UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS


# Recommender system to support comprehensive exploration of large scale scientific datasets

*“ Documento Definitivo”*

**Doutoramento em Informática**

Márcia Cristina Afonso Barros

Tese orientada por:
Prof. Doutor Francisco José Moreira Couto
Prof. Doutor André Moitinho de Almeida

Documento especialmente elaborado para a obtenção do grau de doutor

2021

UNIVERSIDADE DE LISBOA

FACULDADE DE CIÊNCIAS


# Recommender system to support comprehensive exploration of large scale scientific datasets

**Doutoramento em Informática**

Márcia Cristina Afonso Barros

Tese orientada por:

Prof. Doutor Francisco José Moreira Couto

Prof. Doutor André Moitinho de Almeida

Júri:

Presidente:

- Doutor Nuno Fuentecilla Maia Ferreira Neves, Professor Catedrático e Presidente do Departamento de Informática da Faculdade de Ciências da Universidade de Lisboa

Vogais:

- Doutor José Luis Guimarães Oliveira, Professor Catedrático Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro;
- Doutor Joaquim Francisco Ferreira Silva, Professor Auxiliar Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa;
- Doutor Miguel Ângelo Leal da Costa, na qualidade de individualidade de reconhecida competência na área científica;
- Doutor Francisco José Moreira Couto, Professor Associado com Agregação Faculdade de Ciências da Universidade de Lisboa (orientador).

Documento especialmente elaborado para a obtenção do grau de doutor

This work was supported by FCT through funding of Deep Semantic Tagger (DeST) project (ref. PTDC/CCI-BIO/28685/2017), LASIGE Research Unit (ref. UIDB/00408/2020 and ref. UIDP/00408/2020), CENTRA Research Unit (ref. UIDB/00099/2020), and PhD Scholarship ref. SFRH/BD/128840/2017.

2021

This work was supported by FCT through funding of Deep Semantic Tagger (DeST) project (ref. PTDC/CCI-BIO/28685/2017), LASIGE Research Unit (ref. UIDB/00408/2020 and ref. UIDP/00408/2020), CENTRA Research Unit (ref. UIDB/00099/2020), and PhD Scholarship ref. SFRH/BD/128840/2017.


2021

# Abstract

Databases for scientific entities, such as chemical compounds, diseases and astronomical objects, are growing in size and complexity, reaching billions of items per database. Researchers need new and innovative tools for assisting the choice of these items. This work proposes the use of Recommender Systems approaches for helping researchers to find items of interest. We identified as one of the major challenges for applying RS in scientific fields the lack of standard and open-access datasets with information about the preferences of the users. To overcome this challenge, we developed a methodology called LIBRETTI - LIterature Based RecommEndaTion of scienTific Items, whose goal is to create <user,item,rating>datasets related to scientific fields. These datasets are created based on scientific literature, the major resource of knowledge that Science has. LIBRETTI methodology allowed the development and testing of new recommender algorithms specific for each field. Besides LIBRETTI, the main contributions of this thesis are standard and sequence-aware recommendation datasets in the fields of Astronomy, Chemistry, and Health (related to COVID-19 disease), a hybrid semantic recommender system for chemical compounds in large-scale datasets, a hybrid approach based on sequential enrichment (SeEn) for sequence-aware recommendations, a multi-field semantic-based pipeline for recommending biomedical entities related to COVID-19 disease.

**Keywords:** Recommender systems; Large-scale datasets; Scientific Data; External Sources; Ontology

# Resumo

Bases de dados de entidades científicas, como compostos químicos, doenças e objetos astronómicos, têm crescido em tamanho e complexidade, chegando a milhares de milhões de itens por base de dados. Os investigadores precisam de ferramentas novas e inovadoras para auxiliar na escolha desses itens. Este trabalho propõe o uso de Sistemas de Recomendação para auxiliar os investigadores a encontrar itens de interesse. Identificamos como um dos maiores desafios para a aplicação de sistemas de recomendação em áreas científicas a falta de conjuntos de dados padronizados e de acesso aberto com informações sobre as preferências dos utilizadores. Para superar esse desafio, desenvolvemos uma metodologia denominada LIBRETTI - Recomendação Baseada em Literatura de Itens Científicos, cujo objetivo é a criação de conjuntos de dados <utilizador, item, classificação>, relacionados com campos científicos. Estes conjuntos de dados são criados com base no principal recurso de conhecimento que a Ciência possui: a literatura científica. A metodologia LIBRETTI permitiu o desenvolvimento de novos algoritmos de recomendação específicos para vários campos científicos. Além do LIBRETTI, as principais contribuições desta tese são conjuntos de dados de recomendação padronizados nas áreas de Astronomia, Química e Saúde (relacionado com a doença COVID-19), um sistema de recomendação semântica híbrido para compostos químicos em conjuntos de dados de grande escala, uma abordagem híbrida baseada no enriquecimento sequencial (SeEn) para recomendações sequenciais, um pipeline baseado em semântica de vários campos para recomendar entidades biomédicas relacionadas com a doença COVID-19.

**Palavras Chave:** Sistemas de recomendação; Conjunto de Dados de Larga Escala; Dados Científicos; Fontes Externas; Ontologias

# Resumo Alargado

Bases de dados de entidades científicas, como compostos químicos, doenças e objetos astronómicos, têm crescido em tamanho e complexidade, chegando aos milhares de milhões de itens por base de dados. Os investigadores precisam de ferramentas novas e inovadoras para auxiliar na escolha desses itens. Este trabalho propõe o uso de Sistemas de Recomendação (RS) para auxiliar os investigadores na escolha de novos itens de interesse. Os RS têm sido explorados com sucesso num grande número de domínios, por exemplo, filmes e programas de TV, música ou comércio eletrónico. Nestes domínios, temos um grande número de conjuntos de dados disponíveis para testar e avaliar novos algoritmos de recomendação. Por exemplo, temos os conjuntos de dados do Movielens e da Netflix para filmes, o Spotify para música e a Amazon para e-commerce, o que se traduz num grande número de algoritmos de sucesso aplicados a esses campos. Estes dados são um historial da preferência dos utilizadores sobre os itens. Podemos ter uma informação explícita, quando, por exemplo, os utilizadores classificam um item numa escala de zero a dez, ou implícita, quando a informação sobre as preferências são recolhidas através da interação entre utilizadores e os itens, por exemplo, um filme visto ou um produto comprado. No entanto, os RS não são usados com tanta frequência em áreas científicas, como Saúde, Química e Astronomia. Foi identificado como um dos maiores desafios para a aplicação do RS em áreas científicas a falta de conjuntos de dados padronizados e de acesso aberto com as informações sobre as preferências dos utilizadores. Para superar este desafio, foi desenvolvida neste trabalho uma metodologia chamada LIBRETTI - Recomendação Baseada em Literatura de Itens Científicos, cujo objetivo é a criação de conjuntos de dados <utilizador, item, classificação>, relacionados com áreas científicas. Estes conjuntos de dados são criados com base no principal recurso de conhecimento que a Ciência possui: a literatura científica, Os utilizadores nestes novos datasets são os autores das publicações, os itens são as entidades científicas (por exemplo, compostos químicos ou doenças) e as classificações são o número de

