítulo 4 detalha a metodologia utilizada nos experimentos realizados e o Capítulo 5 apresenta e discute os resultados obtidos. Por fim, o Capítulo 6 conclui o trabalho e apresenta perspectivas de trabalhos futuros.

2 REVISÃO DA LITERATURA

Este capítulo traz alguns fundamentos necessários para o entendimento do trabalho. Na Seção 2.1 são discutidos os conceitos básicos dos sistemas de recomendação, os quais podem ser úteis para a elaboração de questionários em pesquisas de clima organizacional. Essa relação embasa a escolha de filtragem colaborativa para a solução do problema abordado, o qual é descrito na Seção 2.2. A Seção 2.3 discute alguns trabalhos diretamente relacionados ao tema da pesquisa.

2.1 Sistemas de recomendação como suporte à pesquisa de clima organizacional

Em sistemas de recomendação existem duas etapas principais. A primeira é estimar eventos (ou interações) desconhecidos entre usuários e itens. Nessa etapa, preditores são construídos a partir de dados de eventos conhecidos. O segundo passo é ranquear os itens com base em estimativas e informações adicionais. Estimativas precisas realizadas na primeira etapa são cruciais para o desenvolvimento de sistemas de recomendação confiáveis. Este trabalho tem como foco a primeira etapa. Portanto, esta etapa será o principal tópico discutido nesta seção.

Os dados de eventos entre usuários e itens são comumente categorizados entre explícitos ou implícitos. Dados explícitos assumem que os eventos disponíveis para treino dos preditores foram explicitamente concedidos por usuários. Como exemplo de dados explícitos tem-se as classificações “Gostei” e “Não gostei” realizadas por usuários para avaliação de conteúdos, como por exemplo, filmes na plataforma Netflix[1]. Contudo, o número de avaliações explícitas é geralmente pequeno quando comparado ao número interações disponíveis (JOHNSON, 2014). Diante desse contexto, pode-se utilizar dados implícitos para a estimação da preferência dos usuários por itens. Por exemplo, para o problema abordado acima, pode-se considerar os filmes assistidos como aqueles aderentes ao perfil do usuário e utilizar essa informação como base para futuras recomendações.

A segunda etapa (etapa de ranqueamento) é mais qualitativa e dependente do problema e do contexto da aplicação. Uma abordagem comum é classificar os itens em ordem decrescente de estimativas. Voltando ao caso da Netlfix, na primeira etapa é construído um preditor que estima a probabilidade de um usuário assistir a um filme que ele ainda não assistiu considerando um histórico de eventos. A partir do modelo preditivo, é possível criar uma lista de recomendação com os cinco itens (filmes) com maior probabilidade de serem assistidos apontados pelo modelo. Entretanto, é possível considerar outros fatores como a diversidade dos itens

[1] Exemplos relativos à Netflix são comuns na literatura, uma vez que o lançamento de uma competição em 2006 (*The Netflix Prize*) lançada pela empresa para a melhora do sistema de recomendação existente, aumentou o interesse comercial e acadêmico desses sistemas (AMATRIAIN; BASILICO, 2016).

e a atualidade do conteúdo (FALK, 2019). Em suma, as estimativas advindas de preditores não são os únicos sinais úteis para construção das listas de recomendação, que podem ser construídas de maneira qualitativa. Mesmo assim, é importante que a métrica a ser otimizada na primeira etapa esteja direcionada aos objetivos da segunda. Schröder, Thiele e Lehner (2011) apresentam algumas diretrizes na escolha das métricas de otimização com base nas características do problema, enquanto que Vargas e Castells (2011) apresentam métricas focadas em otimizar a diversidade e novidade dos itens da lista de recomendação.

A literatura de sistemas de recomendação é vasta e vários algoritmos têm sido propostos para estimar um *evento* ou escore r_{ij} de interação entre o usuário i e o item j. Existem vários eventos possíveis que r_{ij} pode modelar, como eventos de clique, número de compras ou classificações explícitas dadas por usuários a itens no passado. Portanto, o tipo de sinal de r_{ij} depende do tipo de informação disponível para o sistema. Se o sistema tem acesso a avaliações explícitas de usuários em itens, então a interação capturada e generalizada pelo sistema é como as pessoas avaliam os itens nesta escala. Por exemplo, $r_{ij} \in \{1, 2, 3, 4, 5\}$ se as classificações forem dadas em uma escala de 1 a 5 estrelas. Se apenas informações implícitas estiverem disponíveis, como compras ou visualizações de itens, $r_{ij} \in \{0, 1\}$ e as estimativas geralmente representam a probabilidade do usuário interagir com um item. A interação implícita incorpora menos informações porque o valor zero (“0”) não indica uma baixa preferência do item ao usuário, uma vez que ele pode nem saber que o item existe.

Em geral, assume-se que o conjunto de itens M é grande quando comparado ao conjunto de usuários N. Em problemas envolvendo *e-commerce* M pode ser da ordem de milhões. Nessas situações em que $M \gg N$, se torna inviável treinar um modelo supervisionado para cada item, o qual, a partir das informações do usuário como entrada (idade, gênero, geração, etc.), poderia predizer um evento de interação entre o usuário e o item. Em contrapartida, é comum encontrar na literatura estratégias baseadas em conteúdo (do inglês *content-based*), que treinam modelos individuais para cada usuário e que predizem eventos usuário-item utilizando informações sobre os itens como *features* (PAZZANI; BILLSUS, 2007).

Este trabalho foca na construção e avaliação de preditores de filtragem colaborativa (o Capítulo 3 detalha os preditores utilizados), visando à criação de suporte para elaboração de questionários personalizados e relevantes para cada respondente. Para isso, usam-se eventos anteriores usuário-item, aqui redefinidos como *respondente-questão*, estruturados em uma matriz R de eventos históricos. A seção a seguir detalha o problema abordado com destaque para a definição das questões e respostas obtidas em levantamentos realizados por meio de questionários.

2.2 Descrição do problema

A descrição do problema está dividida em duas partes: definição do problema de estimação de eventos (seção 2.2.1) e a especificação deste problema para a recomendação de questões em pesquisas de clima organizacional (seção 2.2.2).

2.2.1 Estimação de eventos

Considere uma matriz parcialmente preenchida $R_{N \times M}$ de N usuários (respondentes) e M itens (perguntas). Cada entrada conhecida r_{ij} de R, com $i = 1 \dots N$ e $j = 1 \dots M$, é um índice (escore) que representa um evento passado do usuário i com o item j. Em geral, r_{ij} é obtido de um conjunto fixo, pequeno e ordinal de valores como $\{1,2,3,4,5\}$ ou $\{0,1\}$. Por exemplo, $r_{ij} = 5$ pode significar o *evento* do usuário i avaliar o item j com nota 5. Na maioria dos problemas, R é uma matriz esparsa (com dados ausentes) porque a maioria dos usuários não interage com a maioria dos itens.

O problema de estimação de eventos consiste na construção de um modelo de previsão que estime índices ou escores desconhecidos de R. Uma estimação (proxy) $\hat{R}$ é obtida a partir dos valores conhecidos de R buscando minimizar uma métrica de erro entre R e $\hat{R}$. Elementos de $\hat{R}$ são indicados por $\hat{r}_{ij}$ e podem, por exemplo, ser usados para guiar o mecanismo de recomendação na etapa de ranqueamento.

2.2.2 Estimação de eventos respondente-questão para pesquisas de clima organizacional

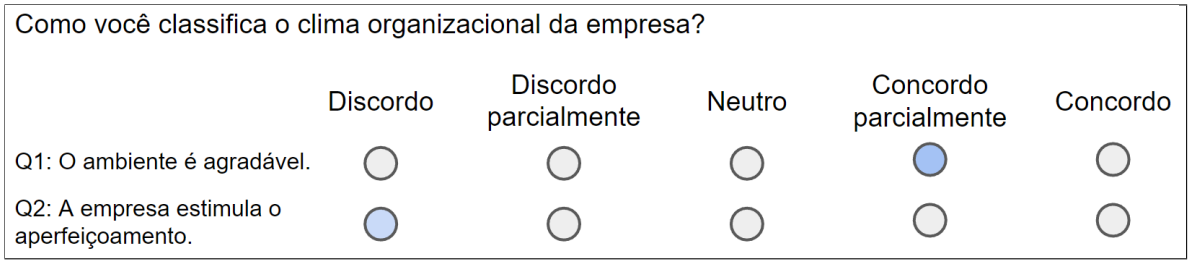
A forma mais comum de grandes organizações avaliarem as percepções dos funcionários é por meio da aplicação de questionários. O objetivo desses levantamentos é captar os sentimentos dos funcionários em relação a um amplo conjunto de aspectos da empresa.

Embora existam várias metodologias de pesquisa e um conjunto heterogêneo de tipos de perguntas (KROSNICK, 2018), os departamentos de RH geralmente se concentram em pesquisas com perguntas baseadas na escala Likert (também denominadas “perguntas de favorabilidade”). Nesse tipo de pergunta, o público recebe uma série de opções (geralmente 5 ou 7) para medir o quanto cada respondente é favorável a um determinado aspecto, ou seja, a sua *favorabilidade*. Nessa configuração, os respondentes são chamados de “favoráveis” ao aspecto avaliado quando respondem 4 ou 5 em uma escala de 1 a 5, ou 6 ou 7 em uma escala de 1 a 7.

A Figura 1 mostra um exemplo de duas questões na escala Likert de cinco pontos que poderiam estar em uma pesquisa de clima organizacional. Para o exemplo da figura, aqueles que respondem *Concordo parcialmente* ou *Concordo* resultam em índices de favorabilidade 4 ou 5 em uma escala de 1 a 5 e são considerados favoráveis à afirmação da questão. Aqueles que respondem *Neutro* (índice 3) são considerados neutros e aqueles que responderam *Discordo*

ou *Discordo parcialmente* (índices de favorabilidade 1 e 2) são considerados desfavoráveis à afirmação. Considerando as marcações de respostas na Figura 1, o respondente é favorável à afirmação da questão Q1 e desfavorável à afirmação da questão Q2.

Figura 1 – Um exemplo de questões na escala Likert, cada uma medindo a favorabilidade do respondente em relação a um determinado aspecto da empresa relativo a clima organizacional.



Fonte: Autoria própria.

Pesquisas de clima tradicionais são formadas por uma grande quantidade de questões de favorabilidade, cada uma avaliando um determinado aspecto da experiência do funcionário ou colaborador. Todavia, grandes questionários tendem a ter problemas, como baixo engajamento e alta taxa de desistência. Este trabalho avalia a possibilidade da construção de questionários personalizados baseados em estimadores que discriminam respostas com alto índices de favorabilidade das respostas com baixo índice de favorabilidade. Uma primeira possibilidade é criar estimadores dos índices 1 a 5 de favorabilidade e recomendar questões com baixa estimativa. Entretanto, este trabalho simplifica o problema e o trata de maneira binária.

Particularmente, o evento de interesse a ser estimado é o evento de baixa favorabilidade que incorpora os respondentes neutros e desfavoráveis. Isso porque, em geral, deseja-se avaliar aspectos das empresas que demandam melhoria e para isso é preciso construir estimadores que buscam predizer a probabilidade de um respondente i avaliar a afirmação de uma questão j com índice de favorabilidade 1, 2 ou 3. Essa configuração gera uma matriz respondente-questão R_s, em que $r_{ij} = 1$ representa o evento do respondente i ser desfavorável ou neutro à afirmação da questão j. Essa simplificação foi realizada pois alguns dos principais modelos de filtragem colaborativa funcionam apenas com dados de eventos binários. Além disso, essa configuração permite que o mesmo conjunto de modelos possa ser aplicado para este problema de estimação e para o problema a seguir de predição de existência de comentário, que é inerentemente binário.

É comum em questionários a possibilidade da adição de comentários para cada pergunta de favorabilidade em um espaço em que o colaborador pode justificar sua nota e detalhar sua perspectiva[2]. Assumindo que esse é o caso, é possível tentar estimar outro tipo de evento: a adição de um comentário por um usuário i em uma questão j. Para esse problema, tem-se

[2] Informação textual vinda de comentários é extremamente útil ao RH, uma vez que o seu formato dissertativo permite ao respondente reportar aspectos específicos da experiência e apontar as causas de eventuais problemas da empresa.

uma segunda matriz respondente-questão R_c, em que $r_{ij} = 1$ representa o evento do respondente i adicionar um comentário na pergunta j; $r_{ij} = 0$, caso contrário. Portanto, este trabalho considera a construção de preditores para dois problemas diferentes relativos às interações respondente-questão. A matriz R_s é utilizada para a estimação do evento de um respondente não ser favorável a um aspecto abordado em uma questão, enquanto que R_c é utilizada para a estimação do evento de escrita de comentários em uma questão por um determinado respondente.

Uma característica importante de R_s e R_c é que, diferentemente do caso usual em que $M \gg N$, tem-se o número de usuários (respondentes) muito maior do que o número de itens (questões). Isso porque, em geral, um universo de 50 questões, cada uma medindo um aspecto diferente da empresa, é suficiente para os gestores entenderem a visão dos seus colaboradores e agirem apropriadamente. A particularidade $M \ll N$, observada no contexto de pesquisa de clima organizacional, permite a exploração de técnicas de predição não muito frequentes na literatura de sistemas de recomendação, como técnicas que utilizam da demografia de pessoas na construção de modelos individuais para cada item (questão), como explorado por Al-Shamri (2016). No entanto, neste trabalho o foco será em estimar os eventos de interação respondente-questão utilizando modelos clássicos de recomendação, em especial os baseados em filtragem colaborativa.

2.3 Trabalhos relacionados

A maioria das pesquisas de clima organizacional aplicadas em grandes empresas costuma ser grande e fixa em termos de sequência e número de perguntas para todos os respondentes, pois os departamentos de RH das empresas desejam conhecer o maior número possível de aspectos sobre a organização. Apesar desse tipo de pesquisa ser simples de aplicar, sofre com o fenômeno da sobrecarga do usuário. Tal fenômeno impacta negativamente na experiência do respondente durante a pesquisa e na posterior análise quantitativa dos dados. Para lidar com essas questões, alguns trabalhos desenvolvem estratégias de elaboração de questionários adaptativos ou dinâmicos.

O design de pesquisa adaptável (ASD do inglês do inglês *adaptive survey design*) encontrado em Chun, Heeringa e Schouten (2018), Schouten, Calinescu e Luiten (2013), Wagner (2008) é um campo de estudo que usa informações baseadas em dados para criar pesquisas de alta qualidade. No entanto, o ADS tem um apelo mais genérico, pois considera vários aspectos da pesquisa para engajar mais participantes a comparecer e concluí-las. Por exemplo, o ADS pode estudar a diminuição esperada na taxa de não resposta (a proporção de não respondentes em relação àqueles que foram contatados) com uma mudança no modo de contato empregado (telefone, e-mail, etc.).

Uma outra linha de pesquisa tenta selecionar perguntas que reduzam a incerteza sobre as respostas. Early, Mankoff e Fienberg (2017) escolhem perguntas que maximizam o ganho de

informação medido pela entropia condicional em uma estratégia sensível ao custo. Nesse trabalho, as questões que exigem mais esforço cognitivo recebem mais penalidades e têm menos probabilidade de serem amostradas. A mesma ideia é seguida por Boim *et al.* (2012), mas estes abordam o problema usando uma formulação de otimização de restrições sem penalidades de custo.

Alguns trabalhos tratam do problema de gerar questionários reduzidos através de uma perspectiva de aprendizagem ativa. A aprendizagem ativa é uma abordagem de aprendizagem que tenta selecionar os dados mais informativos para um modelo de previsão. Nesse contexto, Zhang *et al.* (2020) propõem um mecanismo de questionário reduzido que seleciona perguntas ao usuário com base em sua capacidade de maximizar a precisão de um modelo de fatoração de matrizes. Esse modelo pode então reconstruir a tabela de questões do respondente, preenchendo as entradas vazias com estimativas precisas. Da mesma forma, Early, Mankoff e Fienberg (2017) visam maximizar o preenchimento da pesquisa selecionando perguntas que mais reduzem a incerteza de previsão de um modelo auxiliar que prevê uma variável de interesse. Nesse caso, se ocorrer um desistência de um respondente, o modelo maximizou seu aprendizado com as respostas enviadas. Finalmente, no trabalho publicado por Ortigosa, Paredes e Rodriguez (2010), uma árvore de decisão é construída com questões como nós. A árvore classifica os estilos de aprendizagem a partir das respostas de alunos, tarefa que originalmente demanda um grande número de questões. Ao usar este modelo para guiar a sequência de questões, pode-se fazer apenas as perguntas relevantes para o problema de predição de interesse.

Outros trabalhos desenvolvem abordagens que podem ser usadas para resolver (ou mitigar) o problema de inicialização a frio (LIKA; KOLOMVATSOS; HADJIEFTHYMIADES, 2014), também chamado de *cold-start*, que são situações em que o preditor precisa estimar eventos com itens ou usuários não presentes na etapa de treinamento. Dentro do contexto deste trabalho, quando um novo questionário é lançado, nenhuma resposta sobre usuários e perguntas é conhecida, ou seja, a matriz respondente-questão R está completamente vazia, e nenhuma predição (ou recomendação) é possível de ser realizada. Duas condições são necessárias para que o sistema comece a fazer recomendações assertivas para um usuário i: (i) uma quantidade de pessoas enviou suas respostas antes que i tenha começado a responder às perguntas, e; (ii) i já respondeu algumas perguntas.

São previstas três soluções para este problema. A primeira solução é aplicar uma abordagem baseada em conteúdo, como a proposta por Al-Shamri (2016). Nesse trabalho, o autor usa os dados demográficos dos entrevistados para gerar recomendações, o que dispensa a segunda condição acima, sendo possível gerar recomendações para i desde a primeira questão do questionário. Uma segunda solução é usar técnicas de aprendizagem ativa e de exploração para orientar as recomendações para as primeiras respostas. Por exemplo, a proposta de Zhang *et al.* (2020) poderia ser usada para orientar o mecanismo de recomendação com o objetivo de maximizar o desempenho de um modelo de previsão de fatorização matricial. Outros algoritmos que buscam compensações de exploração-explotação ótimas (com e sem contextos) podem

ser usados nesse sentido (LI; KARATZOGLOU; GENTILE, 2016; WANG; WU; WANG, 2017; WU *et al.*, 2016; SONG; TEKIN; SCHAAR, 2014). A terceira solução possível é usar o histórico de respostas de questionários antigos. Koren (2009) apresenta uma abordagem temporal que incorpora eventos passados para o treino de preditores de filtragem colaborativa, incluindo termos de vieses na formulação do preditor para cada janela temporal significativa.

O presente trabalho não trata do problema de *cold-start* e avalia os preditores em um experimento construído de forma que os dados de treinamento e de teste contenham todos os respondentes e questões. Além disso, o trabalho se concentra no treinamento de modelos de filtragem colaborativa, que tentam prever eventos (de favorabilidade e de emissão de comentários) a partir das respostas dos funcionários em questões de escala Likert, cada uma avaliando um aspecto particular da empresa. Talvez o trabalho mais semelhante ao nosso seja encontrado em Zhang *et al.* (2020). Ao contrário do nosso trabalho que avalia a possibilidade de criação de modelos de predição de eventos relativos à interações respondente-questão para a eventual criação de sistemas de recomendação de questões relevantes (baixa favorabilidade e com comentários), o modelo proposto pelo autor é útil em um contexto de aprendizagem ativa que busca melhorar o modelo a cada nova observação. Além disso, em Zhang *et al.* (2020), os autores buscam entender as relações ocultas nos fatores resultantes do aprendizado, processo que permite a identificação das questões mais informativas, por exemplo.

3 ALGORITMOS DE FILTRAGEM COLABORATIVA

Este capítulo apresenta os conceitos básicos de filtragem colaborativa na Seção 3.1 e também detalha os quatro algoritmos de filtragem colaborativa utilizados para predição: i) item-item na Seção 3.2; ii) fatoração de matrizes na Seção 3.3; iii) fatoração de matrizes logística na Seção 3.4; e iv) filtragem colaborativa neural na Seção 3.5, além da abordagem *baseline* na Seção 3.6.

Para tanto, a seguinte notação é utilizada:

$R_{N \times M}$: matriz de interações com M usuários (respondentes) e N itens (questões).

r_{ij}: escore de interação entre o usuário i e o item j.

Ω_j: conjunto de todos os usuários que interagiram com o item j.

Ψ_i: conjunto de todos os itens que o usuário i interagiu.

$\Psi_{ii'}$: conjunto de todos os itens que ambos os usuários i e i' interagiram.

$\hat{r}_{i,j}$: predição da interação entre o usuário i e o item j.

Ω: conjunto de pares (i,j) em que o usuário i interagiu com o item j.

3.1 Filtragem colaborativa

A literatura sobre sistemas de recomendação concentra-se em estimar interações desconhecidas entre usuários e itens. Em seguida, conjuntos de dados de referência explícitos ou implícitos, como MovieLens[1], são usados para avaliar o desempenho (geralmente MSE (*mean squared error*) ou RMSE (*root mean squared error*)) de cada abordagem de predição utilizada. Estas abordagens recebem como entrada uma matriz R embutida que, mesmo sendo esparsa, contém milhões de eventos de interação item-usuário conhecidos. A partir de R, os preditores constroem modelos capazes de gerar previsões ($\hat{R}$) que podem ser usadas, por exemplo, para recomendações.

A maioria dos sistemas de recomendação na literatura foca no desenvolvimento de algoritmos que predizem $\hat{R}$ a partir de R. Esses algoritmos geralmente são categorizados em duas grandes classes: filtragem colaborativa (FC) e algoritmos baseados em conteúdo (BC).

Em FC, cada interação passada (valores conhecidos de R) colabora com o sistema na compreensão de como o usuário i interage com o item j. Mais explicitamente, todas as classificações $r \in R$ são usadas para avaliar $\hat{r}_{i,j}$; não apenas aqueles relacionados com i ou j[2].

Diferentemente de FC, nos algoritmos BC cada item (ou usuário) é mapeado para um perfil de valores de recursos (ou *features*) conhecidos (PAZZANI; BILLSUS, 2007). Por exemplo, uma questão de um questionário pode ser mapeada (por um especialista) para um vetor de

[1] https://grouplens.org/datasets/movielens/

[2] Em algumas abordagens o escopo de interações que impacta a estimativa $\hat{r}_{i,j}$ é limitada a uma vizinhança de i e j para redução do esforço computacional.

valores, cada valor medindo o grau de interseção entre a pergunta e conceitos semânticos relativos à empresa, como liderança, segurança no trabalho, etc. Um modelo M_i é aprendido para cada usuário i a partir de um conjunto de exemplos (respostas antigas) em uma estrutura de aprendizado supervisionado. As entradas são os vetores contendo as *features* dos itens classificados por i, enquanto as saídas são as classificações. Apenas o comportamento passado do usuário i nos itens de Ψ_i é usado para previsões; ou seja, não há colaboração.

Segundo Su e Khoshgoftaar (2009), os algoritmos FC podem ser categorizados em dois tipos: baseados em memória ou baseados em modelo. Os algoritmos de FC baseados em memória dependem explicitamente das correlações das classificações entre os usuários ou itens mais semelhantes para gerar previsões para pares de usuário-item. Em contraste, algoritmos baseados em modelo tentam aproximar R com $\hat{R}$ para que um erro entre R e $\hat{R}$ seja minimizado. $\hat{R}$ é fatorado em uma matriz de usuários e em uma matriz de iten s que tem dimensionalidade reduzida k. Estas matrizes são multiplicadas para construir $\hat{R}$. Cada usuário i e item j são descritos com k fatores latentes que são aprendidos iterativamente a partir das classificações conhecidas de R. Como sistemas baseados em FC apresentam ótimo desempenho em conjuntos de dados de *benchmark* (SU; KHOSHGOFTAAR, 2009), a maior parte do estado da arte atual de sistema de recomendação é construída com base em FC ou sistemas híbridos que combinam FC com outras técnicas. Assim, FC é a metodologia abordada nesta dissertação. Conforme detalhado a seguir, serão utilizados tanto algoritmos baseados em memória quanto algoritmos baseados em modelo.

Para as etapas de predição dos índices de favorabilidade e ocorrência de comentários, foram selecionados: um algoritmo baseado em memória (item-item), dois algoritmos de fatoração de matriz (baseado em modelo) e uma abordagem baseada em redes neurais. Tanto os modelos de fatoração quanto o neural criam vetores latentes w e u para cada usuário e item, respectivamente. Cada modelo está representando uma abordagem relevante da literatura de sistemas de recomendação usando filtragem colaborativa (RENDLE *et al.*, 2020a; KULKARNI; RAI; KALE, 2020). As seções a seguir explicam as particularidades de cada abordagem escolhida: item-item (II); fatoração de matrizes (MF); fatoração logística de matrizes (LMF); filtragem colaborativa neural (NCF) e um algoritmo de média simples usado como referência (*baseline*).

3.2 Item-Item

O algoritmo item-item (II) é um modelo de FC baseado em memória que utiliza a similaridade das classificações de itens para orientar as recomendações. Um item é recomendado para um usuário se o usuário classificou bem itens semelhantes no passado. Dois itens são semelhantes se os usuários deram classificações semelhantes a eles. A similaridade é frequentemente medida por similaridade de cosseno ou correlação de Pearson. Al-Shamri (2016) e Su e Khoshgoftaar (2009) descrevem várias outras medidas de similaridade que podem ser usadas para comparar itens ou perfis de usuários. Apesar de sua simplicidade conceitual, é uma abor-

dagem muito competitiva, pois fornece resultados de desempenho semelhantes aos preditores de fatoração de matrizes.

Formalmente, para o modelo II a estimativa $\hat{r}_{ij}$ é dada por (1).

$$\hat{r}_{ij} = \bar{r}_j + \frac{\sum_{j' \in \Psi_i} w_{jj'}(r_{ij'} - \bar{r}_{j'})}{\sum_{j' \in \Psi_i} |w_{jj'}|} \tag{1}$$

sendo $w_{jj'}$ a medida de similaridade entre os itens j e j', Ψ_i o conjunto de itens avaliados pelo usuário i, e $\bar{r}_j$ a classificação média do item j (o mesmo para $\bar{r}_{j'}$). A pontuação dada por este método é a classificação média do item mais uma média ponderada das diferenças das classificações com relação a classificação média. O uso das diferenças ao invés da classificação pura ocorre pois cada usuário tem seu próprio viés de avaliação dos itens: um usuário conservador pode considerar apenas a nota cinco (5) como uma boa pontuação, enquanto um usuário mais flexível pode considerar a nota (3) como boa o suficiente. Assim, mede-se quanto o usuário i classifica j' comparado com o quanto a população de usuários classifica j'. A medida de similaridade pode ser dada pelo coeficiente de correlação de Pearson entre as distribuições de classificação de j e j' fornecidas por todos os usuários que avaliaram tanto j quanto j'.

O modelo usuário-usuário funciona de maneira semelhante ao item-item, mas recomenda itens para usuários com base nas semelhanças do usuário em vez das semelhanças do item. No entanto, o modelo II é frequentemente preferido porque as medidas de similaridade entre os itens ($w_{jj'}$) são mais precisas. Esse é o caso porque, na maioria dos problemas, dois itens têm muito mais usuários em comum do que dois usuários têm itens em comum. Item-item também é computacionalmente mais rápido para problemas em que $N >> M$, que é o caso geral.

3.3 Fatoração de matrizes

No algoritmo de fatoração de matrizes (MF do inglês *matrix factorization*), o objetivo é construir $\hat{R}$ como uma aproximação de baixo nível (*low rank*) de R. $\hat{R}$ é então fatorado em duas matrizes, como mostrado em (2).

$$\hat{R}_{N \times M} = W_{N \times k} U^T_{M \times k} \tag{2}$$

Na matriz do usuário W, cada linha representa um vetor de k características latentes de um usuário. Da mesma forma, a matriz de itens U possui linhas que contêm representações de itens. Usuários e itens são então projetados em um espaço compartilhado latente com dimensionalidade k. Esta abordagem de fatoração é semelhante a um SVD (do inglês *Singular Value Decomposition*) truncado que aproxima uma matriz A a $U_{N \times k} S_{k \times k} V^T_{M \times k}$ se A tem posto completo e $N > M$. Substituindo $W = US$, obtemos a mesma aproximação de (2).

Uma maneira possível de construir $\hat{R}$ é atualizar iterativamente as representações vetoriais de W e U para a minimização do erro quadrático médio entre R e $\hat{R}$. Para avaliar o erro individual entre uma previsão $\hat{r}_{ij}$ e a classificação verdadeira r_{ij}, a função de perda regularizada de (3) pode ser usada,

$$J = \sum_{i,j \in \Omega} (r_{ij} - \hat{r}_{ij})^2 + \lambda(||W||_F^2 + ||U||_F^2 + b_2^2 + c_2^2) \tag{3}$$

com penalidade de regularização L2 dada por λ, $|| \cdot ||_F^2$ identificando a norma Frobenius e $\hat{r}_{ij}$ calculada de acordo com (4).

$$\hat{r}_{ij} = w_i^T u_j + b_i + c_j + \mu \tag{4}$$

sendo w_i o vetor de usuário, u_j o vetor de item, b_i o termo de viés para o usuário i, c_j o termo de viés para o item j e μ a classificação média de R. A equação (4) é a maneira mais comum de projetar a classificação de previsão para modelos de fatoração de matrizes. Os vieses são incluídos para modelar especificidades existentes nos itens e usuários, como otimismo do usuário ou popularidade do item, e geralmente aumentam o desempenho (HE *et al.*, 2017). Em Koren (2009), o uso desses vieses foi desenvolvido para capturar efeitos dinâmicos temporais, como mudanças na percepção dos usuários sobre os itens. Os mínimos quadrados alternados (ALS do inglês *alternated least squares*) podem ser usados para encontrar os parâmetros do modelo que minimizam a função de perda J das soluções de forma fechada construídas a partir da derivada de J. Este é o modelo de otimização utilizado neste trabalho.

3.4 Fatoração de matrizes logística

A fatoração de matrizes logística (LMF do inglês *logistic matrix factorization*) é um algoritmo FC projetado para problemas de dados implícitos (JOHNSON, 2014). Semelhantemente ao NCF (Seção 3.5), ele modela a probabilidade de um usuário preferir um item usando uma função logística, mas restringindo $\hat{r}_{ij}$ a uma função linear, como mostrado em (5).

$$\hat{r}_{ij} = w_i^T u_j + b_i + c_j \tag{5}$$

A equação (6) mostra a função de probabilidade (verossimilhança logarítmica) dos parâmetros w_i, u_j, b_i e c_j.

$$L = \log[p(W, U, b, c \mid R)] = \sum_{i,j \in \Omega} \alpha r_{ij} \hat{r}_{ij} - (1 + \alpha r_{ij}) \log(1 + \exp(\hat{r}_{ij})) - \frac{\lambda}{2} {w_i}_2^2 - \frac{\lambda}{2} {u_j}_2^2 \tag{6}$$

com α sendo um hiperparâmetro que pondera observações e λ sendo a penalidade usual de regularização L2 (aplicada apenas aos vetores latentes). O método gradiente descendente é

usado para maximizar L. Para redução do tempo de treinamento, Johnson (2014) sugere usar um procedimento de amostragem negativa e um cronograma de aprendizado adaptativo com AdaGrad.

3.5 Filtragem colaborativa neural

A filtragem colaborativa neural (NCF do inglês *neural colaborative filtering*) de He *et al.* (2017) é uma abordagem atraente para conjuntos de dados implícitos que ganhou atenção na comunidade de sistemas de recomendação devido à sua alta escalabilidade e arquitetura de rede flexível.

No modelo NCF, o produto escalar entre w_i e u_j é substituído por uma função mais complexa que é aprendida por retropropagação. Em sua forma mais simples (utilizada neste trabalho), a rede recebe os vetores de interação $row_i(R)$ do usuário i, e $col_j(R)$ do item j. Em seguida, ele gera vetores latentes de ambas as entradas e os concatena em um único vetor. Este vetor é passado para uma sequência de camadas totalmente conectadas. A função de ativação da unidade de saída é uma função logística adequada para dados implícitos fornecendo $\hat{r}_{ij} \in [0,1]$.

A rede é treinada para minimizar a função de perda de entropia cruzada binária de (7). J é então minimizado com gradiente descendente estocástico (SGD).

$$J = -\sum_{i,j \in \Omega} r_{ij} \log \hat{r}_{ij} + (1 - r_{ij}) \log (1 - \hat{r}_{ij}) \tag{7}$$

Uma vantagem desse modelo é que ele pode ser adaptado facilmente para modelagens baseadas em conteúdo. Por exemplo, é possível substituir $row_i(R)$ por um vetor de recursos relativos ao usuário i, como geração, gênero, etc. Dessa forma, o formato híbrido desse modelo auxilia a estimação em casos de inicialização a frio.

3.6 Média simples (*Baseline*)

Algumas técnicas que recomendam itens baseados na sua popularidade são bastante comuns e podem ser utilizados como *baselines* a serem batidos por técnicas mais robustas. Um desses algoritmos avalia $\hat{r}_{ij}$ como sendo simplesmente a média dos escores de cada interação que envolveu j, segundo a Equação 8.

$$\hat{r}_{ij} = \frac{\sum_{i' \in \Omega_j} r_{i'j}}{|\Omega_j|} \quad i = 1, \ldots, N \tag{8}$$

Note que a predição $\hat{r}_{ij}$ independe do usuário i; ou seja, não envolve personalização, pois fornece um único escore de aderência para o item j. Quanto maior a performance desse

algoritmo, mais os usuários tendem a convergir na sua decisão de interação com os itens. Ou seja, mais a popularidade dos itens é útil na recomendação. Em contrapartida, o objetivo dos algoritmos de recomendação é explorar as diferenças dos usuários para a proposição de recomendações personalizadas.

4 METODOLOGIA EXPERIMENTAL

Este capítulo descreve na Seção 4.1 os dados utilizados dos seis levantamentos (*checkpoints*) utilizados nos experimentos. A Seção 4.2 apresenta o procedimento de configuração dos hiperparâmetros[1] dos modelos MF, LMF e NCF (o modelo II não necessita de configuração). Em seguida, a Seção 4.3 apresenta o cenário de experimentação em que os modelos de predição são treinados e testados, bem como as métricas de avaliação utilizadas.

4.1 Descrição dos dados

Para a experimentação dos modelos discutidos no capítulo anterior, foram selecionados seis levantamentos (cp do inglês *checkpoint*) de pesquisas de clima organizacional, aplicados a uma empresa de tecnologia brasileira de grande porte com mais de 10.000 colaboradores. Os levantamentos foram aplicados entre junho de 2019 e novembro de 2021 a cada dois trimestres. Sendo pesquisas de clima, os questionários são relativamente extensos e com escopo de temas abrangente. Não houve obrigatoriedade na participação das pesquisas, que permaneceram abertas durante um período de 15 dias. Além disso, têm-se apenas os dados de respostas daqueles que finalizaram a submissão do questionário completo.

Para a construção das matrizes respondente-questão utilizadas para os experimentos, foi necessário excluir as respostas de perguntas dissertativas e de alternativas. Mantiveram-se apenas as questões na escala Likert de 5 pontos (questões de favorabilidade), que representam 77,5% do conjunto total de questões avaliadas nos seis *checkpoints*.

Para cada um dos seis *checkpoints*, obtiveram-se as matrizes R_s e R_c completas (r_{ij} com valores conhecidos). Algumas informações sobre cada *checkpoint* estão presentes na Tabela 1. Para um mesmo cp, as matrizes R_c e R_s têm a mesma dimensão $M \times N$, em que M é o número de respondentes e N é o número de questões de favorabilidade. Na Tabela 1, a variável C representa o total de comentários. A taxa de adesão é a representatividade dos respondentes do total de colaboradores que puderam responder à pesquisa. Nota-se que a taxa de adesão fica em torno de 80% para todos os *checkpoints*.

Tabela 1 – Dados de cada *checkpoint* (*cp*) após o processo de filtragem.

cp	Mês/Ano	M	N	C	Taxa de adesão
1	06/2019	3512	25	4778	79,4%
2	11/2019	5491	31	6891	82,0%
3	06/2020	5301	41	7929	81,2%
4	11/2020	5344	51	9627	83,0%
5	06/2021	5313	43	7649	83,4%
6	11/2021	5518	50	9339	81,4%

Fonte: Autoria própria.

[1] Por simplicidade, os hiperparâmetros dos modelos serão denominados parâmetros nesta dissertação.

Em geral, observa-se que o números de questões N cresce a cada levantamento, o que significa que a empresa aumentou o escopo das perguntas ao longo do tempo. Além disso, o número de respondentes M aumentou significativamente entre o $cp = 1$ e o $cp = 2$. Como a taxa de adesão de ambos os levantamentos é similar, pode-se inferir que a empresa ampliou a abrangência da pesquisa após o $cp = 1$.

4.2 Configuração dos modelos de predição

Esta seção descreve o processo de configuração dos modelos utilizados para a estimação dos eventos das matrizes R_s e R_c. Com exceção do modelo II, todos os demais modelos apresentam parêmetros que devem ser configurados. As configurações candidatas são inicialmente definidas a partir de uma busca aleatória (*random search*) no espaço de parâmetros (**??**), seguida de um processo de escolha da melhor configuração, conforme detalhado a seguir.

Todos os modelos parametrizados considerados constroem vetores latentes de dimensionalidade k para cada respondente e questão (vetores w e u). Em geral, quanto maior é o valor de k, maior é a capacidade de ajuste aos dados de treinamento. Portanto, maior é a possibilidade de *overfitting* (BRISCOE; FELDMAN, 2011). Como o valor ideal de k depende do problema e dos dados de treinamento, esse parâmetro foi o primeiro a ter seu valor definido pelo método proposto neste trabalho.

Outro parâmetro selecionado que é comum a todos os modelos parametrizados é o número de iterações (N_{it}) ou épocas no modelo neural. Esse parâmetro também é ajustado para se evitar o fenômeno de *overfitting*, limitando a ação do modelo nos dados de treinamento a apenas algumas rodadas. Foram também incluídos o parâmetro λ de regularização L2 para os modelos LMF e NCF e o parâmetro α, que controla a taxa de atualização (*learning rate*) dos parâmetros de treinamento para os modelos LMF e NCF. Por fim, testam-se diferentes arquiteturas da rede neural do modelo NCF, variando-se a quantidade N_h de camadas escondidas (*hidden layers*) e o tamanho n_s da última camada escondida, o qual define o tamanho das demais.

Os valores para cada parâmetro dos modelos que necessitam de configuração aparecem na Tabela 2. Os valores candidatos foram obtidos considerando uma variação exponencial de base 2 entre os valores. Foram consideradas redes neurais com $L = N_h + 2$ camadas, sendo N_h o total de camadas escondidas, com a camada de entrada $l = 1$ e a camada de saída $l = L$. Considerando n_s o número de neurônios na última camada escondida (camada $l = L - 1$), foram criadas redes em que o número de neurônios de uma camada escondida ($1 < l < L$) é igual a $(2^{(N_h - l + 1)} \cdot n_s)$. Ou seja, o número de neurônios nas camadas escondidas cai pela metade a cada nova camada. Esses valores foram obtidos com base em experimentos preliminares realizados com o conjunto de treinamento disponível do primeiro *checkpoint* ($cp = 1$). Entretanto, alguns parâmetros foram definidos manualmente, como a taxa de amostragem negativa para o modelo NCF, que é definida como 1 de acordo com He *et al.* (2017).