publicações que um autor escreveu sobre uma entidade. Por exemplo, se o autor John Smith escrever três artigos que mencionam o composto químico Paracetamol, no conjunto de dados aparecerá o seguinte: <John Smith,Paracetamol,3>. O LIBRETTI foi avaliado em dois casos de estudo distintos, Astronomia e Química. No caso de estudo em Astronomia, os itens são aglomerados abertos de estrelas, e são utilizadas duas fontes de conhecimento para extrair os artigos ligados aos aglomerados abertos de estrelas, o Simbad e o NASA/astrophysics data system (ADS). No caso de estudo de Química, os itens são compostos químicos, e o estudo utilizou a ontologia Chemical Entities of Biological Interest (ChEBI) como fonte de itens e para localizar os artigos ligados aos compostos químicos. Os resultados foram dois conjuntos de dados de recomendação, o aRM (matriz de recomendação astronómica) e o chERM (matriz de recomendação química), para os casos de estudo em Astronomia e Química, respectivamente. Esses conjuntos de dados foram comparados com um dos conjuntos de dados de recomendação mais usados, o MovieLens-100k, e com o SD4AI, um conjunto de dados também criado a partir da literatura científica, mas para recomendar artigos e tópicos de pesquisa. De acordo com os resultados, pode concluir-se que a literatura científica pode ser utilizada como fonte para a criação de conjuntos de dados de recomendação confiáveis em áreas científicas.

Com estes conjuntos de dados disponíveis, foi possível começar a testar e desenvolver novos algoritmos de recomendação. No campo da Química, foi desenvolvido nesta tese um modelo de recomendação híbrido adequado para conjuntos de dados de opinião implícita, focado em retornar uma lista de classificação de acordo com a relevância dos itens. O modelo integra algoritmos de filtragem colaborativa para feedback implícito (Alternating Least Squares (ALS) e Bayesian Personalized Ranking (BPR)) e um novo algoritmo baseado no conteúdo dos itens (ONTO), usando a similaridade semântica entre os compostos químicos na ontologia ChEBI. Os algoritmos foram avaliados num conjunto de dados implícito de compostos químicos, chERM-20, com mais de 16.000 itens. ALS, BPR e ONTO foram avaliados individualmente e como híbridos. Os resultados para os algoritmos híbridos são melhores quando comparados com os algoritmos individuais.

No entanto, a ciência é mutável ao longo do tempo, e os itens relevantes no passado podem não ser agora relevantes para o utilizador. Assim, desenvolveu-se uma abordagem híbrida entre os métodos de aprendizagem profunda de filtragem colaborativa e os métodos baseados no conteúdo dos itens. A abordagem é chamada de enriquecimento sequencial (SeEn) e consiste em adicionar a uma sequência de itens os n itens mais semelhantes após cada item original. A nova sequência é então passada como entrada para algoritmos de recomendação com reconhecimento de sequência de última geração (BERT4Rec) com o objetivo de melhorar os resultados quando comparados com a sequência original. A abordagem SeEn foi testada em dois conjuntos de dados nas áreas de Química, onde os itens são compostos químicos, e Astronomia, onde os itens são aglomerados abertos de estrelas. Para os compostos químicos, foi utilizada a semelhança semântica para calcular a similaridade entre os itens, utilizando a ontologia chEBI. Para os aglomerados abertos de estrelas, a similaridade foi calculada mapeando os aglomerados abertos de estrelas para o conjunto de dados do Gaia. Gaia é uma missão astronómica da Agência ESpacial Europeia (ESA) cujo objetivo é recolher informações sobre as estrelas da Via Láctea. O conjunto de dados está na terceira versão e tem mais de 1,9 biliões de estrelas. Assim, neste estudo foram utilizados os parâmetros das estrelas do satélite Gaia, por exemplo a distância, para calcular a semelhança entre os aglomerados de estrelas. Confirmando a hipótese, os modelos treinados com os conjuntos de dados enriquecidos alcançaram melhores resultados na avaliação do que os modelos treinados com o conjunto de dados original. O conjunto de dados de Química obteve uma melhoria de 7 pontos percentuais e o conjunto de dados de Astronomia de 16 pontos percentuais.

A pandemia COVID-19 aumentou ainda mais a importância de sistemas automáticos e ferramentas para extrair informações da literatura científica e para fornecer informações personalizadas num formato simples para os investigadores. Até ao momento, a base de dados de artigos de investigação biomédica Pubmed conta com mais de 150.000 artigos sobre a COVID-19, a grande maioria publicada nos anos de 2020 e 2021. Esta enorme quantidade de dados é uma fonte de conhecimento que precisa ser explorada para informações pertinentes.

As ontologias parecem ser uma chave na recomendação de entidades científicas. Assim, esta tese apresenta um pipeline baseado em semântica para recomendação de entidades biomédicas, especialmente desenvolvido para a doença COVID-19.

O pipeline consiste em realizar o reconhecimento de Entidades Nomeadas (NER) num corpus de documentos relacionados com a COVID-19, usando ontologias multidisciplinares para reconhecer e ligar as entidades. As entidades dessas ontologias são compostos químicos (chEBI), doenças (Disease Ontology - DO), fenótipos (Human Phenotype Ontology - HPO), e termos de genes (Gene Ontology - GO). A avaliação foi realizada usando o conjunto de dados COVID-19 Open Research Dataset (CORD-19). O objetivo era testar se o uso de múltiplas ontologias na criação do conjunto de dados de recomendação em áreas científicas melhora o desempenho de algoritmos de filtragem colaborativa de última geração, o que se comprovou após diversos testes. Este método permite a recomendação de entidades de várias áreas científicas relacionadas com a COVID-19.

As principais contribuições desta tese são:

- Métodos e algoritmos:
  - Uma nova metodologia (LIBRETTI) para criar conjuntos de dados de feedback implícito através da literatura científica;
  - Um novo algoritmo de recomendação chamado ONTO, baseado em ontologias e na semelhança semântica das entidades;
  - Um novo algoritmo de recomendação híbrido para conjuntos de dados de feedback implícito;
  - Uma nova abordagem híbrida (SeEn) para recomendações sequenciais;
  - Um pipeline baseado em semântica de vários campos para recomendar entidades biomédicas.
- Conjuntos de dados:
  - em Astronomia para a recomendação de aglomerados abertos de estrelas;
  - em Química, para a recomendação de compostos químicos;

  - em Saúde, para a recomendação de entidades biomédicas relacionadas com a doença COVID-19, sendo estas entidades doenças, termos de genes, compostos químicos, e fenótipos.