Tabela 2 – Lista de valores para os parâmetros de cada modelo que necessita de configuração.

Parâmetro	MF	LMF	NCF
k	{16, 32, 64}	{16, 32, 64, 128}	{16, 32, 64, 128}
N_{it}	{2, 4, 8}	{32, 64, 128, 256, 512}	{1, 2, 4}
λ	{2, 4, 8, 16}	{8, 16, 32, 64}	-
α	-	{0.5, 1, 2}	{1, 2, 4} ($\times 10^{-4}$)
N_h	-	-	{2, 4}
n_s	-	-	{4, 8, 16}

Fonte: Autoria própria.

Para a configuração de parâmetros realizada neste trabalho, selecionam-se exclusivamente os dados do primeiro *checkpoint* ($cp = 1$). Esse *checkpoint* não é utilizado na etapa de comparação de desempenho, em que apenas os *checkpoints* $cp = 2$ a $cp = 6$ são avaliados. A configuração é realizada de maneira independente para cada matriz de eventos R_s e R_c.

A Figura 2 ilustra o processo de configuração dos modelos com os dados do $cp = 1$. Para a escolha da melhor configuração são definidas 50 configurações candidatas para cada modelo por meio de busca aleatória (**??**). Ou seja, as configurações são formadas a partir de seleções aleatórias dos valores mostrados na Tabela 2.

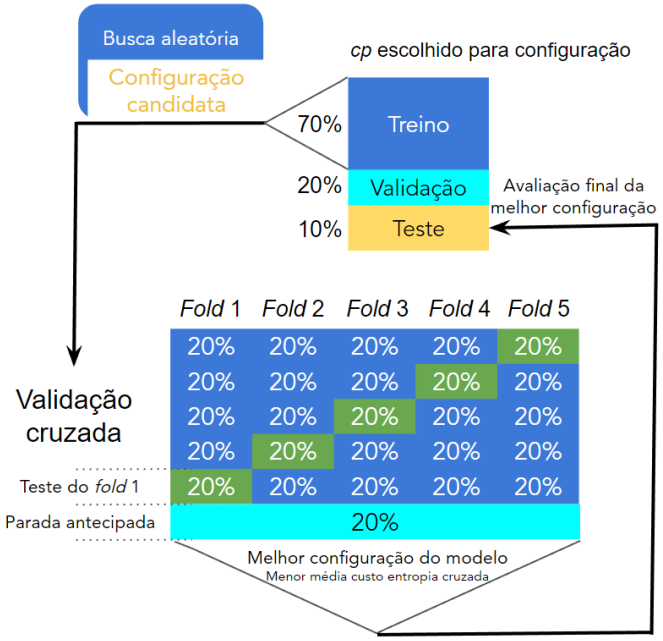
Cada uma das 50 configuração candidatas é treinada e avaliada usando 90% dos dados de eventos do $cp = 1$ (ver topo da Figura 2), sendo 70% dos dados utilizados para o treinamento e 20% para a validação (ou parada antecipada conforme detalhado na sequência). Cada respondente contribui igualmente para cada conjunto de treino e validação. Por exemplo, para um respondente i que interage com 25 questões no $cp = 1$, uma parte formada por 22 questões (90% de 25 arredondado para baixo) é usada para a configuração. Dessas 22 questões, 20 são usadas para treino e duas são usadas para parar antecipadamente o treino quando necessário.

Cada configuração candidata passa por uma validação cruzada (CV) - do inglês *cross-validation* - com 5 *folds* nos 70% dos dados para treinamento. Como em um processo tradicional de CV – conforme mostra região central da Figura 2, a configuração é treinada em 80% dos dados de treino (células em azul escuro) e testada nos 20% restantes do treino (células verdes). Para evitar *overfitting*, controla-se o número de iterações do treinamento em cada *fold* observando-se o custo de entropia cruzada (*cross entropy loss*) no conjunto de validação e realizando-se a parada antecipada (*early-stopping*). A configuração com a menor perda de entropia cruzada média nos *5 folds* é selecionada.

Em seguida, o modelo com a configuração vencedora é re-treinado com todos os 70% dos dados de treinamento (utilizando os mesmos 20% dos dados da validação para parada antecipada) e é submetido a uma avaliação pós-configuração utilizando os 10% restantes dos dados de eventos do $cp = 1$. Isso é feito apenas para uma análise da sua capacidade de generalização.

O processo descrito anteriormente é realizado uma única vez. Após essa etapa, cada modelo parametrizado com sua melhor configuração é selecionado para avaliação nos *checkpoints* restantes, ou seja, naqueles não utilizados na configuração.

Figura 2 – Processo de configuração dos modelos via validação cruzada sobre os dados de treino.



Fonte: Autoria própria.

4.3 Experimento de comparação de desempenho

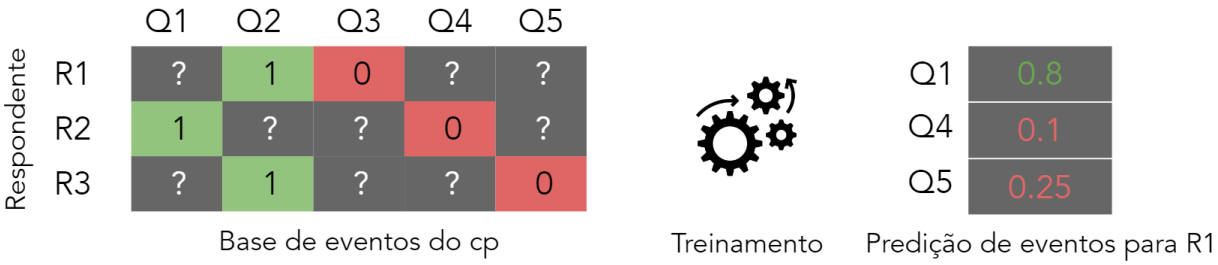
O treinamento e teste dos modelos parametrizados (agora com todos os parâmetros fixados) foram realizados de forma independente para cada *checkpoint* $cp = 2, \ldots, 6$. Em outras palavras, todos os modelos com suas melhores configurações de parâmetros são avaliados nesta etapa final utilizando agora os dados não utilizados na etapa de configuração.

A comparação de desempenho dos modelos acontece em um experimento em que os dados de treinamento e de teste estão em um mesmo *checkpoint*. Simula-se um cenário em que os respondentes têm acesso ao mesmo questionário, cuja ordem das questões é aleatória para cada respondente. Por simplicidade, considera-se que os dados de treinamento (base histórica de eventos) têm a mesma proporção das respostas de cada respondente.

A dinâmica descrita acima é mostrada na Figura 3, na qual 40% dos eventos de cada respondente são conhecidos (dois de cinco). No contexto dos dados de eventos de R_c, um preditor pode ser treinado para prever eventos de emissão de comentários em questões que ainda não foram apresentadas aos respondentes (como as questões Q2, Q3 e Q5 para o respondente R2). Para o respondente R1 da Figura 3, o sistema aponta Q1 como a questão mais provável de receber um novo comentário. Diante disso, um sistema de recomendação pode priorizar essa questão.

Considera-se que, quanto maior o nível de informação disponível aos preditores (proporção de eventos conhecidos em R) maior é a facilidade dos preditores em estimar os eventos não conhecidos. Para avaliar o impacto dessa variável na capacidade de predição dos mode-

Figura 3 – Exemplo de aplicação de um preditor no cenário 1. A tabela da esquerda representa uma matriz respondente-questão com 5 questões e 3 respondentes com 40% dos eventos conhecidos. Para o respondente R1, o sistema tem mais confiança de que Q1 é a questão mais provável de receber um evento da classe positiva.



Fonte: Autoria própria.

los, cada preditor foi treinado com diferentes níveis de informação $NI = 20\%, 40\%, 60\%, 80\%$. Assim, uma porcentagem NI dos eventos de cada respondente foi alocada no conjunto de treinamento segundo uma distribuição uniforme. O restante dos eventos foi utilizado para teste. Isso foi feito para cada $cp = 2, 3, \ldots, 6$ e para cada matriz R_s e R_c de maneira independente.

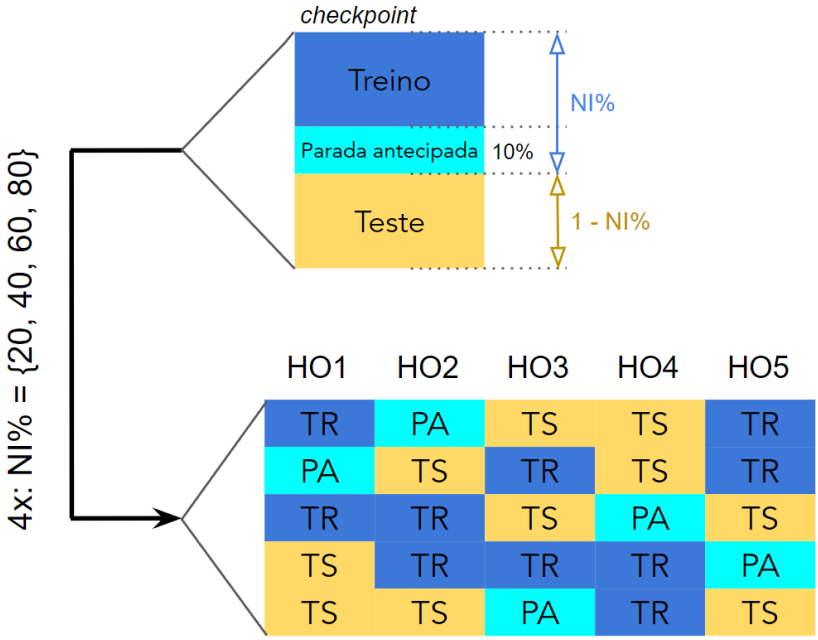
Como uma tentativa de minimizar o viés da escolha dos conjuntos de treino e teste, utiliza-se na avaliação de desempenho dos modelos um esquema de *holdout* (HO) com repetições. Conforme ilustrado na Figura 4, em cada processo completo de *holdout* com cinco repetições, testa-se um nível de informação NI diferente. Dessa forma, os resultados são avaliados quando há mais ou menos informação fornecida ao modelo na fase de treino. Conforme mostra a Figura 4, NI% dos dados - percentual em azul escuro - são usados para treinar os modelos já configurados, sendo que deste total de treino (NI%), 10% - em azul claro - são usados para parada antecipada (*early stopping*). O restante dos dados não utilizado no treino - (100 - NI)% em amarelo - é usado para testar os modelos. A métrica a ser minimizada durante o treinamento e validação é a função de custo de entropia cruzada (*cross-entropy loss function*).

Os modelos são avaliados de duas formas diferentes. Primeiramente, avaliam-se os preditores considerando todas estimativas no conjunto de teste, compilando eventos de todos os usuários (formato padrão de avaliação). Em seguida, avalia-se a capacidade de predição dos modelos “por usuário”. Neste caso, a análise de desempenho dos preditores é realizada com a média de uma métrica m_i avaliada para cada usuário i. Três métricas de avaliação dos preditores são utilizadas neste trabalho: AUC (*area under the ROC curve*), *Precision@k* e *Recall@k*.

4.4 Métricas de avaliação dos preditores

Como os eventos de interação respondente-questão são binários em R_s e em R_c, métricas de problemas de classificação podem ser utilizadas para avaliação dos preditores. Nesse contexto, dentre as métricas mais comuns está o AUC, que mede a área abaixo da curva ROC

Figura 4 – Processo de treino e teste finais via *hold-out* com cinco repetições.



Fonte: Autoria própria.

(Receiver operating characteristic curve), resultando em um valor entre 0 e 1. A curva ROC relaciona a taxa de falsos positivos (*recall*) e a taxa de falsos negativos (*fallout*) para diferentes *thresholds* de classificação de um preditor. Essa métrica, no contexto das pontuações consideradas neste trabalho, mede a probabilidade de um exemplo da classe '0' (pontuações 4 ou 5) escolhido aleatoriamente ter uma estimativa menor de pertencer à classe de interesse, ou seja classe '1' (pontuações 1, 2 ou 3) do que um exemplo da classe '1' aleatório. O AUC é uma métrica mais robusta que a acurácia, tendo uso recomendado principalmente para problemas com classes desbalanceadas (HUANG; LING, 2005), que é o caso dos dados deste trabalho.

Entretanto, a literatura de sistemas de recomendação comumente avalia os preditores utilizando métricas sensíveis à ordem das predições (SCHRÖDER; THIELE; LEHNER, 2011), com foco principalmente nas primeiras posições do *ranking* (que têm prioridade na recomendação). Neste sentido, duas métricas bastante comuns são *precision@k* e *recall@k*. A métrica *precision@k* mede a capacidade de um sistema de recomendar itens relevantes em uma lista de recomendação de tamanho k, enquanto que *recall@k* mede a capacidade de um sistema de capturar itens relevantes em uma lista de recomendação com k itens. As Equações (9) e (10) ilustram esses conceitos, respectivamente.

$$precision = \frac{|itens_{relevantes} \cap itens_{recomendados}|}{|itens_{recomendados}|} \tag{9}$$

$$recall = \frac{|itens_{relevantes} \cap itens_{recomendados}|}{|itens_{relevantes}|} \tag{10}$$

5 RESULTADOS E DISCUSSÃO

Este capítulo apresenta e discute os resultados obtidos da análise exploratória dos dados presentes nos diferentes levantamentos selecionados (Seção 5.1), os melhores parâmetros obtidos para cada modelo utilizando o $cp = 1$ (Seção 5.2), os resultados de predição e ranqueamento dos modelos considerados para dois cenários de avaliação (Seção 5.3) e, por fim, uma discussão dos resultados orientada à construção de sistemas de recomendação (Seção 5.4).

5.1 Análise exploratória dos dados

A Tabela 3 mostra algumas estatísticas extraídas das matrizes R_s e R_c em cada cp.

Tabela 3 – Estatísticas dos dados das matrizes R_s e R_c após o processo de filtragem de cada *checkpoint* (*cp*) para um total de M respondentes e C comentários, sendo M_c o número respondentes que adicionaram comentários.

cp	Número de elementos de cada matriz	C/M_c	M_c/M (%)	Favorabilidade (notas 4 ou 5) (%)
1	87.800	5	27	69
2	170.221	4,9	25,7	73,4
3	217.341	4,9	30,4	81,5
4	272.544	5,5	32,7	82,6
5	228.459	4,9	29,1	81,2
6	275.900	4,4	38,5	83,5

Fonte: Autoria própria.

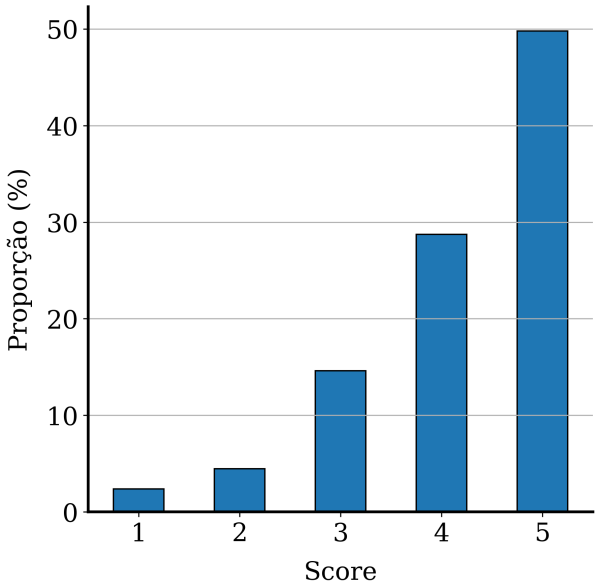
A variável M_c indica o número de respondentes que escreveram ao menos um comentário. O percentual de favorabilidade, ou seja, o percentual de respostas com notas 4 ou 5, é mostrado na última coluna.

A partir da análise da Tabela 3, pode-se observar o seguinte:

- A maioria das respostas é favorável à afirmação da questão (última coluna da tabela). Isso mostra que a maior parte dos respondentes tem uma satisfação média positiva com relação aos aspectos sendo avaliados. Como mostra a Figura 5, existe uma preferência dos respondentes por notas mais altas na agregação dos cps. Esse comportamento se mantém individualmente para cada *cp* e para a grande maioria das questões.

- Conforme mostra a terceira coluna da tabela, em média, aproximadamente 5 comentários são escritos pelos respondentes que adicionaram ao menos um comentário. Entretanto, essa distribuição não é uniforme, como mostra a Figura 6.

Figura 5 – Distribuição dos índices de favorabilidade considerando todos os *cps* nos cinco pontos da escala Likert.



Fonte: Autoria própria.

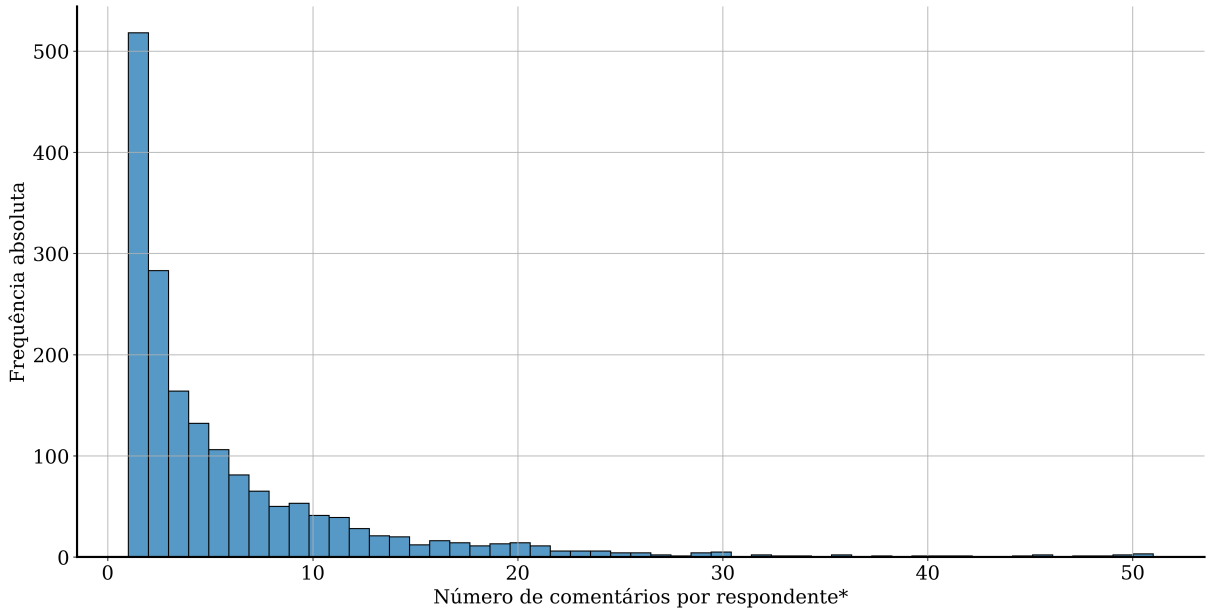
- Outro resultado presente na tabela (quarta coluna) indica que 25% a 40% (30% na média) dos usuários escreveram ao menos um comentário em alguma questão. A Figura 7 mostra que os comentários não se concentram em uma pegunta individual, embora haja preferência dos respondentes em comentar sobre alguns aspectos, como por exemplo aqueles associados às questões Q4 e Q30.