- Outras bases de dados:
  - Uma base de dados com a semelhança semântica entre mais de 16 mil compostos químicos para três medidas de semelhança;
  - Uma base de dados com a semelhança entre aglomerados de estrelas.
- Ferramentas:
  - Uma ferramenta de recomendação para recomendar compostos químicos;
  - O LightDiShIn, um método mais rápido de calcular a semelhança semântica entre as entidades de uma ontologias, implementado na biblioteca DiShIn;

**Palavras Chave:** Sistemas de recomendação; Conjunto de Dados de Larga Escala; Dados Científicos; Fontes Externas; Ontologias

# Acknowledgements

I would like to thank first of all to my advisers, Professor Francisco Couto and Professor André Moitinho, for all the support provided along the way. To my LASIGE colleagues, Diana, Pedro, André, Fernando, Vinícius, Adriano, with whom the gatherings are not only fun but learning. To my CENTRA colleagues, André, Alberto, Helder, and Kora for the support. To Alexandra and Carla, who have always a kind word in the bureaucratic adversities. To my friends, Sofia and Soraia, for the laughs and the cries, for always being there no matter what. To Carlos, my dearest teammate and friend, for the memes, the rubber duck support, and the cheering. To Ana, for listening and caring. To Miguel and Paula, for all the support. To my family, in special to my mother Fátima, and my godparents Alice and Adelino, for spoiling me since always, they are the true reason I could even think of a PhD. To my sister-in-law Beatriz, who was always here. To the cousins Cesário and Alexandra, who always had a word of advise in the decisions. To Safira, the best companion I could have. And specially to my boyfriend, Alexandre, to whom I dedicate this thesis, for supporting me through the ups and downs of this road.

De acordo com o disposto no artigo 24 o do Regulamento de Estudos de Pós-Graduação da Universidade de Lisboa, Despacho n o 7024/2017, publicado no Diário da República –2 a Série –n o 155 –11 de Agosto de 2017, foram utilizados nesta dissertação resultados incluıdos nos seguintes artigos:

- Barros, Marcia; Moitinho, André; Couto, Francisco M; Using Research Literature to Generate Datasets of Implicit Feedback for Recommending Scientific Items, IEEE Access, 7, 176668-176680, 2019, IEEE. Q1 journal. (`https://doi.org/10.1109/ACCESS.2019.2958002`) Capítulo 3.
- Barros, Marcia, Andre Moitinho, and Francisco M. Couto. "Hybrid semantic recommender system for chemical compounds in large-scale datasets." Journal of cheminformatics 13.1 (2021): 1-18. Q1 journal. (`https://doi.org/10.1186/s13321-021-00495-2`) Capítulo 4.
- Barros, Marcia; Moitinho, André; Couto, Francisco M; "Hybrid semantic recommender system for chemical compounds." European Conference on Information Retrieval. Springer, Cham, 2020. Core A conference (`https://doi.org/10.1007/978-3-030-45442-5_12`) Capítulo 4.
- Barros, M.; Lamurias, A.; Sousa, D., Ruas; P., Couto, F. M; (2020, December). COVID-19: A Semantic-Based Pipeline for Recommending Biomedical Entities. In Proceedings of the 1st Workshop on NLP for COVID-19 (Part 2) at EMNLP 2020. Core A conference (`http://dx.doi.org/10.18653/v1/2020.nlpcovid19-2.20`) Capítulo 6.

# Contents

# List of Figures

# List of Tables

# 1
# Introduction

Recommender Systems (RS) are software tools that provide suggestions for items that are most likely of interest to a particular user [165]. The recommendation of items is not new, existing since antiquity, from person to person, and now in modern days, with the evolution of technology and the Web, in several Websites. RS have been implemented in a wide range of fields, such as movies, books, research papers, or e-commerce [34, 124, 178]. Some well-known platforms integrating RS are GroupLens[1], including MovieLens[2], Amazon[3], Netflix[4], and Google News[5]. Due to the large variety of fields using RS, there has been a progressive interest in the research of new recommendation methods and algorithms [15, 34, 40, 44, 124, 151, 184, 194].

Figure 1.1 represents a global view of a RS. The main input of a RS is the feedback from users, which may be explicit or implicit, depending on how the user provides the information describing her/his preferences. Explicit feedback expects a rating provided on purpose by the user to an item. Contrariwise, implicit feedback results from the interaction of a user with a system, for instance, a website. Many platforms record users' behaviours, using this information to infer the level of interest of a user on an item. Explicit or implicit feedback allows the creation of user/item rating matrices, the primary input of a recommender system, where rows represent the users and the columns represent the items.

[1] http://grouplens.org
[2] http://grouplens.org/datasets/movielens/
[3] http://www.amazon.com
[4] http://www.netflix.com
[5] http://news.google.com


Figure 1.1: Recommender System general view.
The information about the preferences of the users is the main input of a recommender system. The output is the recommendation of items. The recommender system is a black box, whose primary goal is to combine the input information to provide the best recommendations, for example, through collaborative-filtering or content-based algorithms.

Depending on the approach, RS may be divided into Collaborative-filtering (CF), when using the similarity between the ratings of the users to provide the recommendations, Content-based (CB), when using the similarity between the features of the items, and hybrid, a combination of both CF and CB. Figure 1.2 shows the main difference between these two approaches. In CF RS, user 1 and user 2 read the same article. Thus they are similar users in terms of preferences. User 2 read a third article, which will be recommended to user 1. In Content-based approaches, user 1 read one article, a second article is similar to the one user 1 read, which will be recommended to user 1. CB approaches require having access to a list of features. In some fields, such as movies or books, these features are easy to define. For example, for movies, the features may be the director, genre, and actors. In other fields, the selection of the features is not so obvious.

Recommender algorithms have their inherent challenges [125]. CF is not efficient for new users and new items, which is called the cold start problem. CB deals well with the recommendations of new items since this approach is based on the item's features. However, it does not deal well with new users since it depends on the information about the interests of the users to recommend similar items. Additional challenges are related to the sparsity of the data, i.e., a large number of items and users, and few ratings, the scalability of the algorithms, and the quality of the recommendations.

Databases for scientific entities, such as chemical compounds, diseases and astronomical objects, are growing in size and complexity, reaching the billions of items per database. It is hard not to become "data stunned" by the large quantity, dimensionality, connectivity and all aggravated by noise. The researchers need new and innovative tools for assisting the choice of these items. RS could be a good solution for providing personalized suggestions to the