A baixa quantidade de comentários por respondente atrelada ao fato de que os comentários não se concentram em um subconjunto restrito de perguntas, mostra que os respondentes adicionam comentários em perguntas específicas a partir de um comportamento individualizado. Ou seja, o conjunto de perguntas relevantes é restrito e varia para cada respondente. Isso reforça o uso de sistemas de recomendação para a construção de questionários personalizados (e menores) com base na probabilidade de ocorrência do evento de escrita de um comentário de um respondente i em uma questão j.

De maneira similar, observa-se na Figura 8 que não há uma concentração de respostas neutras e não favoráveis em um subconjunto de questões (apesar de haver algumas questões com maior frequência desse evento). Nesse contexto, sistemas de recomendação podem ser úteis para a criação de questionários com foco em questões relacionadas aos aspectos falhos da experiência do colaborador.

Em contrapartida, conforme mostrado na Tabela 1, há um aumento médio do número de questões ao longo de tempo. Como discutido na introdução do trabalho, o aumento do número de perguntas aumenta a carga cognitiva do respondente necessária para a finalizar o questionário podendo gerar perda de informação relevante.

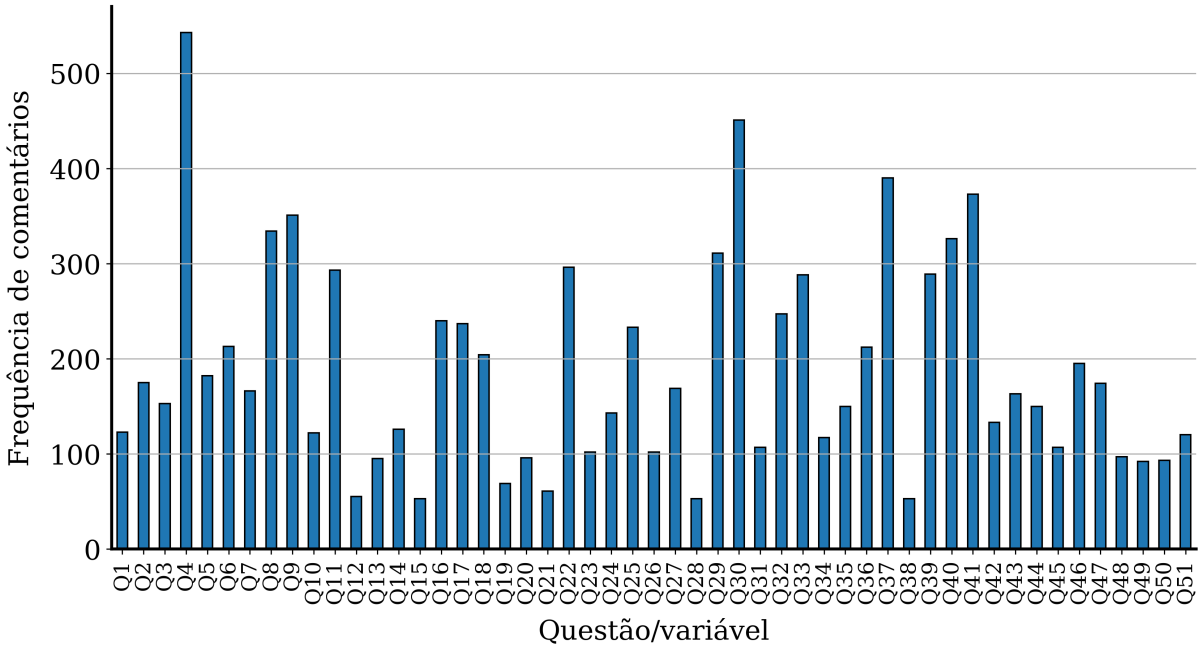
Figura 6 – Número de comentários por respondente que adicionou algum comentário no questionário do $cp = 4$. Em média, aproximadamente 5 comentários são escritos por respondentes que deixaram comentários.



Fonte: Autoria própria.

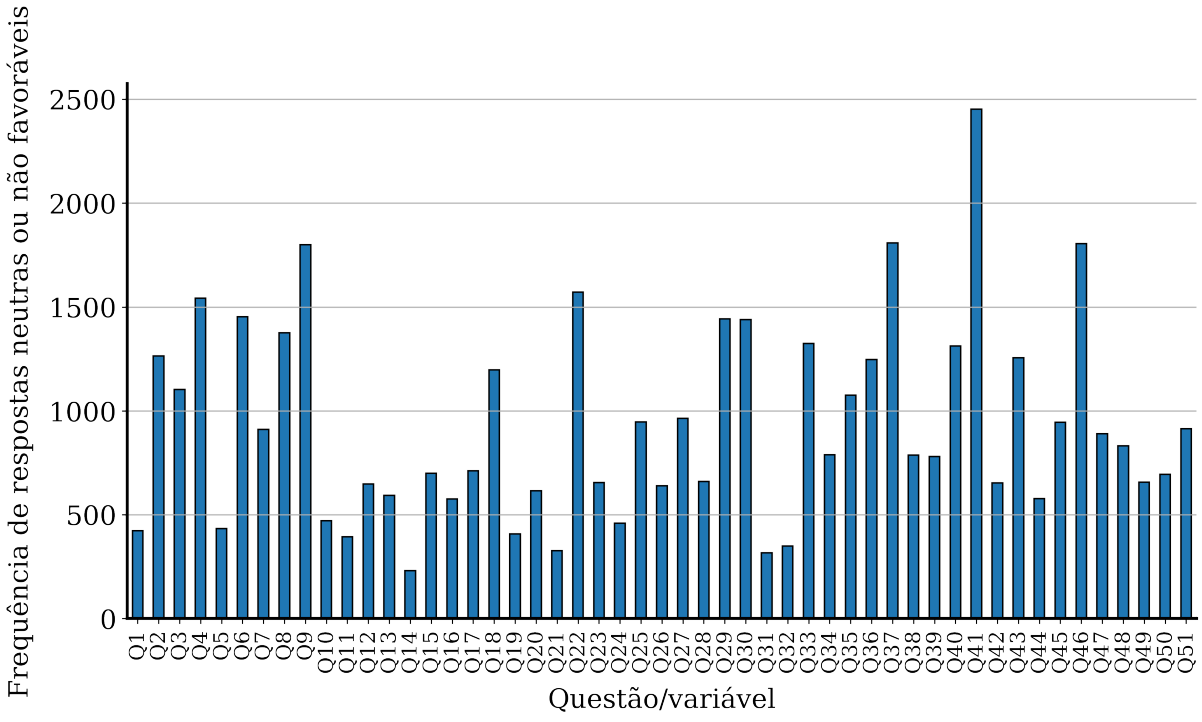
A Figura 9 mensura a similaridade dos levantamentos utilizados em termos de questões e respondentes. As linhas verdes superiores às demais mostram que maioria das questões e dos respondentes se mantém do $cp = X$ para o $cp = Y$ seguinte. As linhas laranjas mostram o número respondentes e o número de questões que estão presentes no $cp = Y$ que não estão presentes no $cp = X$ (as linhas azuis indicam o caso inverso). Em suma, para todos os pares de *cps* adjacentes, o número de respondentes e o número de questões em comum (linha verde) é maior que o número de questões e de respondentes novos (linha laranja, que representa situações de *cold-start*). Em média, essa diferença é de 192% para os respondentes e de 236% para as questões.

Figura 7 – Número de comentários registrados por questão do *checkpoint* $cp = 4$. Os comentários abrangem todas as questões em uma distribuição não uniforme. Isso também acontece para os demais *cps*.



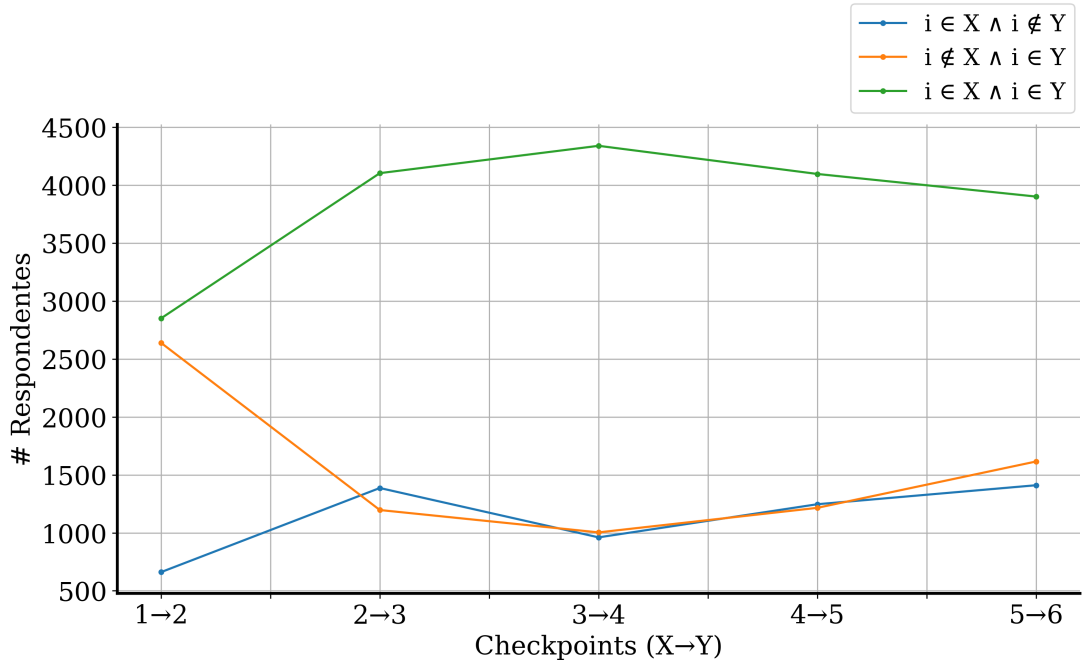
Fonte: Autoria própria.

Figura 8 – Número de respostas neutras ou não favoráveis por questão do $cp = 4$. Esses eventos acontecem para todas as questões em maior ou menor grau. O mesmo padrão é observado nos demais *cps*.

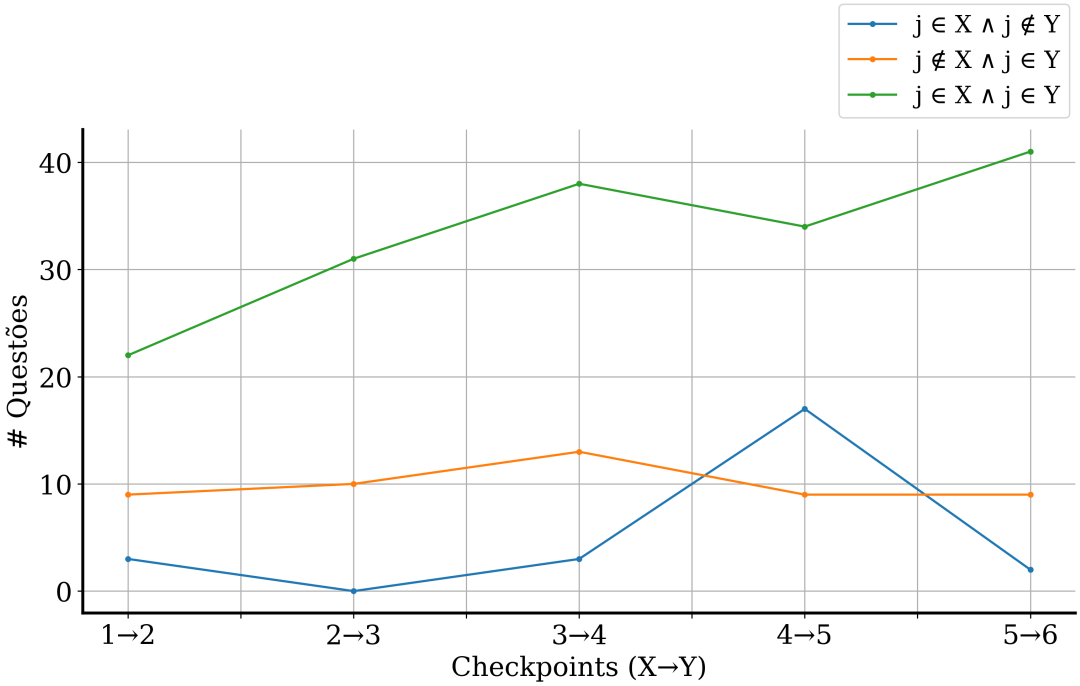


Fonte: Autoria própria.

Figura 9 – Interseção entre os conjuntos de respondentes e de questões para cada par (X, Y) de *checkpoints* adjacentes. As linhas verdes mostram o número de respondentes e o número de questões que se mantém de um *cp* para o outro. As linhas laranjas mostram o número respondentes e o número de questões que estão presentes apenas no *cp* mais recente ($cp = Y$); as linhas azuis indicam o caso inverso.



(a) Número de respondentes para cada par de *cps* adjacentes.



(b) Número de questões para cada par de *cps* adjacentes.

Fonte: Autoria própria.

5.2 Resultados da configuração dos modelos

O procedimento de configuração baseada em dados, descrito na Seção 4.2, foi realizado de maneira independente para as matrizes R_s e R_c, considerando apenas os dados do $cp = 1$. Note que os modelos de predição II e *baseline* não possuem parâmetros de configuração.

Os parâmetros de configuração vencedores da busca aleatória e validação cruzada para cada modelo treinado com os dados da matriz R_s do $cp = 1$ aparecem sublinhados na Tabela 4.

Tabela 4 – Lista de valores e vencedores sublinhados da etapa de configuração dos modelos para a matriz R_s.

Parâmetro	MF	LMF	NCF
k	{16, 32, 64}	{16, 32, 64, 128}	{16, 32, 64, 128}
N_{it}	{2, 4, 8}	{32, 64, 128, 256, 512}	{1, 2, 4}
λ	{2, 4, 8, 16}	{8, 16, 32, 64}	-
α	-	{0.5, 1, 2}	{1, 2, 4} ($\times 10^{-4}$)
N_h	-	-	{2, 4}
n_s	-	-	{4, 8, 16}

Fonte: Autoria própria.

A Tabela 5 destaca os parâmetros vencedores dos modelos que foram treinados com os dados da matriz R_c.

Tabela 5 – Lista de valores e vencedores sublinhados da etapa de configuração dos modelos para a matriz R_c.

Parâmetro	MF	LMF	NCF
k	{16, 32, 64}	{16, 32, 64, 128}	{16, 32, 64, 128}
N_{it}	{2, 4, 8}	{32, 64, 128, 256, 512}	{1, 2, 4}
λ	{2, 4, 8, 16}	{8, 16, 32, 64}	-
α	-	{0.5, 1, 2}	{1, 2, 4} ($\times 10^{-4}$)
N_h	-	-	{2, 4}
n_s	-	-	{4, 8, 16}

Fonte: Autoria própria.

Pelas duas tabelas verifica-se que, com exceção do modelo LMF para os dados de R_c, fatores de dimensão de no máximo 32 são suficientes para modelar os respondentes e as questões pelos três modelos. Cada componente do vetor w_i e do vetor u_j é um aspecto latente utilizado em maior ou menor grau para o cálculo do escore de predição $\hat{r}_{ij}$. O fato de que são necessários no máximo 32 componentes pode indicar que a quantidade de aspectos que influenciam na decisão de um respondente em ser ou não favorável (R_s) e em escrever ou não um comentário (R_c) não é muito alta. Pressupõe-se um número reduzido de aspectos, pois questões de um questionário geralmente são correlacionadas (ZHANG *et al.*, 2020). De fato, o coeficiente Alfa de Cronbach, índice geralmente utilizado para avaliar o grau de consistência de

um questionário a partir da estimativa da "correlação média" das respostas (HORA; MONTEIRO; ARICA, 2010), é de 0.89 para os dados de R_s e de 0.92 para os dados de R_c (cálculo realizado com as respostas do $cp = 1$). Esses valores apontam uma alta confiabilidade do questionário, significando que as respostas possuem alta correlação entre si.

Em contrapartida, em problemas mais complexos espera-se a necessidade de fatores com maior número de componentes. Por exemplo, Bell e Koren (2007) apresentam bons resultados de RMSE para um modelo de fatoração de matrizes com $k = 60$ cuja função de perda é similar à da Equação 2 para a predição de notas de avaliação de filmes dadas por espectadores. A maior dimensionalidade pode ser explicada pela grande variedade de temas de um catálogo com mais de 17,000 filmes utilizado para treino e teste.

As Tabelas 4 e 5 mostram também que o número N_{it} de iterações ou épocas necessárias para convergência das configurações vencedoras se manteve baixo, sobretudo para ao modelo NCF. Em ambos os problemas, a melhor configuração do modelo MF utilizou o número máximo de 8 épocas, o que pode indicar um gargalo da parametrização e consequente restrição de performance desse modelo. Para ambos os problemas (R_s e R_c), a melhor configuração do modelo NCF possui uma arquitetura com $N_h = 4$ camadas escondidas, com a última possuindo $n_s = 8$ neurônios, ou seja, 4 camadas com 64, 32, 16 e 8 neurônios da entrada para a saída, respectivamente.

A Tabela 6 mostra os resultados sobre o comportamento dos modelos na etapa de configuração. Primeiramente, nota-se que os tempos médios de execução de cada modelo foram

Tabela 6 – Resultados da entropia cruzada (*cross-entropy loss*) obtida na configuração dos modelos MF, LMF e NCF para R_s e R_c usando $cp = 1$. O tempo em segundos é o tempo médio de execução das 50 configurações testadas. O valor mínimo indica o resultado da melhor configuração (menor custo observado no conjunto de validação).

	R_s			R_c		
Entropia cruzada	MF	LMF	NCF	MF	LMF	NCF
Média	1,99	1,87	0,61	0,30	1,76	0,27
Mediana	0,56	0,96	0,61	0,15	0,99	0,18
Desvio	2,47	1,39	0,03	0,33	1,21	0,17
Mínimo	0,44	0,86	0,56	0,12	0,91	0,14
Tempo médio (s)	21,2	13,9	15,4	17,0	13,5	16,3

Fonte: Autoria própria.

similares para ambos os problemas. Isso significa que os parâmetros candidatos da Tabela 2 são equilibrados do ponto de vista de esforço computacional. O modelo LMF possui menor tempo médio de execução. Entretanto, as Tabelas 4 e 5 mostram que as configurações vencedoras desse modelo se restringiram a 64 épocas (de um máximo de 512). Ou seja, maiores tempos computacionais para esse modelo não se traduziram e uma melhor capacidade de predição.

Comparando-se o valor mínimo das configurações no conjunto de validação (20% dos dados do $cp = 1$), nota-se que os modelos MF e NCF apresentam os melhores resultados (menores valores do custo de entropia cruzada ou L_{ce}). Para ambos os modelos, a melhor

performance acontece com os dados de R_c. Apesar do modelo NCF apresentar rodadas mais consistentes (menor desvio), o modelo MF apresenta os menores mínimos. Para todos os modelos, a média da entropia cruzada está acima da mediana, o que indica uma distribuição de custo com cauda à direita. Isso significa que existem algumas configurações com performance muito baixa. Um exemplo é o caso da configuração $k = 32$, $N_{it} = 2$ e $\lambda = 2$ para o modelo MF, que resultou em $L_{ce} = 8.38$ (um caso claro de *underfitting*).

Configurações com desempenhos extraordinariamente baixos podem explicar o desvio especialmente alto do modelo MF com dados de R_s, como também os altos desvios de performance do modelo LMF. Provavelmente, valores mais assertivos para os parâmetros poderiam ser definidos de forma a evitar essas configurações. Entretanto, o procedimento de configuração se mostra efetivo quando observados os valores de performance dos modelos considerados no conjunto de teste, conforme mostra a Tabela 7.

Tabela 7 – Valores de AUC das melhores configurações dos modelos MF, LMF e NCF para R_s e R_c nos 10% dos dados de teste do $cp = 1$.

teste de configuração ($cp1$)			
R_s	AUC		
	MF	LMF	NCF
	0,870	0,823	0,828
R_c	AUC		
	MF	LMF	NCF
	0,948	0,935	0,937

Fonte: Autoria própria.

Para essa finalidade, avaliou-se a capacidade de generalização dos modelos configurados. Mais especificamente, as melhores configurações mostradas nas Tabelas 4 e 5 foram avaliadas em 10% dos dados separados para teste do $cp = 1$ (vide região amarela da figura 2).

Os resultados desse experimento, que são mostrados na Tabela 7 em função da métrica AUC, indicam que os três modelos conseguem generalizar o aprendizado dos eventos respondente-questão, obtendo valores altos de AUC tanto para R_s quanto para R_c. Assim como acontece no conjunto de validação da etapa de configuração (valores *mínimos* da Tabela 6), os melhores resultados acontecem para o modelo MF usando o conjunto de dados R_c.

5.3 Resultados da comparação de desempenho dos preditores

Esta seção, além de considerar os modelos II e *baseline*, discute também os resultados dos modelos de predição utilizando os cinco *checkpoints* restantes quando se extrai o $cp = 1$ de configuração. Conforme descrito na Seção 4.3, os modelos de predição são avaliados de duas formas. Primeiramente, avaliam-se os modelos considerando todo o conjunto de teste (desempenho global) seguindo um formato padrão de avaliação de desempenho de preditores. Em seguida, avaliam-se os modelos por usuário e reporta-se a média dos desempenhos considerando todos os usuários. Em ambas as formas de avaliação, o processo de *holdout* com cinco

repetições é realizado de maneira independente para os quatro diferentes níveis de informação NI, considerando cada *checkpoints* $cp = 2, \ldots, 6$ e cada matriz respondente-questão R_s e R_c.

5.3.1 Resultados de desempenho global

As Tabelas 8 e 9 mostram os resultados de AUC dos modelos treinados com os dados de R_s e R_c respectivamente, considerando o primeiro tipo de avaliação.

Tabela 8 – Desempenho (global) de média e desvio padrão (subscrito) dos valores de AUC obtidos na predição de eventos de favorabilidade (matriz R_s) pelos algoritmos II, LMF, MF, NCF e *baseline* para cada *checkpoint* cp e nível NI de informação com *hold-out* de cinco repetições, usando $cp = 1$ para configuração. Células coloridas indicam o melhor desempenho médio para um dado cp e NI.

AUC global - matriz R_s						
cp	***NI* (%)**	II	MF	LMF	NCF	*Baseline*
2	**20**	$0{,}8017_{0{,}0014}$	$0{,}7634_{0{,}0019}$	$0{,}76_{0{,}0013}$	$0{,}7521_{0{,}0026}$	$0{,}6508_{0{,}0004}$
	40	$0{,}828_{0{,}0007}$	$0{,}8228_{0{,}001}$	$0{,}8104_{0{,}0004}$	$0{,}8128_{0{,}0009}$	$0{,}6511_{0{,}001}$
	60	$0{,}8394_{0{,}0011}$	$0{,}847_{0{,}0016}$	$0{,}8263_{0{,}0012}$	$0{,}8312_{0{,}0015}$	$0{,}6513_{0{,}002}$
	80	$0{,}8431_{0{,}0024}$	$0{,}8572_{0{,}0017}$	$0{,}8313_{0{,}0025}$	$0{,}8376_{0{,}0023}$	$0{,}6514_{0{,}0023}$
3	**20**	$0{,}8232_{0{,}0009}$	$0{,}7933_{0{,}001}$	$0{,}7821_{0{,}0023}$	$0{,}7876_{0{,}0011}$	$0{,}6703_{0{,}0011}$
	40	$0{,}8474_{0{,}0008}$	$0{,}8423_{0{,}001}$	$0{,}8296_{0{,}0014}$	$0{,}8321_{0{,}0007}$	$0{,}67_{0{,}0009}$
	60	$0{,}8572_{0{,}0018}$	$0{,}8591_{0{,}0021}$	$0{,}8439_{0{,}0015}$	$0{,}85_{0{,}0021}$	$0{,}6716_{0{,}0036}$
	80	$0{,}8608_{0{,}001}$	$0{,}8634_{0{,}0028}$	$0{,}8489_{0{,}0013}$	$0{,}8562_{0{,}0015}$	$0{,}6746_{0{,}0049}$
4	**20**	$0{,}8429_{0{,}0005}$	$0{,}8163_{0{,}0029}$	$0{,}8038_{0{,}0014}$	$0{,}8107_{0{,}0023}$	$0{,}668_{0{,}0006}$
	40	$0{,}8637_{0{,}0007}$	$0{,}8636_{0{,}0011}$	$0{,}8456_{0{,}0015}$	$0{,}8495_{0{,}0009}$	$0{,}6684_{0{,}0005}$
	60	$0{,}871_{0{,}0009}$	$0{,}8757_{0{,}0016}$	$0{,}8573_{0{,}0007}$	$0{,}8634_{0{,}0011}$	$0{,}6684_{0{,}0022}$
	80	$0{,}875_{0{,}0012}$	$0{,}8788_{0{,}0017}$	$0{,}8621_{0{,}001}$	$0{,}8706_{0{,}0025}$	$0{,}6682_{0{,}0018}$
5	**20**	$0{,}8376_{0{,}0012}$	$0{,}8165_{0{,}0015}$	$0{,}8039_{0{,}0018}$	$0{,}809_{0{,}0023}$	$0{,}6581_{0{,}0004}$
	40	$0{,}8582_{0{,}0008}$	$0{,}8608_{0{,}0009}$	$0{,}8402_{0{,}0014}$	$0{,}8436_{0{,}0007}$	$0{,}659_{0{,}0006}$
	60	$0{,}8662_{0{,}0008}$	$0{,}8778_{0{,}0006}$	$0{,}8515_{0{,}001}$	$0{,}8587_{0{,}0016}$	$0{,}659_{0{,}0019}$
	80	$0{,}87_{0{,}0018}$	$0{,}8844_{0{,}0013}$	$0{,}8566_{0{,}0019}$	$0{,}8659_{0{,}0015}$	$0{,}6589_{0{,}0027}$
6	**20**	$0{,}8525_{0{,}0008}$	$0{,}8272_{0{,}0019}$	$0{,}813_{0{,}0026}$	$0{,}8212_{0{,}0021}$	$0{,}6877_{0{,}0007}$
	40	$0{,}8727_{0{,}0014}$	$0{,}8729_{0{,}0006}$	$0{,}8549_{0{,}0018}$	$0{,}859_{0{,}0009}$	$0{,}6883_{0{,}0012}$
	60	$0{,}8787_{0{,}0009}$	$0{,}886_{0{,}0015}$	$0{,}8644_{0{,}0009}$	$0{,}8715_{0{,}001}$	$0{,}6883_{0{,}0008}$
	80	$0{,}8842_{0{,}0011}$	$0{,}8932_{0{,}0014}$	$0{,}8711_{0{,}001}$	$0{,}8793_{0{,}0014}$	$0{,}6887_{0{,}0025}$

Fonte: Autoria própria.

Os resultados da Tabela 8 mostram que é possível aprender a discriminar notas baixas dadas por funcionários em grandes pesquisas. Já os resultados da Tabela 9 mostram que é possível aprender a discriminar os casos de interações respondente-questão que geram comentários.

Embora não tenham sido encontradas diferenças significativas entre os desempenhos dos modelos para um mesmo cp e NI, uma discussão direta pode ser feita. Por exemplo, nota-se que valores médios de AUC do modelo *baseline* são inferiores aos dos demais modelos, o

Tabela 9 – Desempenho (global) de média e desvio padrão (subscrito) dos valores de AUC obtidos na predição de eventos de comentários (matriz R_c) pelos algoritmos II, LMF, MF, NCF e *baseline* para cada *checkpoint* cp e nível NI de informação com *hold-out* de cinco repetições. Sublinhados estão os casos em que o modelo LMF apresenta melhores resultados médios para um *cp* com um menor percentual de NI, em um provável caso de *underfitting*. Células coloridas indicam o melhor desempenho médio para um dado cp e NI.

AUC global - matriz R_c						
cp	***NI* (%)**	II	MF	LMF	NCF	*Baseline*
2	**20**	$0{,}8579_{0{,}0022}$	$0{,}8228_{0{,}0015}$	$0{,}7897_{0{,}0059}$	$0{,}8141_{0{,}0026}$	$0{,}6005_{0{,}0013}$
	40	$0{,}9092_{0{,}0025}$	$0{,}8977_{0{,}0048}$	$0{,}8876_{0{,}005}$	$0{,}8961_{0{,}0048}$	$0{,}6017_{0{,}0038}$
	60	$0{,}9248_{0{,}0031}$	$0{,}9209_{0{,}0037}$	$0{,}9145_{0{,}0058}$	$0{,}9204_{0{,}0041}$	$0{,}6058_{0{,}003}$
	80	$0{,}9325_{0{,}001}$	$0{,}9277_{0{,}0025}$	$0{,}9251_{0{,}0025}$	$0{,}9296_{0{,}0016}$	$0{,}6039_{0{,}0065}$
3	**20**	$0{,}8516_{0{,}0051}$	$0{,}8274_{0{,}0034}$	$0{,}7874_{0{,}0159}$	$0{,}823_{0{,}0046}$	$0{,}6275_{0{,}002}$
	40	$0{,}9014_{0{,}0034}$	$0{,}8928_{0{,}003}$	$0{,}8785_{0{,}0076}$	$0{,}8919_{0{,}0035}$	$0{,}6313_{0{,}0044}$
	60	$0{,}9173_{0{,}0019}$	$0{,}9144_{0{,}003}$	$0{,}8795_{0{,}0245}$	$0{,}9138_{0{,}0031}$	$0{,}6325_{0{,}0038}$
	80	$0{,}9224_{0{,}0048}$	$0{,}9163_{0{,}0041}$	$0{,}8846_{0{,}0054}$	$0{,}919_{0{,}0034}$	$0{,}633_{0{,}0057}$
4	**20**	$0{,}8604_{0{,}0022}$	$0{,}84_{0{,}0034}$	$0{,}7241_{0{,}0113}$	$0{,}8372_{0{,}0032}$	$0{,}6579_{0{,}0007}$
	40	$0{,}9033_{0{,}0013}$	$0{,}8971_{0{,}0017}$	$\underline{0{,}878_{0{,}0091}}$	$0{,}8962_{0{,}002}$	$0{,}6563_{0{,}0025}$
	60	$0{,}919_{0{,}0013}$	$0{,}918_{0{,}0015}$	$\underline{0{,}8602_{0{,}0117}}$	$0{,}9166_{0{,}0012}$	$0{,}6605_{0{,}0065}$
	80	$0{,}925_{0{,}0035}$	$0{,}9191_{0{,}0056}$	$0{,}8831_{0{,}0079}$	$0{,}9234_{0{,}0035}$	$0{,}6603_{0{,}0069}$
5	**20**	$0{,}8655_{0{,}0047}$	$0{,}8437_{0{,}0063}$	$0{,}8145_{0{,}0095}$	$0{,}839_{0{,}0052}$	$0{,}6405_{0{,}0013}$
	40	$0{,}908_{0{,}0011}$	$0{,}8998_{0{,}0012}$	$\underline{0{,}8867_{0{,}0032}}$	$0{,}8984_{0{,}0012}$	$0{,}6454_{0{,}0009}$
	60	$0{,}9232_{0{,}0026}$	$0{,}9209_{0{,}0022}$	$\underline{0{,}8832_{0{,}0251}}$	$0{,}919_{0{,}0028}$	$0{,}6468_{0{,}0031}$
	80	$0{,}93_{0{,}0016}$	$0{,}9257_{0{,}0019}$	$0{,}8938_{0{,}0051}$	$0{,}9263_{0{,}0029}$	$0{,}6428_{0{,}0026}$
6	**20**	$0{,}8651_{0{,}0042}$	$0{,}8548_{0{,}0036}$	$0{,}7238_{0{,}0159}$	$0{,}8515_{0{,}0029}$	$0{,}7094_{0{,}0019}$
	40	$0{,}9084_{0{,}0018}$	$0{,}9051_{0{,}0021}$	$\underline{0{,}8859_{0{,}0085}}$	$0{,}9012_{0{,}002}$	$0{,}7115_{0{,}0028}$
	60	$0{,}9226_{0{,}0014}$	$0{,}922_{0{,}0016}$	$\underline{0{,}8605_{0{,}0126}}$	$0{,}915_{0{,}001}$	$0{,}7166_{0{,}0042}$
	80	$0{,}9288_{0{,}0011}$	$0{,}9277_{0{,}0014}$	$0{,}8916_{0{,}005}$	$0{,}9229_{0{,}0018}$	$0{,}7164_{0{,}004}$

Fonte: Autoria própria.

que mostra vantagem dos modelos de filtragem colaborativa. Os baixos desvios encontrados reforçam esse resultado, indicando robustez dos modelos para diferentes conjuntos de treino.

Ao comparar os resultados dos modelos de predição de filtragem colaborativa, nota-se uma ligeira vantagem dos modelos MF e II frente aos modelos NCF e LMF. Isso ocorre para a grande maioria das combinações de *cp* e NI. A vantagem do modelo MF com relação ao modelo NCF é consistente com a discussão teórica e empírica apresentada por Rendle *et al.* (2020b). A baixa performance relativa dos modelos NCF e LMF provavelmente se deve ao fato de serem projetados para problemas implícitos e, portanto, sofrem por não serem capazes de discriminar valores zero (ou seja, pontuações altas em R_s e inexistência de comentários em R_c) de valores ausentes. Isso acontece pois o treinamento de ambos os modelos inclui os pares respondente-questão do conjunto de teste como eventos da classe zero.

Os modelos II e MF possuem desempenhos similares em ambos os problemas, apesar do modelo MF parecer se beneficiar mais com o aumento do NI para o problema de predição

de favorabilidade R_s, enquanto que o modelo II tem melhor desempenho nos dados de R_c. Ressalta-se que os resultados do algoritmo II são muito satisfatórios, dada sua simplicidade conceitual e seu procedimento de treinamento rápido e direto (comparado aos demais). Além disso, acredita-se que o bom desempenho do MF se deve ao uso de termos de viés que desacoplam o verdadeiro sinal comportamental (aquele a ser aprendido) dos padrões de ruído das classificações disponíveis.

Em geral, os modelos de filtragem colaborativa apresentam melhores resultados quando treinados com mais dados. Ou seja, o AUC aumenta quanto maior é o NI e para *checkpoints* com maior número de eventos (vide coluna 2 da Tabela 3 que descreve os *datasets*). Isso acontece para ambos os problemas considerados (R_s e R_c). Entretanto, um comportamento atípico ocorre para o modelo LMF com os dados de R_c, em que a presença de maiores níveis de informação (maior conjunto de treino) faz com que a performance média do modelo diminua. Esses casos aparecem sublinhados na Tabela 9. Isso pode indicar uma limitação desse modelo em se ajustar aos dados de treinamento, provavelmente ocasionada pala configuração de seus parâmetros realizada com os dados do $cp = 1$ em um caso *underfitting*. Apesar do melhor desempenho observado para maiores valores de NI, é interessante notar que os resultados são bastante positivos mesmo para o caso em que $NI = 20\%$. Ou seja, uma amostra relativamente pequena das interações de cada respondente, apenas 5 de 25 para o $cp = 2$, por exemplo, é suficiente para os modelos de filtragem colaborativa conseguirem produzir boas estimativas para os eventos da classe '1' em R_s e R_c.

Além disso, os desempenhos dos modelos de fatoração são compatíveis com os dados de desempenho mostrados na Tabela 7. Isso significa que as configurações ajustadas com dados do $cp = 1$ se mantiveram consistentes mesmo quando aplicadas a dados de eventos de outros *checkpoints*, com características diferentes do $cp = 1$. O *checkpoint* mais distinto é o $cp = 6$, que possui três vezes o número de eventos do cp usado para configuração, além de possuir dados desbalanceados em R_s, com 16,5% dos eventos pertencentes a classe '1', contra 31% no $cp = 1$. Os bons resultados de AUC mostram que os parâmetros ajustados no processo de configuração são pouco sensíveis às características da matriz R_s, o que é positivo.

Comparando-se os resultados das Tabelas 8 e 9, notam-se valores mais altos de AUC para o problema de predição da existência de comentários (R_c). Acredita-se que esse é um resultado interessante, uma vez que é intuitivamente mais difícil modelar o comportamento que engaja um respondente a escrever comentários (por opção) do que avaliar negativamente um aspecto da empresa. Entretanto, essa é uma visão simplista do comportamento humano no preenchimento de questões relativas à experiência passada que precisa de uma averiguação mais aprofundada. Por exemplo, um dado quantitativo que contraria essa suposição anterior são os valores de Alfa de Cronbach (reportados na Seção 5.2), de 0.89 para os dados de R_s e 0.92 para os dados de R_c. Esses valores mostram que existem correlações em R_c que podem ser exploradas até mais profundamente pelos preditores do que as de R_s.

A Figura 10 mostra as estimativas para o modelo MF e o *baseline* para o problema de predição de favorabilidade (dados de R_s) com os dados do $cp = 2$ com $NI = 40\%$. Escolheu-se o MF para essa comparação, pois esse é o algoritmo baseado em modelo mais frequentemente usado. O topo da Figura 10(a) mostra as estimativas de predição obtidas pelo modelo MF. À esquerda, têm-se as estimativas para os dados de teste da primeira rodada do *holdout* do $cp = 2$, considerando apenas os dez primeiros respondentes e as dez primeiras questões deste questionário. As células sem valor representam eventos utilizados no treino, enquanto que os valores em vermelho representam os eventos em que $r_{ij} = 1$. Para os dados de R_s, esses são os casos em que o respondente i deu nota 1, 2 ou 3 para a questão j. À direita, tem-se as dez interações respondente-questão com maior estimativa de predição. A parte inferior da Figura 10(b) apresenta esses mesmos dados considerando as estimativas do modelo *baseline*.

Figura 10 – Estimativas (à esquerda) dos modelos MF (a) e *baseline* (b) para o problema de favorabilidade (R_s) para os dez primeiros respondentes e dez primeiras questões do conjunto de teste da primeira rodada de *holdout* do $cp = 2$. Os valores inexistentes são eventos usados para treino. À direita, têm-se as dez interações respondente-questão com as maiores estimativas de predição. Os valores em vermelho se referem aos eventos da classe '1' ($r_{ij} = 1$); nesse caso, respostas 1, 2 ou 3 na escala Likert de 5 pontos.

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
R1	0.051515	0.220626	-	-	-	0.192794	0.083651	0.167589	0.113851	-
R2	0.140021	0.401967	-	-	-	-	0.196461	0.251353	-	0.146240
R3	-	0.259914	-	-	0.022853	-	0.104879	0.187059	0.117492	-
R4	-	0.619900	-	-	-	-	0.258585	0.331007	0.279920	0.291849
R5	0.185304	-	0.275487	-	-	0.318711	0.161392	0.248089	-	-
R6	0.051189	0.232068	0.156375	-	-	-	-	0.176423	0.127990	0.103237
R7	0.197111	-	-	0.327668	0.144996	-	-	-	0.225577	-
R8	-	-	-	0.470917	0.212000	0.677058	0.492508	-	0.346283	0.504518
R9	0.294023	0.656539	0.517902	0.564226	0.230016	0.538191	0.539238	0.405675	0.427024	0.338661
R10	-	0.302504	0.217230	-	0.014166	0.311296	-	0.224710	-	-

Top-10	Estimativa
R8, Q6	0.677058
R9, Q2	0.656539
R4, Q2	0.619900
R9, Q4	0.564226
R9, Q7	0.539238
R9, Q6	0.538191
R9, Q3	0.517902
R8, Q10	0.504518
R8, Q7	0.492508
R8, Q4	0.470917

(a) Estimativas do modelo MF.