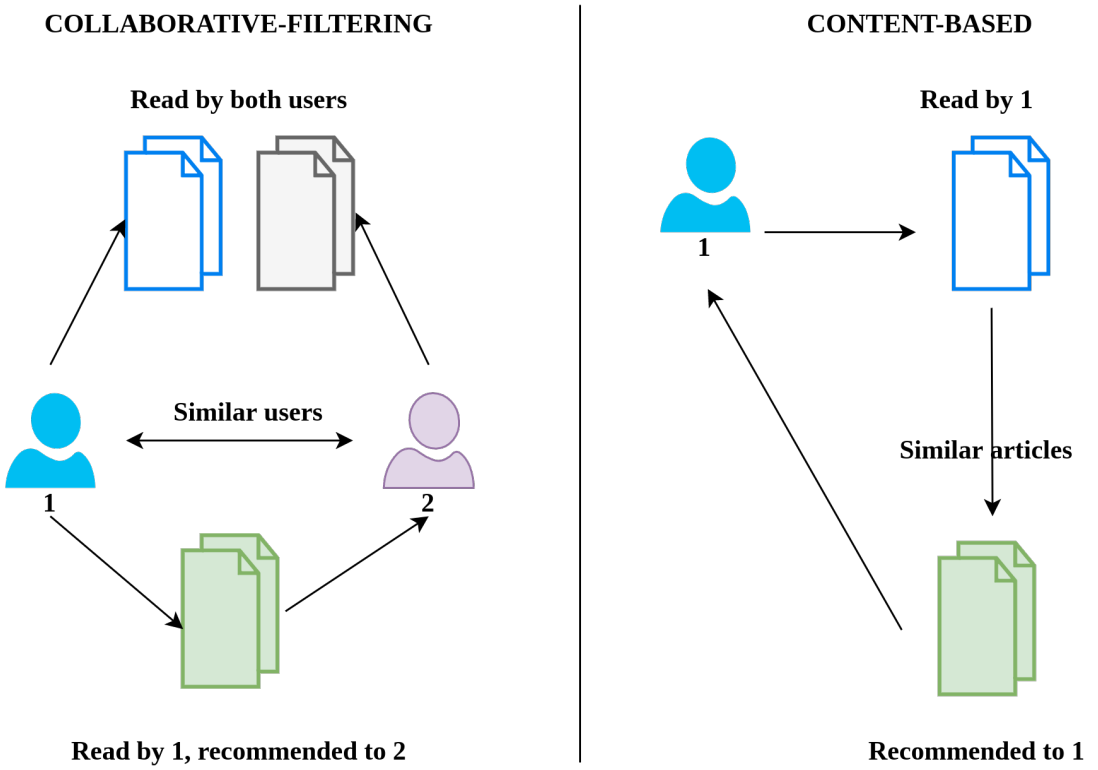


Figure 1.2: Collaborative-filtering vs content-based.

researchers about what scientific entities fit within their interests. Despite all the research and the extensive range of fields already using RS, in scientific fields RS are not widely used [242]. One of the main reasons is the lack of information about the preferences of the users inside a scientific field. Most of the information collected about users' feedback is maintained private and protected. The lack of information about the users is a problem since RS needs the users' feedback as the input of the system. Other issues that the existent recommender algorithms cannot address are related to the requirements of scientific datasets, as the specificity of the features of the items, and the restricted number of users for each scientific domain, leading to highly sparse datasets [162]. For example, in Astronomy or Chemistry, the users of a system are limited to the people interested in these fields. The few users and the lack of log files about users' preferences also pose a problem in evaluating RS in Scientific areas since there is a lack of reliable datasets to test the recommendation algorithms. There is a growing need for open-access datasets in scientific fields.

In this work, we propose a solution for the lack of open-access recommendation datasets

in scientific fields. We want to implement a solution suitable for several fields, which will allow the researchers to develop, test, and evaluate new solutions for different scientific fields without the restrictions of private datasets. The solution will be assessed in Chemistry and Astronomy fields by creating a recommendation dataset for recommending chemical compounds and open clusters of stars, respectively.

Besides the lack of recommendation datasets in scientific fields, we also identified another challenge related to the specificity and peculiarity of each field. For example, if we want to develop a content-based RS for chemical compounds, we first need to identify the best way to calculate the similarity between the chemical compounds. Ontologies, which are dictionaries hierarchically organized of entities from a specific area, may be used to this end. In this work, we intend to study how ontologies may improve the results of state-of-the-art recommendation algorithms for chemical compounds.

But science is mutable over time, and relevant items in the past may no longer be relevant for a user. One must consider that the traditional matrix RS methods may not be the most suitable to recommend the next best item. In this case, we want to develop sequence-aware RS suitable for scientific items to improve state-of-the-art recommendations.

The emergence of the COVID-19 pandemic further increased the importance of automatic systems and tools for extracting information from scientific literature and for providing personalized information in a clean and straightforward format to the researchers. To this date, the database for biomedical research articles Pubmed[1], accounts with more than 150.000 articles about COVID-19, the large majority published in the years 2020 and 2021. This massive raw amount of data is a source of knowledge that needs to be mined for pertinent information.

## 1.1 Objectives

The main goal of this thesis is to study how the use of RS may help researchers to find new scientific entities of interest. There are many challenges associated with the use of RS in scientific fields. The main challenges identified are

- The lack of standard open-access recommendation datasets for scientific fields;

[1] `https://pubmed.ncbi.nlm.nih.gov/`

- The identification of the features for the scientific items to be used in content-based RS;
- How to handle the temporal evolution of scientific fields;
- The large-scale of the scientific databases;
- How to deal with recommender systems where there are entities from multi-fields which may be of interest.

To answer to these challenges, in this thesis we define several Research Questions which aim to shed some light on the challenges:

- **Research Question 1 (RQ1)** May the use of research literature mitigate the lack of recommendation datasets for developing, testing and evaluating recommendation algorithms in scientific fields?
- **Research Question 2 (RQ2)** Does the use of semantic similarity between the Chemical Compounds calculated through ontologies for creating a CB algorithm improve the results of state-of-the-art collaborative-filtering algorithms for implicit feedback recommendation datasets?
- **Research Question 3 (RQ3)** Will the semantic enrichment of sequences of items with the n most similar items improve the results of state-of-the-art sequence-aware recommendations algorithms?
- **Research Question 4 (RQ4)** Will the use of multiple ontologies in the creation of the recommendation dataset in scientific fields improve the performance of state-of-the-art CF algorithms, in particular when comparing with datasets with only one ontology?

The main requirements for testing and evaluating the previous research questions are as follows:

- Access to a list of scientific items;
- Access to databases of research articles;
- Datasets with the preferences of the users for scientific data (user/item rating matrix) for offline evaluation.

To answer RQ1, we will develop a new methodology based on scientific literature for creating recommendation datasets in various fields of science, such as Astronomy and Chemistry. We expect to achieve standard <user,item,rating>recommendation datasets in the studied fields. The created datasets will be evaluated using recommendation algorithms and compared with state-of-the-art recommendation datasets, such as MovieLens.

RQ2 is related to the identification of features and methods for content-based RS. To answer RQ2, we will develop a tool for recommending chemical compounds. Using this tool, we will test how the semantic similarity between the entities of an ontology affects the results of CB, CF and hybrid RS. The used ontology will be the Chemical Entities of Biological Interest (ChEBI) since the entities are chemical compounds. The evaluation will be made using datasets created in RQ1.

RQ3 aims at exploring the temporal and sequential evolution of the scientific fields. To answer it, we will explore state-of-the-art sequence-aware recommendation approaches, and test how the creation of a new hybrid using the similarity between the entities to enrich the sequences will improve the results of the recommendations. These methods will be evaluated using a variation of the datasets created in RQ1, by transforming the standard <user,item,rating>datasets in sequences of items by user, ordered according to the year.

In many fields, specially biomedical fields, there is a close connection between entities. To answer RQ4, we will test how applying Named Entity Recognition (NER) to scientific text for more than one scientific field improves the results of state-of-the-art recommendation algorithms.

The main contributions of this work are:

- Methods and algorithms:
  - A new methodology (LIBRETTI) to create datasets of implicit feedback through the scientific literature, helping researchers to find scientific items of interest (RQ1);
  - A new CB semantic recommender algorithm named ONTO based on ontologies (RQ2);
  - A new hybrid recommender algorithm for datasets of implicit feedback (RQ2);
  - A new hybrid approach (SeEn) for sequence-aware recommendations (RQ3);

  - A multi-field semantic-based pipeline for recommending biomedical entities related to COVID-19 disease (RQ4).