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
R1	0.161765	0.415747	-	-	-	0.366337	0.227449	0.265049	0.217869	-
R2	0.161765	0.415747	-	-	-	-	0.227449	0.265049	-	0.249062
R3	-	0.415747	-	-	0.100325	-	0.227449	0.265049	0.217869	-
R4	-	0.415747	-	-	-	-	0.227449	0.265049	0.217869	0.249062
R5	0.161765	-	0.302020	-	-	0.366337	0.227449	0.265049	-	-
R6	0.161765	0.415747	0.302020	-	-	-	-	0.265049	0.217869	0.249062
R7	0.161765	-	-	0.377026	0.100325	-	-	-	0.217869	-
R8	-	-	-	0.377026	0.100325	0.366337	0.227449	-	0.217869	0.249062
R9	0.161765	0.415747	0.302020	0.377026	0.100325	0.366337	0.227449	0.265049	0.217869	0.249062
R10	-	0.415747	0.302020	-	0.100325	0.366337	-	0.265049	-	-

Top-10	Estimativa
R3, Q2	0.415747
R9, Q2	0.415747
R6, Q2	0.415747
R1, Q2	0.415747
R4, Q2	0.415747
R10, Q2	0.415747
R2, Q2	0.415747
R7, Q4	0.377026
R8, Q4	0.377026
R9, Q4	0.377026

(b) Estimativas do modelo *baseline*.

Fonte: Autoria própria.

As listas à direita da Figura 10 reúnem as interações respondente-questão mais prováveis de pertencerem à classe '1', segundo os respectivos modelos. Por exemplo, de todas as interações respondente-questão (do escopo reduzido) da Figura 10, o modelo MF tem mais confiança de que o respondente R8 é propenso a dar notas baixas à questão Q6 (o que de fato ocorreu). Já para o modelo *baseline*, as estimativas de predição são constantes para

uma mesma questão: $\hat{r}_{ij}$ para o item j é o mesmo para todo usuário i. Conforme explicado na Seção 3.6, sabe-se que 41,5% das respostas da questão Q2 nos dados de treino da primeira rodada de *holdout* do $cp = 2$ e $NI = 40\%$ são 1, 2 ou 3. Esse modelo heurístico é caracterizado por possuir estimativas mais altas concentradas em um único item, o que implica em baixa diversidade de itens em uma lista de recomendação que prioriza as maiores estimativas.

Entretanto, nota-se que as maiores estimativas do modelo MF indicam uma preferência desse modelo em recomendar questões aos respondentes R8 e R9 (em 9 dos top-10 eventos). Esses respondentes provavelmente dão notas mais baixas que os demais. O modelo MF pode capturar esse comportamento a partir do ajuste apropriado do termo de viés b_i da Equação (4). Diferentemente do que acontece para o modelo *baseline*, esse é um comportamento aprendido durante o treino para reduzir a função de custo definida na Equação (3).

A Figura 10 também auxilia na justificativa da escolha da métrica AUC para avaliação dos modelos. A maior estimativa do modelo MF é aproximadamente 0,68 - distante do valor real 1 (*ground-truth*). Contudo, para preditores usados em sistemas de recomendação, o objetivo final é que os eventos da classe de interesse (classe '1' que neste trabalho envolve as opiniões 1, 2 e 3) estejam nas primeiras posições do *ranking* de estimativas. Como o AUC avalia a capacidade do modelo de aumentar a taxa de verdadeiros positivos, sem aumento da taxa dos falsos positivos, é uma métrica que captura essa capacidade de ordenação. Entretanto, o AUC faz essa avaliação considerando todo o conjunto de teste, o que inclui as estimativas nas últimas posições. Schröder, Thiele e Lehner (2011) propõem uma versão alternativa do AUC, o LAUC (*Limited Area Under the Curve*), que avalia o desempenho de modelos considerando apenas as maiores estimativas. Esta avaliação alternativa está fora do escopo do trabalho mas, futuramente, espera-se avaliar os preditores utilizando o LAUC.

Finalmente, para o caso da predição de favorabilidade (matriz R_s), um sistema preditor construído para estimar a pontuação na escala Likert (1 a 5) usando métricas como o MSE ou RMSE poderia ter melhores resultados. Bell e Koren (2007) apresentam um modelo de fatoração de matrizes muito similar ao modelo MF apresentado na Equação (4), porém orientado a previsão de notas de avaliação dadas na escala de 1 a 5. A maior granularidade das respostas na escala original de 5 pontos pode auxiliar os modelos para uma detecção mais fina de comportamentos de avaliação de respondentes nos itens de um questionário.

5.3.2 Resultados de desempenho por usuário

Nesta seção, o desempenho dos preditores é avaliado individualmente para cada respondente. Neste caso, uma métrica m_i é obtida para cada respondente i nos dados de teste de

R_s e R_c[1]. Assim como na seção anterior, isso é feito de maneira independente para cada matriz. Os resultados de desempenho dos usuários são combinados utilizando a média aritmética. Isso acontece para cada rodada de *holdout*. Por fim, avalia-se a média aritmética e o desvio das diferentes rodadas.

Neste cenário, é possível avaliar a métrica AUC apenas para respondentes que possuam ao menos um evento positivo e um evento negativo no seu respectivo conjunto de teste. Caso contrário, a métrica é indefinida. Além disso, para conjuntos de teste com uma única classe, as métricas *precision@k* e *recall@k* não geram valores informativos. Dessa forma, decidiu-se por fazer a avaliação dos modelos de predição apenas para respondentes que contenham eventos da classe '1' e classe '0' no conjunto de teste.

As Tabelas 10 e 11 mostram os valores de AUC médios, obtidos como descrito anteriormente para as Tabelas 8 e 9. Comparado com o desempenho global descrito na seção

Tabela 10 – Desempenho (por usuário) de média e desvio padrão (subscrito) dos valores de AUC obtidos na predição de eventos de favorabilidade (matriz R_s) pelos algoritmos II, LMF, MF, NCF e *baseline* para cada *checkpoint* cp e nível NI de informação com *hold-out* de cinco repetições. Células coloridas indicam o melhor desempenho médio para um dado cp e NI.

AUC individual - matriz R_s						
cp	***NI* (%)**	II	MF	LMF	NCF	*Baseline*
2	**20**	$0{,}7087_{0{,}0013}$	$0{,}7207_{0{,}0018}$	$0{,}7052_{0{,}0026}$	$0{,}7175_{0{,}0036}$	$0{,}721_{0{,}0009}$
	40	$0{,}7284_{0{,}0016}$	$0{,}738_{0{,}0024}$	$0{,}7182_{0{,}0008}$	$0{,}7209_{0{,}0015}$	$0{,}7208_{0{,}0015}$
	60	$0{,}732_{0{,}0025}$	$0{,}7545_{0{,}0015}$	$0{,}718_{0{,}0018}$	$0{,}7217_{0{,}0021}$	$0{,}7189_{0{,}0017}$
	80	$0{,}7302_{0{,}0016}$	$0{,}7628_{0{,}0024}$	$0{,}7135_{0{,}0025}$	$0{,}7188_{0{,}0021}$	$0{,}7138_{0{,}0017}$
3	**20**	$0{,}7442_{0{,}0024}$	$0{,}7547_{0{,}0015}$	$0{,}7378_{0{,}0047}$	$0{,}7529_{0{,}0015}$	$0{,}7549_{0{,}0012}$
	40	$0{,}7656_{0{,}0016}$	$0{,}7693_{0{,}002}$	$0{,}7489_{0{,}0025}$	$0{,}7541_{0{,}0015}$	$0{,}7533_{0{,}0015}$
	60	$0{,}7662_{0{,}0033}$	$0{,}7757_{0{,}0034}$	$0{,}7468_{0{,}0026}$	$0{,}7539_{0{,}0018}$	$0{,}749_{0{,}0031}$
	80	$0{,}7646_{0{,}0047}$	$0{,}7729_{0{,}0028}$	$0{,}7434_{0{,}0058}$	$0{,}7519_{0{,}0035}$	$0{,}7445_{0{,}0055}$
4	**20**	$0{,}7619_{0{,}0017}$	$0{,}7637_{0{,}0013}$	$0{,}7429_{0{,}0045}$	$0{,}762_{0{,}0012}$	$0{,}7637_{0{,}0006}$
	40	$0{,}7781_{0{,}0011}$	$0{,}7864_{0{,}0014}$	$0{,}7546_{0{,}0033}$	$0{,}7627_{0{,}0009}$	$0{,}7608_{0{,}0013}$
	60	$0{,}7771_{0{,}0016}$	$0{,}7913_{0{,}0023}$	$0{,}7521_{0{,}0021}$	$0{,}7624_{0{,}0027}$	$0{,}7551_{0{,}0015}$
	80	$0{,}7701_{0{,}0048}$	$0{,}7865_{0{,}0063}$	$0{,}7423_{0{,}0045}$	$0{,}7576_{0{,}0078}$	$0{,}7435_{0{,}0052}$
5	**20**	$0{,}7464_{0{,}0012}$	$0{,}7489_{0{,}0016}$	$0{,}7285_{0{,}0049}$	$0{,}7425_{0{,}0006}$	$0{,}7444_{0{,}0011}$
	40	$0{,}7639_{0{,}0008}$	$0{,}7775_{0{,}0014}$	$0{,}739_{0{,}0022}$	$0{,}7455_{0{,}0018}$	$0{,}7427_{0{,}0009}$
	60	$0{,}7645_{0{,}0024}$	$0{,}7912_{0{,}003}$	$0{,}7358_{0{,}0016}$	$0{,}7491_{0{,}004}$	$0{,}7384_{0{,}0025}$
	80	$0{,}7595_{0{,}0035}$	$0{,}7941_{0{,}0027}$	$0{,}7282_{0{,}0039}$	$0{,}747_{0{,}0026}$	$0{,}729_{0{,}003}$
6	**20**	$0{,}7903_{0{,}0007}$	$0{,}7928_{0{,}0012}$	$0{,}7756_{0{,}0055}$	$0{,}7908_{0{,}0012}$	$0{,}7927_{0{,}0004}$
	40	$0{,}8027_{0{,}0018}$	$0{,}8093_{0{,}0015}$	$0{,}7842_{0{,}0038}$	$0{,}7904_{0{,}0014}$	$0{,}7895_{0{,}0011}$
	60	$0{,}7997_{0{,}003}$	$0{,}8144_{0{,}0027}$	$0{,}7802_{0{,}0027}$	$0{,}7865_{0{,}0035}$	$0{,}7821_{0{,}0025}$
	80	$0{,}7926_{0{,}0037}$	$0{,}8126_{0{,}0034}$	$0{,}7704_{0{,}0038}$	$0{,}7804_{0{,}0038}$	$0{,}7713_{0{,}0032}$

Fonte: Autoria própria.

[1] Como descrito na Seção 4.3, a divisão de treino e teste foi realizada de tal maneira que todos os respondentes tenham dados de eventos no conjunto de teste. Assim, $NI\%$ dos eventos de cada respondente estão no conjunto de treino e validação.

Tabela 11 – Desempenho (por usuário) de média e desvio padrão (subscrito) dos valores de AUC obtidos na predição de eventos de comentários (matriz R_c) pelos algoritmos II, LMF, MF, NCF e *baseline* para cada *checkpoint* cp e nível NI de informação com *hold-out* de cinco repetições. Células coloridas indicam o melhor desempenho médio para um dado cp e NI.

AUC individual - matriz R_c						
cp	***NI* (%)**	II	MF	LMF	NCF	*Baseline*
2	**20**	$0{,}6245_{0{,}0054}$	$0{,}6711_{0{,}0086}$	$0{,}5737_{0{,}0202}$	$0{,}6622_{0{,}0127}$	$0{,}681_{0{,}004}$
	40	$0{,}6633_{0{,}0059}$	$0{,}6804_{0{,}0052}$	$0{,}6241_{0{,}0097}$	$0{,}6754_{0{,}0041}$	$0{,}6772_{0{,}004}$
	60	$0{,}6941_{0{,}0053}$	$0{,}6879_{0{,}0076}$	$0{,}6382_{0{,}0261}$	$0{,}6728_{0{,}0072}$	$0{,}6733_{0{,}0074}$
	80	$0{,}7112_{0{,}0117}$	$0{,}6928_{0{,}0108}$	$0{,}6412_{0{,}0094}$	$0{,}6684_{0{,}0101}$	$0{,}6656_{0{,}0094}$
3	**20**	$0{,}6547_{0{,}0031}$	$0{,}701_{0{,}0017}$	$0{,}5866_{0{,}0427}$	$0{,}6966_{0{,}0023}$	$0{,}7057_{0{,}0039}$
	40	$0{,}6958_{0{,}0057}$	$0{,}7102_{0{,}004}$	$0{,}6437_{0{,}0251}$	$0{,}7063_{0{,}0039}$	$0{,}7051_{0{,}0045}$
	60	$0{,}7228_{0{,}0031}$	$0{,}7147_{0{,}0046}$	$0{,}6252_{0{,}0446}$	$0{,}7019_{0{,}003}$	$0{,}7018_{0{,}0025}$
	80	$0{,}7285_{0{,}0103}$	$0{,}7088_{0{,}0142}$	$0{,}6149_{0{,}0059}$	$0{,}6921_{0{,}0102}$	$0{,}6898_{0{,}0097}$
4	**20**	$0{,}6633_{0{,}0026}$	$0{,}7224_{0{,}0036}$	$0{,}5876_{0{,}0357}$	$0{,}7209_{0{,}0041}$	$0{,}7293_{0{,}0016}$
	40	$0{,}717_{0{,}0061}$	$0{,}7319_{0{,}003}$	$0{,}6648_{0{,}018}$	$0{,}7263_{0{,}0036}$	$0{,}7268_{0{,}0032}$
	60	$0{,}745_{0{,}0021}$	$0{,}7418_{0{,}0034}$	$0{,}601_{0{,}0073}$	$0{,}7258_{0{,}0051}$	$0{,}7254_{0{,}0047}$
	80	$0{,}7518_{0{,}0074}$	$0{,}7389_{0{,}0125}$	$0{,}6167_{0{,}0143}$	$0{,}7173_{0{,}008}$	$0{,}7153_{0{,}0073}$
5	**20**	$0{,}6641_{0{,}0049}$	$0{,}714_{0{,}0026}$	$0{,}6254_{0{,}0205}$	$0{,}7123_{0{,}002}$	$0{,}7206_{0{,}002}$
	40	$0{,}7106_{0{,}0076}$	$0{,}7271_{0{,}0032}$	$0{,}6606_{0{,}0168}$	$0{,}7179_{0{,}004}$	$0{,}7193_{0{,}0042}$
	60	$0{,}7399_{0{,}0056}$	$0{,}7347_{0{,}0067}$	$0{,}6301_{0{,}0437}$	$0{,}7181_{0{,}0049}$	$0{,}7174_{0{,}0054}$
	80	$0{,}7492_{0{,}0064}$	$0{,}7363_{0{,}0074}$	$0{,}6242_{0{,}0171}$	$0{,}7079_{0{,}0106}$	$0{,}7059_{0{,}0106}$
6	**20**	$0{,}7577_{0{,}001}$	$0{,}8081_{0{,}0026}$	$0{,}6998_{0{,}0315}$	$0{,}8068_{0{,}0026}$	$0{,}8121_{0{,}0022}$
	40	$0{,}7954_{0{,}0025}$	$0{,}8111_{0{,}0026}$	$0{,}7529_{0{,}0185}$	$0{,}8064_{0{,}0019}$	$0{,}8074_{0{,}002}$
	60	$0{,}8157_{0{,}0044}$	$0{,}8143_{0{,}0045}$	$0{,}6931_{0{,}0195}$	$0{,}8039_{0{,}0038}$	$0{,}8037_{0{,}0036}$
	80	$0{,}8181_{0{,}0066}$	$0{,}8052_{0{,}0062}$	$0{,}7022_{0{,}0197}$	$0{,}7918_{0{,}0062}$	$0{,}7902_{0{,}006}$

Fonte: Autoria própria.

anterior, verifica-se uma queda no desempenho dos modelos dos algoritmos FC e uma melhora do modelo *baseline* por usuário. Neste caso, o *baseline* torna-se mais competitivo do que no caso global, embora poucos sejam os casos em que ele seja o melhor (células coloridas nas Tabelas 10 e 11).

Isso significa que apesar dos modelos FC conseguirem em uma visão global priorizar interações cujos eventos são da classe de interesse, essa capacidade diminui quando se busca fazer isso por usuário. Essa queda é tal que um sistema heurístico que seleciona as questões que possuem mais eventos da classe '1' é suficiente para entregar performance similar. T O modelo MF é sistematicamente melhor que os outros para o problema com dados de R_s, embora as diferenças não sejam relevantes. Já para o problema com dados de R_c, os melhores resultados concentram-se nos modelos II e MF. A exceção é o modelo LMF que apresenta o pior desempenho em todas as combinações de *cp* e nível de informação com dados de R_c.

Contudo, um comportamento notável ocorre nos dados da Tabela 11, na qual o modelo *baseline* apresenta melhores resultados para $NI = 20\%$, enquanto que os modelos de FC (com exceção do LMF) apresentam resultados melhores para $NI = 40\%$ a 80%. Esse tipo

de comportamento pode indicar que os sistemas de filtragem colaborativa necessitam de um percentual de informação mais alto de cada respondente para recomendação individual, ou seja, recomendação de questões em nível de usuário. Essa interpretação reforça o uso de sistemas híbridos que utilizam diferentes abordagens para tratar o problema de previsão de acordo com as características dos dados.

Um comportamento atípico mostrado nas Tabelas 10 e 11 é a diminuição de performance com o aumento do NI em alguns casos. Na maioria dos resultados reportados para um mesmo cp e modelo (incluindo o *baseline*), os desempenhos são melhores para $NI = 60\%$ do que $NI = 80\%$ e para $NI = 40\%$ do que $NI = 60\%$. Uma possibilidade da causa desse comportamento atípico se baseia nos resultados mostrados na Tabela 12, que avalia os conjuntos de testes dos usuários para os dados de R_s.

Tabela 12 – Algumas métricas avaliadas nos dados de teste de R_s. IT se refere ao número de itens no conjunto de teste de cada usuário para a configuração de *cp* e NI. A penúltima coluna se refere ao número médio EI_t de eventos de interesse (notas 1,2 e 3) considerando todos os usuários e a última coluna o número médio EI_u de eventos de interesse no conjunto de teste de cada usuário.

cp	*NI* (%)	IT (%)	EI_t	EI_u
2	**20**	25	7,45	3,35
	40	19	5,85	3,25
	60	12	3,95	3,04
	80	6	2,27	2,65
3	**20**	33	7,23	4,57
	40	25	5,70	4,39
	60	16	3,98	4,02
	80	8	2,39	3,35
4	**20**	41	8,44	4,86
	40	31	6,66	4,65
	60	20	4,63	4,32
	80	10	2,78	3,60
5	**20**	34	7,66	4,44
	40	26	6,11	4,25
	60	17	4,32	3,93
	80	9	2,69	3,34
6	**20**	40	7,68	5,21
	40	30	6,02	4,98
	60	20	4,37	4,57
	80	10	2,67	3,74

Fonte: Autoria própria.

A Tabela 12 mostra que para níveis de informação mais elevados, diminui a proporção de eventos de interesse (notas baixas ou presença de comentário). Essa configuração de eventos dificulta a predição uma vez que aumenta a chance de eventos com notas altas ou ausência de comentários terem maior estimativa do que os eventos de interesse. A maior quantidade de

dados no treinamento pode não compensar essa dificuldade adicional, gerando uma queda no AUC. Entretanto, essa hipótese ainda precisa ser avaliada mais profundamente.