- Datasets and knowledge bases:

  - Novel open-access recommendation datasets in the field of Astronomy for recommending open clusters of stars, and in the field of Chemistry for recommending chemical compounds (RQ1);
  - A database with the semantic similarity between more than 16000 chemical compounds (RQ2);
  - Sequential dataset in the field of Chemistry, for the recommendation of chemical compounds and in the field of Astronomy, for the recommendation of open clusters of stars (RQ3);
  - A database with the similarity between 2000 open clusters of stars (RQ3);
  - A multi-field recommendation dataset with scientific items from four ontologies (ChEBI, Disease Ontology (DO), Gene Ontology (GO),Human Phenotype Ontology (HPO)), created from the CORD-19 dataset (RQ4);

- Tools:

  - cARM - create Astro Ratings Matrix `https://github.com/lasigeBioTM/cARM` (RQ1);
  - CheRM - Chemical Compounds Recommender Matrix `https://github.com/lasigeBioTM/CheRM` (RQ1);
  - SemanticSimDBcreator - Semantic Similarity Database Creator `https://github.com/lasigeBioTM/SemanticSimDBcreator` (RQ2);
  - ChemRecSys - Chemical Compounds Recommender System (`https://github.com/lasigeBioTM/ChemRecSys` (RQ2);
  - A faster semantic similarity calculation for DiShIn library `https://github.com/lasigeBioTM/DiShIn` (RQ2);
  - SeEn: Sequential enrichment of datasets `https://github.com/lasigeBioTM/SeEn` (RQ3);

  - Knowledge-extraction-from-CORD-19 https://github.com/lasigeBioTM/knowledge-extraction-from-CORD-19 (RQ4);
  - RecSys.Scifi: Recommender Systems Datasets in Scientific Fields tutorial https://github.com/lasigeBioTM/RecSys.Scifi.tutorial (RQ4).

The main publications of this work are:
Journal papers:

- Barros, Marcia, Andre Moitinho, and Francisco M. Couto. "Hybrid semantic recommender system for chemical compounds in large-scale datasets." Journal of cheminformatics 13.1 (2021): 1-18. Q1 journal.

  (https://doi.org/10.1186/s13321-021-00495-2)

- Barros, Marcia; Moitinho, André; Couto, Francisco M; Using Research Literature to Generate Datasets of Implicit Feedback for Recommending Scientific Items, IEEE Access, 7, 176668-176680, 2019, IEEE. Q1 journal.

  (https://doi.org/10.1109/ACCESS.2019.2958002)

- Accepted: Barros, M.; Sousa, D., Ruas; P., Couto, F. M; COVID-19 recommender system based on an annotated multilingual corpus. Genomics & Informatics (2021) Q3 Journal.

Conference papers:

- Barros, M.; Lamurias, A.; Sousa, D., Ruas; P., Couto, F. M; (2020, December). COVID-19: A Semantic-Based Pipeline for Recommending Biomedical Entities. In Proceedings of the 1st Workshop on NLP for COVID-19 (Part 2) at EMNLP 2020. Core A conference (http://dx.doi.org/10.18653/v1/2020.nlpcovid19-2.20)

- Barros, Marcia; Moitinho, André; Couto, Francisco M; "Hybrid semantic recommender system for chemical compounds." European Conference on Information Retrieval. Springer, Cham, 2020. Core A conference (https://doi.org/10.1007/978-3-030-45442-5_12)

Other:

- Tutorial: Barros, M., Couto, F. M., Pato, M., & Ruas, P. (2021, August). Creating Recommender Systems Datasets in Scientific Fields. In Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining (pp. 4029-4030). Core A conference (`https://doi.org/10.1145/3447548.3470805`)

- PhD Track: Recommender Systems for Scientific Fields at Symposium on Intelligent Data Analysis 2021. Core A conference

## 1.2 Structure of this document

The remainder of this document is organised as follows: Chapter 2 provides detailed insight about RS, including types of RS, challenges, evaluation, and state-of-the-art. Chapter 3 presents the work for answering RQ1 about the use of research literature for creating recommendation datasets in scientific fields corresponding to the paper *Barros, Marcia; Moitinho, André; Couto, Francisco M; Using Research Literature to Generate Datasets of Implicit Feedback for Recommending Scientific Items, IEEE Access, 7, 176668-176680, 2019, IEEE*. Chapter 4 presents the work for answering RQ2 about the use of the semantic similarity between the chemical compounds in an ontology for improving the recommendation results corresponding to the paper *Barros, Marcia, Andre Moitinho, and Francisco M. Couto. "Hybrid semantic recommender system for chemical compounds in large-scale datasets." Journal of cheminformatics 13.1 (2021): 1-18.* Chapter 5 presents the work for answering RQ3 about the enrichment of sequences for sequence-aware recommendation algorithms. Chapter 6 presents the work for answering RQ4 about the use of diverse sources of knowledge in the improvement of the recommendations corresponding to the paper *Barros, M.; Lamurias, A.; Sousa, D., Ruas; P., Couto, F. M; (2020, December). COVID-19: A Semantic-Based Pipeline for Recommending Biomedical Entities. In Proceedings of the 1st Workshop on NLP for COVID-19 (Part 2) at EMNLP 2020.* Finally, Chapter 7 provides an overall conclusion of this thesis, as well as the future work.

# 2
# Recommender Systems

The main idea of Recommender Systems (RS) is to predict if a user is interested in some item/product. RS are mainly based on information from users' past behaviour, collected from explicit or implicit feedback. Explicit feedback means that the users wittingly indicate if they liked or not of some item, for example, through star system as the one used by Internet Movie Database (IMDB)[1], where 1 corresponds to "did not like", and 10 "liked very much", or a thumbs up or down system, like the one used by Youtube[2]. The interaction of the users with the items allows collecting implicit feedback. For example, watching a movie, searching or purchasing an item indicates a likely interest in that item. To collect implicit feedback, users do not have to actively and accurately indicate that they liked or disliked the item. Implicit feedback data have inherent problems associated:

- there is normally no negative feedback, we cannot know if the user did not like the item she/he saw;
- there is associated noise, for example, items open unintentionally;
- the numerical value of the rating might only refer to a user's preferences with some degree of confidence. For example, we assume that if a user watched a movie till the end, she/he liked it. If she/he left in the first moments, the item was not interesting to this user. But this is just an assumption, and without the explicit indication of interest we cannot know for sure if the user liked or not the item.

[1] `https://www.imdb.com/`
[2] `https://www.youtube.com/`

The information about the preferences of the users in a certain field is the base for RS, allowing the creation of user/item rating matrices, as represented in Table 2.1. For example, from Table 2.1, the user *Chavez R* rated the item (R)-noradrenaline with 1, but she/he did not rate the item caffeine. The goal of the RS is to predict what rating *Chavez R* would give to the item caffeine and decide if this item should or not be recommended to the user.