A avaliação em nível de usuário permite a exploração de métricas que se baseiam em uma lista de recomendação com tamanho k, como é o caso das métricas *precision@k* e *recall@k*. Nesse contexto, assume-se que cada usuário possui uma lista de recomendação própria com itens em ordem decrescente de estimativas. As Tabelas 13 e 14 possuem a mesma estrutura das tabela anteriores, porém mostram os resultados médios dos modelos considerando a métrica *precision@k* para $k = 1$. As Tabelas 15 e 16 mostram os resultados médios da métrica *recall@k* para $k = IT/4$, em que IT é o tamanho do conjunto de teste de cada usuário. Naturalmente, IT varia com a variação do NI e do cp considerados (vide as 3 primeiras colunas da Tabela 12).

A métrica *precision@k* para $k = 1$ mede a chance de recomendação de uma questão de baixa favorabilidade (ou que envolva a emissão de comentário) se o sistema pudesse recomendar apenas uma única questão. A Tabela 13 mostra que o modelo II, quando treinado com NI igual a 20%, sugere em média uma questão de baixa favorabilidade (notas 1, 2 ou 3) para 56,85% dos respondentes.

Assim como as Tabelas 10 e 11, os resultados das Tabelas 13 e 14 mostram que o modelo MF apresenta resultados melhores, principalmente para R_s. Observa-se também uma redução geral de desempenho dos modelos em relação às Tabelas 10 e 11. O modelo *baseline* torna-se competitivo, principalmente para R_c. Nota-se ainda que o modelo MF é favorecido com o aumento do nível de informação e o *baseline* é melhor para casos com menor nível de informação.

Já a métrica *recall@k* com $k = IT/4$ das Tabelas 15 e 16 avalia a proporção de eventos de interesse que são recuperados por uma lista de recomendação de questões com tamanho de um quarto do conjunto de teste original (e ordenada em ordem crescente de estimativas). A Tabela 16 mostra que, para o caso $cp = 6$ e $NI = 20\%$, o modelo *baseline* consegue recuperar em média 86,7% dos eventos que possuem comentários em uma lista de 10 questões.

Comportamentos semelhantes aos das Tabelas 13 e 14 são também observados para os modelos II, MF e *baseline*. Ou seja, MF é melhor para R_s e favorecido com o aumento do nível de informação, sendo o *baseline* melhor para casos com menor nível de informação. A exceção está nos resultados do modelo LMF para o problema envolvendo R_c, cujos valores estão pelo menos 5 pontos percentuais abaixo dos demais.

Os resultados de *recall@k* indicam uma boa perspectiva de aplicação da proposta na construção de questionários reduzidos. De maneira geral, os dados de desempenho mostram que é possível capturar pelo menos 70% dos eventos positivos com uma lista reduzida de questões com apenas 25% do tamanho original para qualquer configuração de NI, cp e matriz R_s ou R_c usando alguma das técnicas de estimação, com exceção do modelo LMF. Essa taxa de recuperação aumenta significativamente em alguns casos, chegando a mais de 85% para o $cp = 6$ nos dados de R_c.

Tabela 13 – Desempenho (por usuário) de média e desvio padrão (subscrito) da métrica *precision@k* para $k = 1$ obtidos na predição de eventos de favorabilidade (matriz R_s) pelos algoritmos II, LMF, MF, NCF e *baseline* para cada *checkpoint* cp e nível NI de informação com *hold-out* de cinco repetições. Células coloridas indicam o melhor desempenho médio para um dado cp e NI.

precision@k individual - matriz R_s						
cp	***NI* (%)**	II	MF	LMF	NCF	*Baseline*
2	**20**	$0{,}5685_{0{,}0049}$	$0{,}5646_{0{,}0021}$	$0{,}5491_{0{,}0211}$	$0{,}5578_{0{,}0172}$	$0{,}5668_{0{,}0026}$
	40	$0{,}586_{0{,}0049}$	$0{,}5937_{0{,}0049}$	$0{,}5675_{0{,}0071}$	$0{,}5704_{0{,}0035}$	$0{,}569_{0{,}0016}$
	60	$0{,}598_{0{,}0048}$	$0{,}6284_{0{,}0027}$	$0{,}5815_{0{,}0052}$	$0{,}5877_{0{,}0065}$	$0{,}5808_{0{,}0054}$
	80	$0{,}6256_{0{,}0046}$	$0{,}6679_{0{,}0071}$	$0{,}6101_{0{,}0081}$	$0{,}6126_{0{,}0059}$	$0{,}6098_{0{,}0082}$
3	**20**	$0{,}5335_{0{,}0076}$	$0{,}5277_{0{,}0115}$	$0{,}5214_{0{,}0032}$	$0{,}5258_{0{,}0042}$	$0{,}5246_{0{,}0054}$
	40	$0{,}5548_{0{,}0037}$	$0{,}5653_{0{,}0073}$	$0{,}5351_{0{,}0043}$	$0{,}5348_{0{,}0025}$	$0{,}5339_{0{,}0032}$
	60	$0{,}5685_{0{,}0116}$	$0{,}5879_{0{,}0125}$	$0{,}5429_{0{,}0073}$	$0{,}5514_{0{,}0094}$	$0{,}5448_{0{,}0089}$
	80	$0{,}6016_{0{,}0078}$	$0{,}6299_{0{,}0088}$	$0{,}5781_{0{,}0089}$	$0{,}5958_{0{,}0092}$	$0{,}5793_{0{,}0093}$
4	**20**	$0{,}5293_{0{,}0051}$	$0{,}5225_{0{,}0039}$	$0{,}5186_{0{,}0049}$	$0{,}5192_{0{,}0033}$	$0{,}5189_{0{,}0026}$
	40	$0{,}5373_{0{,}0047}$	$0{,}5653_{0{,}0042}$	$0{,}5092_{0{,}0038}$	$0{,}5109_{0{,}0049}$	$0{,}5102_{0{,}0053}$
	60	$0{,}5394_{0{,}0116}$	$0{,}5896_{0{,}003}$	$0{,}5065_{0{,}0095}$	$0{,}5175_{0{,}0123}$	$0{,}506_{0{,}0096}$
	80	$0{,}571_{0{,}0114}$	$0{,}6225_{0{,}0111}$	$0{,}5381_{0{,}0084}$	$0{,}5723_{0{,}01}$	$0{,}5361_{0{,}0121}$
5	**20**	$0{,}5263_{0{,}0084}$	$0{,}5198_{0{,}0069}$	$0{,}4885_{0{,}0393}$	$0{,}5068_{0{,}009}$	$0{,}5102_{0{,}0047}$
	40	$0{,}5337_{0{,}0032}$	$0{,}5703_{0{,}0046}$	$0{,}5078_{0{,}0017}$	$0{,}5113_{0{,}0062}$	$0{,}507_{0{,}0023}$
	60	$0{,}5413_{0{,}0035}$	$0{,}6062_{0{,}0044}$	$0{,}5099_{0{,}0055}$	$0{,}5247_{0{,}0037}$	$0{,}5111_{0{,}0046}$
	80	$0{,}5749_{0{,}0035}$	$0{,}6481_{0{,}0066}$	$0{,}5388_{0{,}0076}$	$0{,}5645_{0{,}006}$	$0{,}5357_{0{,}0032}$
6	**20**	$0{,}5854_{0{,}0036}$	$0{,}5753_{0{,}003}$	$0{,}573_{0{,}0015}$	$0{,}5716_{0{,}0048}$	$0{,}5743_{0{,}002}$
	40	$0{,}5813_{0{,}0072}$	$0{,}6086_{0{,}0095}$	$0{,}5581_{0{,}0119}$	$0{,}5676_{0{,}0069}$	$0{,}5605_{0{,}0084}$
	60	$0{,}5745_{0{,}0052}$	$0{,}6302_{0{,}0102}$	$0{,}5486_{0{,}007}$	$0{,}5622_{0{,}0094}$	$0{,}5491_{0{,}0073}$
	80	$0{,}5982_{0{,}0095}$	$0{,}6603_{0{,}0091}$	$0{,}5731_{0{,}0079}$	$0{,}5945_{0{,}0107}$	$0{,}5735_{0{,}0076}$

Fonte: Autoria própria.

5.4 Discussão dos resultados

Os bons resultados de AUC global para os dois problemas (R_s e R_c) apresentados na Seção 5.3.1 mostram que os eventos de interesse (baixa favorabilidade em R_s e emissão de comentários em R_c) possuem maiores estimativas que os eventos complementares. Isso indica que a sugestão de pares respondente-questão é efetiva quando realizada pelos modelos de filtragem colaborativa. Por exemplo, é possível construir um sistema de recomendação treinado com os dados de R_c que sugira questões para respondentes na tentativa de obter o maior número de comentários com o menor número de disparos de questões. Esse tipo de abordagem é mais similar à pesquisas de pulso (WELBOURNE, 2016), que são menores e mais frequentes, do que pesquisas de clima, que são extensas e aplicadas em larga escala.

Já os dados de performance por usuário, reportados na Seção 5.3.2 mostram que questões relevantes (com baixa favorabilidade e que geram comentários) possuem maiores estimativas do que questões não relevantes para cada respondente. Particularmente, é possível recuperar mais de 70% das questões relevantes para um usuário em um questionário com um

Tabela 14 – Desempenho (por usuário) de média e desvio padrão (subscrito) da métrica *precision@k* para $k = 1$ obtidos na predição de eventos de comentários (matriz R_c) pelos algoritmos II, LMF, MF, NCF e *baseline* para cada *checkpoint* cp e nível NI de informação com *hold-out* de cinco repetições. Células coloridas indicam o melhor desempenho médio para um dado cp e NI.

precision@k individual - matriz R_c						
cp	***NI* (%)**	II	MF	LMF	NCF	*Baseline*
2	**20**	$0{,}2868_{0{,}0163}$	$0{,}3444_{0{,}011}$	$0{,}1845_{0{,}0658}$	$0{,}3372_{0{,}0178}$	$0{,}3461_{0{,}0095}$
	40	$0{,}3594_{0{,}0095}$	$0{,}373_{0{,}0081}$	$0{,}3235_{0{,}0417}$	$0{,}358_{0{,}0089}$	$0{,}3541_{0{,}0052}$
	60	$0{,}4365_{0{,}023}$	$0{,}4284_{0{,}0178}$	$0{,}3633_{0{,}0197}$	$0{,}3815_{0{,}0163}$	$0{,}3832_{0{,}0194}$
	80	$0{,}5231_{0{,}0321}$	$0{,}5065_{0{,}0237}$	$0{,}4428_{0{,}0279}$	$0{,}4682_{0{,}0216}$	$0{,}4637_{0{,}0238}$
3	**20**	$0{,}2507_{0{,}0093}$	$0{,}296_{0{,}0353}$	$0{,}1647_{0{,}0528}$	$0{,}2946_{0{,}0344}$	$0{,}314_{0{,}0067}$
	40	$0{,}339_{0{,}0028}$	$0{,}3408_{0{,}0106}$	$0{,}2496_{0{,}0521}$	$0{,}3231_{0{,}0084}$	$0{,}3241_{0{,}0051}$
	60	$0{,}4006_{0{,}0166}$	$0{,}3996_{0{,}0099}$	$0{,}2756_{0{,}0565}$	$0{,}3626_{0{,}0113}$	$0{,}3616_{0{,}0113}$
	80	$0{,}47_{0{,}0064}$	$0{,}4792_{0{,}0162}$	$0{,}3329_{0{,}014}$	$0{,}4316_{0{,}0142}$	$0{,}4318_{0{,}0177}$
4	**20**	$0{,}2397_{0{,}0131}$	$0{,}3078_{0{,}0183}$	$0{,}2159_{0{,}0696}$	$0{,}3111_{0{,}0194}$	$0{,}3217_{0{,}0034}$
	40	$0{,}331_{0{,}0148}$	$0{,}3341_{0{,}0126}$	$0{,}253_{0{,}0739}$	$0{,}3191_{0{,}0118}$	$0{,}3244_{0{,}0123}$
	60	$0{,}3875_{0{,}0105}$	$0{,}394_{0{,}0114}$	$0{,}2478_{0{,}0283}$	$0{,}3543_{0{,}005}$	$0{,}3517_{0{,}0049}$
	80	$0{,}4644_{0{,}009}$	$0{,}4747_{0{,}0146}$	$0{,}3019_{0{,}0208}$	$0{,}4187_{0{,}0078}$	$0{,}4112_{0{,}0079}$
5	**20**	$0{,}302_{0{,}0135}$	$0{,}3761_{0{,}0034}$	$0{,}2562_{0{,}0987}$	$0{,}376_{0{,}0035}$	$0{,}3786_{0{,}0024}$
	40	$0{,}389_{0{,}0128}$	$0{,}3827_{0{,}0098}$	$0{,}3335_{0{,}0766}$	$0{,}3769_{0{,}0134}$	$0{,}3771_{0{,}0123}$
	60	$0{,}4322_{0{,}0109}$	$0{,}4402_{0{,}0093}$	$0{,}3196_{0{,}0582}$	$0{,}3988_{0{,}0059}$	$0{,}3993_{0{,}0029}$
	80	$0{,}4703_{0{,}0205}$	$0{,}4818_{0{,}0199}$	$0{,}3181_{0{,}0112}$	$0{,}4182_{0{,}0123}$	$0{,}4167_{0{,}0133}$
6	**20**	$0{,}4711_{0{,}0175}$	$0{,}5819_{0{,}0066}$	$0{,}4968_{0{,}0989}$	$0{,}582_{0{,}0063}$	$0{,}5821_{0{,}0065}$
	40	$0{,}5625_{0{,}0098}$	$0{,}5646_{0{,}0088}$	$0{,}5442_{0{,}0172}$	$0{,}5561_{0{,}0107}$	$0{,}5607_{0{,}0088}$
	60	$0{,}5699_{0{,}0108}$	$0{,}578_{0{,}0101}$	$0{,}4377_{0{,}0269}$	$0{,}554_{0{,}0089}$	$0{,}5547_{0{,}0057}$
	80	$0{,}5982_{0{,}0133}$	$0{,}6003_{0{,}0119}$	$0{,}4547_{0{,}0315}$	$0{,}5567_{0{,}0111}$	$0{,}5589_{0{,}0081}$

Fonte: Autoria própria.

quarto do tamanho original a partir da observação de no mínimo 40% dos eventos da pesquisa. Para pesquisas maiores, menores níveis de informação seriam necessários.

Entretanto, sob a análise individual (por usuário) verifica-se, para os dois problemas, que um sistema heurístico frequentista não-personalizado é competitivo para essa tarefa. Estima-se que a performance similar dos modelos de filtragem colaborativa com relação à abordagem *baseline*, ocorra neste cenário devido ao formato de treinamento dos modelos, que minimiza a perda global ao invés das perdas individuais por respondente. Neste caso, acredita-se que mudanças na etapa de treino possam reverter esse cenário (ao invés de mudanças nos modelos). Por exemplo, uma solução de compromisso pode vir de um agrupamento de usuários com características semelhantes de forma que o treinamento global possa ser utilizado, mas agora para um grupo menor de usuários.

Os resultados de desempenho obtidos na Seção 5.3 assumem que que cada respondente contribui igualmente para os dados de treino. Entretanto, em questionários reais, estes são respondidos assincronamente pelos respondentes, gerando matrizes respondente-questão assimétricas. Essa distribuição de dados diferenciada deverá impactar os resultados de desem-

Tabela 15 – Desempenho (por usuário) de média e desvio padrão (subscrito) da métrica *recall@k* para $k = IT/4$ em que IT é igual ao tamanho do conjunto de teste do usuário para um nível de informação NI, obtidos na predição de eventos de favorabilidade (matriz R_s) pelos algoritmos II, LMF, MF, NCF e *baseline* para cada *checkpoint* cp e nível NI de informação com *hold-out* de cinco repetições. Células coloridas indicam o melhor desempenho médio para um dado cp e NI.

		recall@k individual - matriz R_s				
cp	***NI* (%)**	II	MF	LMF	NCF	*Baseline*
2	**20**	$0{,}7024_{0{,}0051}$	$0{,}7127_{0{,}0067}$	$0{,}6917_{0{,}0103}$	$0{,}7126_{0{,}0108}$	$0{,}7147_{0{,}0052}$
	40	$0{,}713_{0{,}0024}$	$0{,}72_{0{,}002}$	$0{,}7045_{0{,}0023}$	$0{,}7065_{0{,}0018}$	$0{,}707_{0{,}002}$
	60	$0{,}7356_{0{,}0018}$	$0{,}752_{0{,}0008}$	$0{,}7218_{0{,}0026}$	$0{,}7254_{0{,}0036}$	$0{,}7224_{0{,}0025}$
	80	$0{,}7207_{0{,}0037}$	$0{,}7477_{0{,}0033}$	$0{,}7047_{0{,}0042}$	$0{,}7123_{0{,}0028}$	$0{,}7057_{0{,}0034}$
3	**20**	$0{,}7608_{0{,}0035}$	$0{,}7743_{0{,}0035}$	$0{,}7594_{0{,}0131}$	$0{,}7751_{0{,}0039}$	$0{,}7745_{0{,}001}$
	40	$0{,}7801_{0{,}0025}$	$0{,}7794_{0{,}0033}$	$0{,}7618_{0{,}0045}$	$0{,}7674_{0{,}0012}$	$0{,}767_{0{,}0012}$
	60	$0{,}7915_{0{,}0015}$	$0{,}7953_{0{,}0029}$	$0{,}773_{0{,}0039}$	$0{,}7779_{0{,}0019}$	$0{,}7756_{0{,}0027}$
	80	$0{,}774_{0{,}0044}$	$0{,}7787_{0{,}0024}$	$0{,}756_{0{,}0063}$	$0{,}7627_{0{,}0029}$	$0{,}7573_{0{,}0058}$
4	**20**	$0{,}7971_{0{,}0012}$	$0{,}7981_{0{,}0042}$	$0{,}7748_{0{,}0139}$	$0{,}798_{0{,}0045}$	$0{,}8001_{0{,}0024}$
	40	$0{,}806_{0{,}0018}$	$0{,}8118_{0{,}0014}$	$0{,}7829_{0{,}0052}$	$0{,}7906_{0{,}0012}$	$0{,}7894_{0{,}0012}$
	60	$0{,}8117_{0{,}0014}$	$0{,}8189_{0{,}0025}$	$0{,}7873_{0{,}0031}$	$0{,}7952_{0{,}0036}$	$0{,}7904_{0{,}0018}$
	80	$0{,}7912_{0{,}0063}$	$0{,}7993_{0{,}0052}$	$0{,}7642_{0{,}0046}$	$0{,}7777_{0{,}0085}$	$0{,}7672_{0{,}006}$
5	**20**	$0{,}7782_{0{,}0035}$	$0{,}7816_{0{,}0059}$	$0{,}758_{0{,}0096}$	$0{,}7725_{0{,}0067}$	$0{,}7743_{0{,}0052}$
	40	$0{,}7923_{0{,}0027}$	$0{,}8022_{0{,}0013}$	$0{,}7663_{0{,}0042}$	$0{,}7716_{0{,}0066}$	$0{,}7695_{0{,}0037}$
	60	$0{,}7645_{0{,}004}$	$0{,}787_{0{,}0056}$	$0{,}7337_{0{,}0043}$	$0{,}7489_{0{,}006}$	$0{,}7354_{0{,}004}$
	80	$0{,}7215_{0{,}0043}$	$0{,}7556_{0{,}0058}$	$0{,}6897_{0{,}0038}$	$0{,}7106_{0{,}0057}$	$0{,}6904_{0{,}0034}$
6	**20**	$0{,}8321_{0{,}0012}$	$0{,}8379_{0{,}0028}$	$0{,}8193_{0{,}0078}$	$0{,}8379_{0{,}0032}$	$0{,}8382_{0{,}0025}$
	40	$0{,}8433_{0{,}0025}$	$0{,}8457_{0{,}0016}$	$0{,}8261_{0{,}0039}$	$0{,}8349_{0{,}0024}$	$0{,}8356_{0{,}002}$
	60	$0{,}8349_{0{,}0037}$	$0{,}8426_{0{,}0029}$	$0{,}8198_{0{,}0044}$	$0{,}824_{0{,}0032}$	$0{,}8221_{0{,}004}$
	80	$0{,}8152_{0{,}0074}$	$0{,}827_{0{,}0027}$	$0{,}798_{0{,}0074}$	$0{,}8053_{0{,}0072}$	$0{,}7981_{0{,}0075}$

Fonte: Autoria própria.

penho. Uma outra configuração possível de ser testada é a estimação de eventos de um *cp* com base na observação de interações respondente-questão registrados no *cp* mais recente. A Figura 9 mostra que existe a possibilidade de aplicação de estimadores treinados com dados históricos sem a necessidade de envolver soluções específicas para o problema de *cold-start*, pois *cps* adjacentes são similares em relação ao público e aos seus temas. A análise de desempenho dos preditores aplicados nessas diferentes configurações pode auxiliar no entendimento dos desafios e oportunidades para a criação de sistemas de recomendação de questões com base em preditores.