Figure 2.1 shows the division of the main approaches used in RS: Collaborative-filtering (CF) (model-based, memory-based (user-based, item-based)), Content-based (CB), and Hybrid.

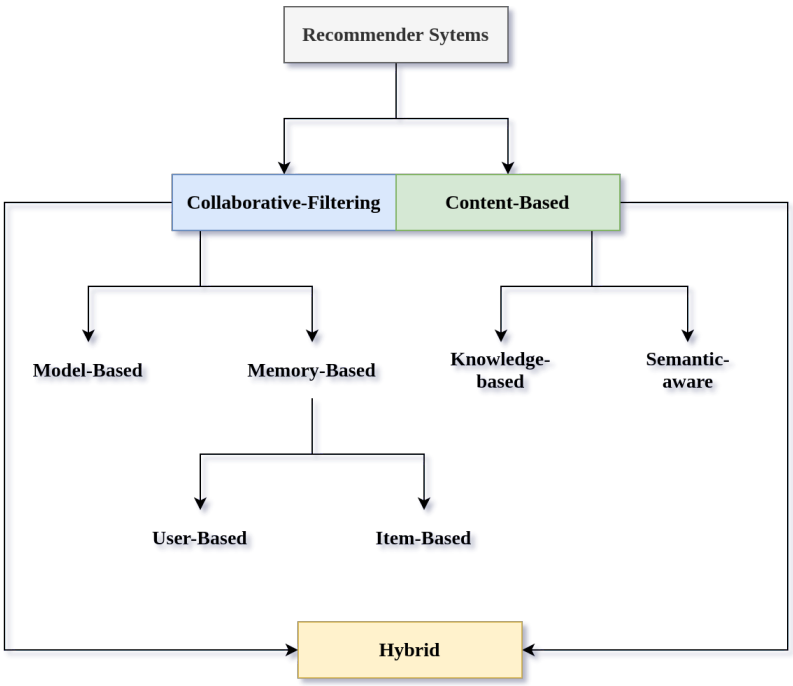


Figure 2.1: Recommender Systems main approaches [20]

## 2.1 Recommender systems approaches

### 2.1.1 Collaborative Filtering

The concept of CF first appeared in Goldberg et al. (1992) [70], with a system called Tapestry. Tapestry allowed to filter the electronic mail using the feedback of other users about

Table 2.1: User/Item rating matrix example.

The columns correspond to items, and the rows to users. The position user/item is the rating a user attributed to an item.

| user/item | (R)-noradrenaline | feruloylacetate(1) | andrastin A | caffeine |
|---|---|---|---|---|
| Chavez R | 1 | 1 | 1 | ? |
| John Smith | 1 | 1 | 2 | 5 |
| Jane Sim | 1 | 5 | 5 | 1 |

the read emails. Here was born the concept of collaborative filtering: using the similarity between the past interests of the users to predict which items they will have interest now.

CF is divided in two methods, memory-based and model-based (see Figure 2.1)[184]. Memory-based methods are divided into CF user-based and CF item-based, as represented in Figure 2.2 [63]. CF user-based compares the patterns of ratings of the users by calculating the similarity between the rows (users) of the rating matrix (Figure 2.2 - left). CF item-based algorithms compare the ratings of items, using the rating matrix columns to find similarities between the way items are rated by a user (Figure 2.2 - right). Memory-based methods use similarity metrics for finding the most similar users.

| | i1 | i2 | i3 |
|---|---|---|---|
| u1 | 1 | 0 | 1 |
| u2 | 0 | 1 | 0 |
| u3 | 1 | 1 | ? |

| | u1 | u2 | u3 |
|---|---|---|---|
| i1 | 1 | 0 | 1 |
| i2 | 0 | 1 | 1 |
| i3 | 1 | 0 | ? |

Figure 2.2: Collaborative-Filtering user-based vs Collaborative-Filtering item-based.
The right rating matrix is transposed.

Some of the most used metrics for calculating the similarity between the users when using a memory-based approach are Cosine similarity (Equation 5.4, where x and y are two non-zero vectors) and the Pearson correlation coefficient (Equation 2.2, where *n* is the sample size, $x_i$ and $y_i$ are the individual sample points indexed with i, and $\overline{x}$ and $\overline{y}$ are the sample mean). For both situations, the closer the results are to 1, the higher is the similarity.

$$cosine\ similarity(\boldsymbol{x}, \boldsymbol{y}) = \frac{\boldsymbol{x} \cdot \boldsymbol{y}}{||\boldsymbol{x}|| \cdot ||\boldsymbol{y}||} \tag{2.1}$$

$$Pearson\ correlation(\boldsymbol{x}, \boldsymbol{y}) = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2 \sum_{i=1}^{n}(y_i - \overline{y})^2}} \tag{2.2}$$

Using the data from Table 2.1 as an example, to predict the rating *Chavez R* would attribute to *caffeine* it is necessary to calculate the similarity between *Chavez R* and *John Smith*, and *Chavez R* and *Jane Sim*. Using Cosine similarity, *Chavez R* - *John Smith* similarity is 0.94, and *Chavez R* - *Jane Sim* is 0.88, thus, the most similar user to *Chavez R* is *John Smith*. *John Smith* rated *caffeine* with 5, which means that the recommender algorithm would predict that *Chavez R* would rate *caffeine* as 5. These methods are also referred to as neighbourhood-based since they use the n most similar users to predict the recommendations. It is easy to justify why an item is being recommended by saying that similar users also were interested in the recommended items.

Model-based methods use a branch of Artificial Intelligence for predicting the recommendations, supervised or unsupervised machine learning, instead of similarity functions, creating a trained model. Machine Learning methods use the information about the past preferences of the users to learn what would be the behaviour of a user before a new item [235]. Training and predicting are two separated phases. Machine learning approaches used to create the models are decision trees, rule-based methods, Bayes classifiers, regression models, support vector machines, and neural networks. The major difference between typical classification machine learning approaches and RS is that in the first case, feature variables and class variables are well defined and separated, train and test are separated, the columns are features, and the rows are instances. In the second case (recommender systems), variables and classes are not well defined since it depends on the entries being considered for the predictions (ratings), the train and test are the items the user already rated and the unrated items, respectively, the columns are items, and the features are users. Still, we may transpose columns and rows for predicting the same item. Compared with memory-based methods, model-based methods require less storage space and are faster in both training and prediction phases [20].

In the past, the method with the best results in several datasets of reference, such as MovieLens and Netflix, was Latent factor models. The goal of these models is to reduce

the dimensionality of the rating matrix. One particular type of latent model is Collaborative-filtering Matrix Factorization (CFMF), which goal is to minimize the least-squares of the rating matrix R and the matrix resultant from the dot product of the user matrix U and item matrix V, according to Equation 2.3 [105].

$$R = U \cdot V^T \tag{2.3}$$

CFMF has been integrated into several recommendation algorithms, such as Alternating Least Squares (ALS) [90], and Bayesian Personalized Ranking (BPR) [161], mainly applied to recommendation datasets of implicit feedback. ALS is a latent factor algorithm that addresses the confidence of a user-item pair rating, which goal is to minimize the least-squares error of the observed ratings by factorizing the rating matrix in user and item matrix. ALS has the advantage of being easily parallelized. Some recent studies focused on speeding up the implementation of this algorithm [78, 115]. BPR is also a latent factor algorithm, but it is more appropriate for ranking a list of items. BPR does not just consider the unobserved user-item pairs as zeros but also discerns a user's preference between an observed and an unobserved rating. Several studies have been using BPR in the recommendation of items from implicit feedback datasets.

However, nowadays, Artificial Neural Networks (ANN) development overcome CFMF methods in the recommendation of items [233]. ANN is a branch of machine learning, which goal is to mimic the connections of the brain. The main units are neurons activated by the inputs of the system.

Some well known ANN methods used in RS are Neural Collaborative-filtering [82], Collaborative Denoising Auto-encoder (CDAE) [223], Deep Matrix Factorization [224], Recurrent Neural Networks based RS, for example, GRU4Rec [87], and neural attention models, such as BERT4Rec Sun et al. [185].

Despite the evolution of the model-based CF methods, their major downgrade is the lack of explainability of the recommendation since the models tend to be black boxes. We only are aware of the input and the output, without further explanations about the recommended items. In general, a big challenge of CF approach is the cold start problem for new items and new users and data sparsity. We have cold start when a new user did not rate any item in the dataset, or a new item was not yet rated by any user. Section 2.2 address both challenges.

### 2.1.2 Content-based

CB RS do not need the similarity between users to recommend items. Instead, this method uses items properties to predict the rating a user would attribute to an unrated item (Figure 1.2). The essential data sources of CB algorithms are a user with rated items and well-defined properties/features for those items. One challenge is converting unstructured data, such as text from reviews, into a structured dataset of features. Some items have obvious features. For example, when the items are movies, the features used to find the most similar items may be the genre, director, and authors. In other fields, the task of finding features for the items is not that obvious. Thus, one of the tools used by CB for this purpose is ontologies [190], which provide controlled vocabularies of terms and definitions to represent the entities of a specific field of study [28, 198].

The ratings from other users play no role in the CB approach. CB is the appropriate approach for new items that have zero ratings from users. New items will be recommended given that the user already had liked a similar item. The goal of CB is to find the most similar items to the ones the users already saw, which may be achieved using Nearest Neighbor Classification. The similarity between items may be calculated, for example, using Cosine similarity function (Equation 5.4), or, in the case of structured data, the Euclidean distance (Equation 2.4, where *n* is the sample size, $x_i$ and $y_i$ are the individual sample points indexed with i) is more appropriate [149]. The similarity between the items may also be calculated using machine learning methods, such as clustering [21].

$$EuclideanDistance = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{2.4}$$

One of the drawbacks of the CB approach is the lack of novelty and diversity. CB algorithms only recommend similar items to the ones the user already liked. CB does not have the problem of cold start for new items; however, the problem exists for new users [123]. CB approaches cannot deal with new users without any rated item.

A particular type of CB algorithm is Knowledge-based (KB). KB RS does not use ratings to provide recommendations. Instead, this approach is based on similarities between the requirements of the users and the description of the items. This approach is useful, for example, in the case of items that are not purchased very often, such as houses or luxury goods, or items with complex attributes, such as cars. In this case, the items do not have

enough ratings, and the complexity of the domain makes it harder to extract the desired features to compare. The best solution is for users to provide explicit information about the properties of the items they are looking for [19]. KB does not have the problem of cold start, but it is limited to the explicit information provided by the users, lacking the novelty.

Another type of CB RS is semantic-aware RS. This type of approach makes use of semantic relationships between items, for example, through domain-specific knowledge such as ontologies and Linked Data [56]. This approach is helpful for connecting items and users and extracting features from External Sources of Knowledge, improving and enriching the recommendations.

### 2.1.3 Hybrid

All the previous described RS approaches have inherent problems. Thus, many systems are developed as hybrid RS, which use at least two methods for creating the recommendations.

Hybrid methods allow improving the results from single techniques. For example, a hybrid system between CF and CB will mitigate the cold start problem for new items and the diversity problem. A hybrid between CF and KB will eliminate both new item/new user cold start problem [17]. The challenge is to test which method, or combination of methods, provide the best result for each situation.

When creating the hybrid RS, we may use monolithic, ensemble, or mixed designs. The monolithic uses several data types, not existing a clear distinction between the content-based and collaborative-filtering modules. For example, monolithic can use *feature augmentation*, where the features from various sources are combined, and *Meta-level*, where one RS uses as input the model created by another RS.

The ensemble design consists of combining the results of two separated recommendations algorithms. *Weighted* methods combine the scores of different recommender algorithms into a final score by weighing the scores. Some metrics are shown in Equations 2.5 and 2.6, where $S_{\text{CFI1}}$ is the score obtained for item 1 using a collaborative-filtering algorithm, and $S_{\text{CBI1}}$ is the score for item 1 obtained with a CB algorithm.

$$Metric1 = S_{\text{CFI1}} \times S_{\text{CBI1}} \tag{2.5}$$

$$Metric2 = \frac{S_{\text{CFI1}} + S_{\text{CBI1}}}{2} \tag{2.6}$$

Other types of ensemble hybrid RS are *Switching*, and *Cascade*. *Switching* consists of switch between different algorithms, depending on the needs at the moment. In *Cascade* methods, the results of one algorithm are refined by the results of a previous one, creating a cascade of algorithms.

### 2.1.4 Recommender systems and ontologies

The notion of ontology is not new and has long been used for classifying and describing concepts. At the time of the rising of the semantic web, ontologies were adapted to computational reasoning and knowledge sharing since they are normally expressed as OWL which structured format (triplets of subject, predicate and object) makes them ideal for computer processing. More recently, ontologies were adapted to the biological/biomedical domain. Some examples of well-known bio-ontologies are the Chemical Entities of Biological Interest (ChEBI) [2, 81], the Gene Ontology (GO) [9, 49], and the Disease Ontology (DO) [7, 169]. Bio-ontologies are particularly important for providing a unique identifier for biomedical entities. The name of biomedical entities may change over time, and different researchers may refer to them differently. One of the advantages of ontologies is storing lists of these descriptors.For example, for the chemical entity caffeine [3], chEBI identifies more than 20 synonyms. Another significant advantage of the ontologies is that we can relate the entities through their semantic similarity, a measure based on the ontology's semantic structure.

In the recommendation systems field, ontologies are used for representing knowledge about the items and users in the recommendation process. The most common task is the creation of user and items profiles. Developing user's profiles as ontologies allows to use semantic similarity metrics to find the most similar users in CF approaches. Ontological items' profiles allow to calculate the semantic similarity between the items, which may be used subsequently into a CB approach [68, 73, 109, 212]. Some well known similarity metrics are the Resnik [163], Lin [120], and Jiang and Conrath (JC) [96]. These measures are based on the information content of the entities, given by the probability of the entity appears in the ontology, and in the shared information content, calculated from the common ancestors. Resnik and Lin are real similarity measures, whereas JC is a distance measure,

subsequently converted to similarity. Lin and JC have a range between zero and one. The higher the value, the more similar the entities are.