Além disso, como explicado na Seção 2.1, sistemas de recomendação não precisam ser construídos apenas com as estimativas de preditores individuais. Por exemplo, pode-se construir uma lista de recomendação que combina as estimativas de um preditor treinado com os dados de R_c com um preditor treinado com os dados de R_s, sugerindo, portanto, questões que são relativas a aspectos falhos da empresa e que geram comentários a respeito desses aspectos.

Tabela 16 – Desempenho (por usuário) de média e desvio padrão (subscrito) da métrica *recall@k* para $k = IT/4$ em que IT é igual ao tamanho do conjunto de teste do usuário para um nível de informação NI, obtidos na predição de eventos de comentários (matriz R_c) pelos algoritmos II, LMF, MF, NCF e *baseline* para cada *checkpoint* cp e nível NI de informação com *hold-out* de cinco repetições. Células coloridas indicam o melhor desempenho médio para um dado cp e NI.

		recall@k individual - matriz R_c				
cp	***NI* (%)**	II	MF	LMF	NCF	*Baseline*
2	**20**	$0{,}6434_{0{,}0092}$	$0{,}6937_{0{,}02}$	$0{,}5748_{0{,}0385}$	$0{,}6785_{0{,}0173}$	$0{,}7088_{0{,}0113}$
	40	$0{,}6799_{0{,}0095}$	$0{,}6884_{0{,}007}$	$0{,}6175_{0{,}0256}$	$0{,}6821_{0{,}0077}$	$0{,}6862_{0{,}0088}$
	60	$0{,}7285_{0{,}0091}$	$0{,}7176_{0{,}0106}$	$0{,}6567_{0{,}032}$	$0{,}7069_{0{,}0119}$	$0{,}7055_{0{,}0097}$
	80	$0{,}7346_{0{,}0081}$	$0{,}7122_{0{,}0106}$	$0{,}6555_{0{,}0071}$	$0{,}6882_{0{,}0094}$	$0{,}6863_{0{,}0125}$
3	**20**	$0{,}68_{0{,}0062}$	$0{,}7404_{0{,}0063}$	$0{,}596_{0{,}0746}$	$0{,}7351_{0{,}0103}$	$0{,}744_{0{,}0036}$
	40	$0{,}7198_{0{,}0103}$	$0{,}7428_{0{,}0074}$	$0{,}651_{0{,}0395}$	$0{,}7394_{0{,}0078}$	$0{,}7391_{0{,}006}$
	60	$0{,}7678_{0{,}0047}$	$0{,}7538_{0{,}0062}$	$0{,}6479_{0{,}0545}$	$0{,}7416_{0{,}0041}$	$0{,}7453_{0{,}0053}$
	80	$0{,}758_{0{,}0162}$	$0{,}7331_{0{,}0182}$	$0{,}6342_{0{,}0112}$	$0{,}719_{0{,}0127}$	$0{,}7183_{0{,}0161}$
4	**20**	$0{,}6989_{0{,}0041}$	$0{,}7648_{0{,}0038}$	$0{,}5924_{0{,}0233}$	$0{,}7609_{0{,}0071}$	$0{,}7802_{0{,}0066}$
	40	$0{,}7574_{0{,}0081}$	$0{,}7751_{0{,}0059}$	$0{,}6902_{0{,}026}$	$0{,}7722_{0{,}0062}$	$0{,}7716_{0{,}0061}$
	60	$0{,}7985_{0{,}0052}$	$0{,}7865_{0{,}0053}$	$0{,}6218_{0{,}007}$	$0{,}776_{0{,}0086}$	$0{,}7775_{0{,}0044}$
	80	$0{,}7972_{0{,}0089}$	$0{,}776_{0{,}0189}$	$0{,}6387_{0{,}0191}$	$0{,}7569_{0{,}0061}$	$0{,}7572_{0{,}0067}$
5	**20**	$0{,}699_{0{,}0069}$	$0{,}7598_{0{,}0096}$	$0{,}6478_{0{,}0366}$	$0{,}7541_{0{,}0055}$	$0{,}7695_{0{,}0099}$
	40	$0{,}7546_{0{,}0089}$	$0{,}7762_{0{,}0043}$	$0{,}6888_{0{,}0373}$	$0{,}7653_{0{,}0055}$	$0{,}7693_{0{,}0063}$
	60	$0{,}7612_{0{,}0082}$	$0{,}7538_{0{,}0154}$	$0{,}616_{0{,}0558}$	$0{,}7346_{0{,}0058}$	$0{,}7355_{0{,}0074}$
	80	$0{,}7403_{0{,}0071}$	$0{,}7248_{0{,}0078}$	$0{,}5984_{0{,}0208}$	$0{,}6886_{0{,}0183}$	$0{,}6872_{0{,}0163}$
6	**20**	$0{,}793_{0{,}0048}$	$0{,}8636_{0{,}0084}$	$0{,}7164_{0{,}0403}$	$0{,}8628_{0{,}0074}$	$0{,}8661_{0{,}005}$
	40	$0{,}8373_{0{,}0058}$	$0{,}8609_{0{,}001}$	$0{,}7795_{0{,}0314}$	$0{,}8588_{0{,}0037}$	$0{,}8606_{0{,}0024}$
	60	$0{,}8561_{0{,}0052}$	$0{,}8573_{0{,}0048}$	$0{,}7106_{0{,}0324}$	$0{,}851_{0{,}0063}$	$0{,}8521_{0{,}0066}$
	80	$0{,}8528_{0{,}0064}$	$0{,}8375_{0{,}0047}$	$0{,}7266_{0{,}024}$	$0{,}8282_{0{,}0096}$	$0{,}8305_{0{,}0069}$

Fonte: Autoria própria.

6 CONCLUSÕES E TRABALHOS FUTUROS

Este trabalho abordou o problema de predição de favorabilidade (discriminação entre baixos e altos escores na escala Linkert) e de ocorrência ou não de comentários associados em pesquisas de clima organizacional. Para isso, foram usados algoritmos baseados em filtragem colaborativa como modelos preditores. Cinco algoritmos (item-item, fatoração de matriz, fatoração de matriz logística, filtragem colaborativa neural e um mecanismo de média simples) foram testados em uma ampla pesquisa com diversos pontos verificação (*checkpoints*). Dados de seis *checkpoints* foram organizados de modo a fornecer duas matrizes em cada *checkpoint*: uma contendo pares respondente-questão, cujo evento de interesse é a resposta de baixa favorabilidade (notas 1, 2 ou 3) e outra cujo evento de interesse é a ocorrência de comentários associados a essas respostas.

Nos experimentos, primeiramente foi realizada uma análise exploratória dos dados com levantamento de algumas características dos diferentes *checkpoints*. Posteriormente, aplicou-se um processo de configuração sistemática dos modelos, com os dados do primeiro *checkpoint*. Em seguida, os desempenhos dos algoritmos foram avaliados (usando a técnica de *holdout* com cinco repetições) nos *checkpoints* restantes considerando-se diferentes níveis de informação, variando de 20% a 80%. Dois casos distintos foram considerados: i) teste global usando a métrica AUC; ii) teste por respondente usando as métricas AUC, *precision@k* e *recall@k*.

Os resultados da análise exploratória mostraram que a maioria das respostas é de alta favorabilidade (scores 4 e 5 na escala Likert) e que maioria dos respondentes não adicionam comentários. A média de comentários por respondente que incluem ao menos um comentário gira em torno de cinco. Não há uma preferência clara em relação a qual questão do questionário terá comentário associado ou não, ou qual questão será avaliada com baixa favorabilidade por um usuário. Além disso, levantamentos realizados próximos uns dos outros possuem, na sua maioria, os mesmos respondentes e questões.

Em relação à etapa de configuração dos algoritmos de filtragem colaborativa realizada com os dados do primeiro *checkpoint*, verificou-se que os melhores resultados ocorreram com o uso de vetores latentes com 32 elementos. Isso sugere que é relativamente baixa a quantidade de aspectos que influenciam o colaborador ao avaliar uma questão ou emitir um comentário. Além disso, as configurações vencedoras desta etapa se mostraram eficientes mesmo quando aplicadas em *checkpoints* com características diversas na etapa de comparação de desempenho.

Os resultados de desempenho mostraram que, na avaliação em um conjunto de teste composto por todos os respondentes, os algoritmos de filtragem colaborativa apresentaram bons resultados para AUC, sendo capazes de explorar os padrões existentes nas interações respondente-questão melhor que uma abordagem *baseline* frequentista. Já na avaliação do desempenho por usuário, o algoritmo *baseline* passou a ser competitivo dada a queda de performance dos modelos de filtragem colaborativa. Esse resultado não é de todo inesperado uma

vez que estes modelos foram treinados de uma forma global, e neste cenário estão sendo testados de forma individual.

Como um dos resultados mais interessantes do cenário de análise individual, destaca-se que é possível recuperar em torno de 70% das questões que geram eventos de interesse (baixa favorabilidade e que emitem comentários) para um determinado usuário com apenas um quarto da lista de questões original. Para alguns modelos treinados com $cp = 6$, esse índice sobre para 85%.

Os resultados de desempenho em ambos os cenários (análise global e individual) assumem uma configuração de dados em que os respondentes contribuem igualmente nos conjuntos de treino e teste (para evitar o problema de inicialização a frio). No entanto, outras possibilidades podem ser exploradas, como configurações assimétricas (mais comuns em casos de questionários que são respondidos assincronamente) e configurações que combinam dados de *checkpoints* diferentes. A avaliação dos preditores em diferentes configurações de treino e teste podem auxiliar no entendimento de desafios e oportunidades de uso dos preditores para sistemas de recomendação de questões.

Um sistema de recomendação poderia ser construído de maneira a considerar: i) os aspectos mais importantes para cada respondente (aqueles em que o respondente estaria engajado o suficiente para adicionar um comentário a respeito); e ii) os aspectos que são vistos negativamente pelos respondentes e que são importantes de serem identificados pela empresa. Esse sistema poderia auxiliar a empresa a manter um conjunto reduzido de perguntas a serem aplicadas em questionários focados nos aspectos importantes para os colaboradores e que merecem mais atenção.

Vale destacar que esta dissertação não explora outros aspectos de sistemas de recomendação no contexto de pesquisas em clima organizacional por meio de recomendação de questões. O trabalho limita-se a dar suporte à recomendação de questões a partir da predição de favorabilidade das questões e da emissão de comentários.

Em trabalhos futuros, diferentes frentes de pesquisa podem ser consideradas, tais como:

- explorar o problema de inicialização a frio a partir do uso de sistemas híbridos que utilizam informações relevantes dos respondentes e das questões;
- testar os preditores item-item e fatoração matricial (que tiveram melhor desempenho) em outras configurações de dados de treino;
- testar versões mais avançadas do algoritmo de fatoração matricial, como o modelo SVD++ proposto por Koren (2008);
- avaliar a possibilidade do uso de informação contextual na recomendação, tais como os tópicos extraídos dos comentários;
- estudar a relação entre os respondentes e questões no espaço latente compartilhado gerado pelos modelos fatoriais;

- comparar o desempenho dos algoritmos com métricas que levam em consideração a diversidade das questões;
- usar a métrica LAUC (*Limited Area Under the Curve*) de Schröder, Thiele e Lehner (2011) ao invés de AUC;
- combinar preditores treinados para diferentes problemas de estimação, construindo um sistema de recomendação multiobjetivo.

REFERÊNCIAS

AL-SHAMRI, M. Y. H. User profiling approaches for demographic recommender systems. **Knowledge-Based Systems**, Elsevier, v. 100, p. 175–187, 2016.

AMATRIAIN, X.; BASILICO, J. Past, present, and future of recommender systems: An industry perspective. *In*: **Proceedings of the 10th ACM Conference on Recommender Systems**. [*S.l.*: *s.n.*], 2016. p. 211–214.

BELL, R. M.; KOREN, Y. Lessons from the netflix prize challenge. **ACM SIGKDD Explorations Newsletter**, ACM New York, NY, USA, v. 9, n. 2, p. 75–79, 2007.

BOIM, R. *et al.* Asking the right questions in crowd data sourcing. *In*: IEEE. **2012 IEEE 28th International Conference on Data Engineering**. [*S.l.*], 2012. p. 1261–1264.

BRISCOE, E.; FELDMAN, J. Conceptual complexity and the bias/variance tradeoff. **Cognition**, Elsevier, v. 118, n. 1, p. 2–16, 2011.

CHUN, A. Y.; HEERINGA, S.; SCHOUTEN, J. Responsive and adaptive design for survey optimization. **Journal of Official Statistics**, v. 34, n. 3, p. 581–597, 2018.

EARLY, K.; MANKOFF, J.; FIENBERG, S. E. Dynamic question ordering in online surveys. **Journal of Official Statistics**, v. 33, 2017. ISSN 20017367.

FALK, K. **Practical Recommender Systems**. [*S.l.*]: Manning Publications, 2019.

GONZALEZ, J. M.; ELTINGE, J. L. Adaptive matrix sampling for the consumer expenditure quarterly interview survey. *In*: **Proceedings of the Section on Survey Research Methods, American Statistical Association**. [*S.l.*: *s.n.*], 2008. p. 2081–8.

HE, X. *et al.* Neural collaborative filtering. *In*: **Proceedings of the 26th International World Wide Web Conference**. [*S.l.*]: International World Wide Web Conf. Steering Committee, 2017. p. 173–182. ISBN 9781450349130.

HORA, H. R. M. D.; MONTEIRO, G. T. R.; ARICA, J. Confiabilidade em questionários para qualidade: um estudo com o coeficiente alfa de cronbach. **Produto & Produção**, v. 11, n. 2, 2010.

HUANG, J.; LING, C. X. Using auc and accuracy in evaluating learning algorithms. **IEEE Transactions on Knowledge and Data Engineering**, IEEE, v. 17, n. 3, p. 299–310, 2005.

JOHNSON, C. C. Logistic matrix factorization for implicit feedback data. **Advances in Neural Information Processing Systems**, v. 27, n. 78, p. 1–9, 2014.

KOREN, Y. Factorization meets the neighborhood: a multifaceted collaborative filtering model. *In*: **Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. [*S.l.*: *s.n.*], 2008. p. 426–434.

KOREN, Y. Collaborative filtering with temporal dynamics. *In*: **Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. [*S.l.*: *s.n.*], 2009. p. 447–456.

KROSNICK, J. A. Questionnaire design. *In*: **The Palgrave Handbook of Survey Research**. [*S.l.*]: Springer, 2018. p. 439–455.

KULKARNI, P. V.; RAI, S.; KALE, R. Recommender system in elearning: a survey. *In*: SPRINGER. **Proceedings of International Conference on Computational Science and Applications**. [*S.l.*], 2020. p. 119–126.

LAVRAKAS, P. J. **Encyclopedia of Survey Research Methods**. [*S.l.*]: Sage publications, 2008.

LI, S.; KARATZOGLOU, A.; GENTILE, C. Collaborative filtering bandits. *In*: **Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval**. [*S.l.*: *s.n.*], 2016. p. 539–548.

LIKA, B.; KOLOMVATSOS, K.; HADJIEFTHYMIADES, S. Facing the cold start problem in recommender systems. **Expert Systems with Applications**, Elsevier, v. 41, n. 4, p. 2065–2073, 2014.

OLIVEIRA, M. F. B. G.; DELGADO, M.; LÜDERS, R. Comparative analysis of collaborative filtering-based predictors of scores in surveys of a large company. *In*: SBC. **Anais do XVIII Encontro Nacional de Inteligência Artificial e Computacional (ENIAC)**. [*S.l.*], 2021. p. 739–750.

ORTIGOSA, A.; PAREDES, P.; RODRIGUEZ, P. Ah-questionnaire: An adaptive hierarchical questionnaire for learning styles. **Computers & Education**, Elsevier, v. 54, n. 4, p. 999–1005, 2010.

PAZZANI, M. J.; BILLSUS, D. Content-based recommendation systems. *In*: **The Adaptive Web**. [*S.l.*]: Springer, 2007. p. 325–341.

RENDLE, S. *et al.* Neural collaborative filtering vs. matrix factorization revisited. 5 2020. Disponível em: http://arxiv.org/abs/2005.09683.

RENDLE, S. *et al.* Neural collaborative filtering vs. matrix factorization revisited. *In*: **Fourteenth ACM Conference on Recommender Systems**. [*S.l.*: *s.n.*], 2020. p. 240–248.

SCHOUTEN, B.; CALINESCU, M.; LUITEN, A. Optimizing quality of response through adaptive survey designs. **Survey Methodology**, v. 39, n. 1, p. 29–58, 2013.

SCHRÖDER, G.; THIELE, M.; LEHNER, W. Setting goals and choosing metrics for recommender system evaluations. *In*: **UCERSTI2 workshop at the 5th ACM Conference on Recommender Systems, Chicago, USA**. [*S.l.*: *s.n.*], 2011. v. 23, p. 53.

SONG, L.; TEKIN, C.; SCHAAR, M. V. D. Online learning in large-scale contextual recommender systems. **IEEE Transactions on Services Computing**, IEEE, v. 9, n. 3, p. 433–445, 2014.

SU, X.; KHOSHGOFTAAR, T. M. A survey of collaborative filtering techniques. **Advances in Artificial Intelligence**, Hindawi, v. 2009, 2009.

VARGAS, S.; CASTELLS, P. Rank and relevance in novelty and diversity metrics for recommender systems. *In*: **Proceedings of the Fifth ACM Conference on Recommender Systems**. [*S.l.*: *s.n.*], 2011. p. 109–116.

WAGNER, J. R. **Adaptive Survey Design to Reduce Nonresponse Bias**. 2008. Tese (Doutorado) — University of Michigan, 2008.

WANG, H.; WU, Q.; WANG, H. Factorization bandits for interactive recommendation. *In*: **Thirty-First AAAI Conference on Artificial Intelligence**. [*S.l.*: *s.n.*], 2017.

WELBOURNE, T. M. The potential of pulse surveys: Transforming surveys into leadership tools. **Employment Relations Today**, Wiley Periodicals, Inc. Hoboken, USA, v. 43, n. 1, p. 33–39, 2016.

WU, Q. *et al.* Contextual bandits in a collaborative environment. *In*: **Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval**. [*S.l.*: *s.n.*], 2016. p. 529–538.

ZHANG, C. *et al.* Active matrix factorization for surveys. **Annals of Applied Statistics**, v. 14, 2020. ISSN 19417330.