Ontologies are also often used for the Named Entity Recognition (NER) and Named Entity Linking (NEL) tasks, both branches of the information extraction field [72]. In these cases, the ontologies are used as dictionaries of terms for searching in text and retrieve the entities related to the field of the ontology, also linking different words to the same entity. For example, we may use the ChEBI ontology for searching chemical compounds in research articles. In RS, NER with resource to ontologies has been performed mostly for extracting information from text, for example, from reviews, to improve the recommendations.

### 2.1.5 Sequence-aware recommendations

Typical recommendation datasets are represented as matrix format, with items in the columns, users in the rows, and the ratings assigned to the pairs <user, item>. However, some situations require knowledge about the order in which the items were seen, especially in scientific fields, where the scientific entities raise different degrees of interest to the researchers over time. For example, according to Pubmed[1], the chemical compound Fluvoxamine[2] was losing research interest and now it is increasing again, as shown in Figure 2.3. This may be because Fluvoxamine has been investigated in the context of COVID-19 [26].

To assess the recommendations, the rating matrix is translated for sequences ordered by interaction time, and the goal is to predict the best next item [157]. Sequence-aware recommendations have been developed and applied for movies, music, e-commerce, but to the best of our knowledge, not in scientific fields. There are already algorithms dealing with sequential recommendations. There are some common baselines, such as the most popular, and k-nearest-neighbours approaches. We also have non-deep learning approaches, such as matrix factorization and Markov chains [175]. Most recently, deep learning approaches have emerged as state-of-the-art for sequence-aware recommendations, such as, GRU4Rec [87], CASER [188], SASRec [97] and BERT4Rec [185]. The last one outperformed all the other algorithms. BERT4Rec is based on the famous BERT model from Google. Its major difference from other deep learning algorithms is that it is bidirectional, reading the sequences from left to right and right to left. The first step of BERT4Rec is an embedding layer, where

[1] `https://pubmed.ncbi.nlm.nih.gov/?term=Fluvoxamine`
[2] `https://www.ebi.ac.uk/chebi/searchId.do?chebiId=CHEBI:5138`

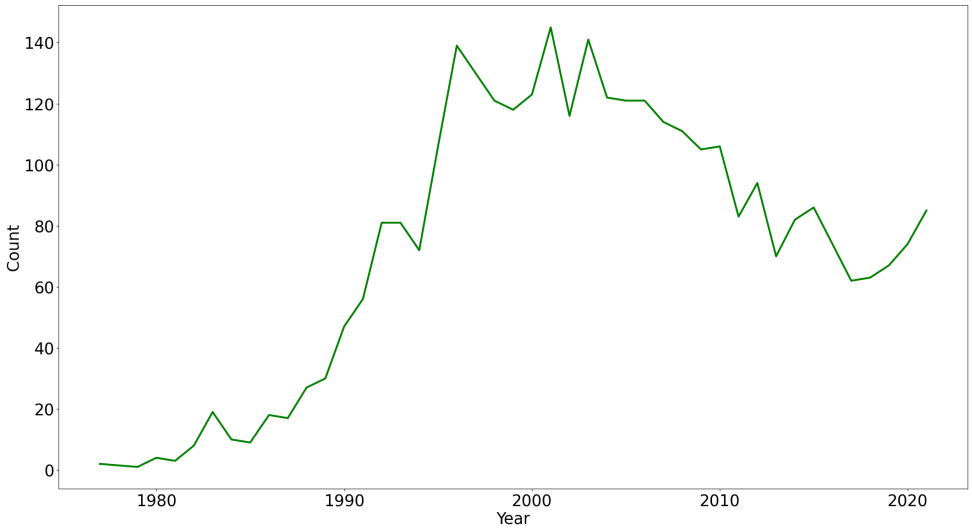


Figure 2.3: Fluvoxamine research articles by year in Pubmed.

it combines the position and the item, and then several transformer layers. The transformer method is a deep learning model for Natural Language Processing (NLP), based on multi-head self-attention and another layer of position wise feed forward. BERT4Rec has several transformer layers, and they are connected bidirectionally. A percentage of the items are masked in the sequence for training, which increases the number of training examples. The output has the probability for the next items.

## 2.2 Challenges

As seen in Section 2.1, each recommender approach has its challenges, such as cold start and sparsity of data. From the analysis of the state-of-the-art (Table 2.3), four significant problems were identified: cold start, the sparsity of data, incorrect recommendations, and the scalability of the algorithms [165].

Cold start happens when a new item or a new user enters the RS. New items do not have any feedback from users. Thus, for example, in the CF approach, new items without ratings will never be recommended. That is why the CB approach does not have the cold start problem for new items since CB RS recommend the items based on their properties.

New users have similar problems since they have not yet rated any item. Thus, the system does not know their preferences, and it cannot find the most similar user to this new user. Cold start for new users is a challenge in both CF and CB approaches.

The sparsity of data is related to the lack of ratings, affecting the completion of the rating matrix. For example, in a system with thousands of items, each item will have few or even no ratings, making it challenging to find the best recommendations based on other users' ratings. This is a problem of CF approach.

Scalability is another issue often found in RS, especially in CF memory-based approaches. Due to the increased number of items and users, some RS algorithms are not ready for Big Data problems because they cannot analyse the data in real-time to provide recommendations on the fly. Finally, another problem often found in RS is incorrect recommendations. All approaches face this problem, and it is the goal of many studies to improve the accuracy of a system.

## 2.3 Evaluation Methods

There are several methods for evaluating the performance of a RS, depending on the available resources and on the goal of the RS itself. Suppose we have the RS running on a platform, such as YouTube[1] or IMDB[2]. In that case, we may perform online tests by implementing two algorithms, randomly attributing them to the users, and measuring the recommendations' clicking rate. These are known as A/B tests. However, in most cases, we have only access to offline datasets, i.e., datasets with the past information of the users' preferences. Despite the disadvantage of not having access to the users' immediate preferences, using offline datasets provide the chance to test and evaluate new recommendation algorithms without the extra work of developing an online platform and interacting with real users. Also, testing the algorithms offline indicates the best algorithm to be subsequently implemented in online platforms. Thus, offline evaluation requires a dataset with the users' preferences for splitting into train and test sets. The goal is to predict the best items for each user and then use the test set for confirming if the recommended items are relevant for the user [170, 174].

[1] `https://www.youtube.com/`
[2] `https://www.imdb.com/`

Depending on the goal of the algorithm, the type of evaluation will be different. There are algorithms whose goal is to predict the rating a user would give to an item and others whose goal is to recommend a ranked list of items, i.e., the top@k items, where k is the size of the list. In the first case, these algorithms are evaluated for the predicted rating, using metrics such as Mean Squared Error (MSE - Equation 2.7), and Root Mean Squared Error (RMSE - Equation 2.8). MSE measures the average of the squared difference between the real rating of an item and the rating predicted by a recommender algorithm for all *n* items being analyzed. RMSE is calculated the same way as MSE, but the application of the squared root allows a better interpretation of the results.

$$MSE = \frac{1}{n}\sum_{t=1}^{n}(y\_real - y\_pred)^2 \tag{2.7}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y\_real - y\_pred)^2} \tag{2.8}$$

In the second case, when the algorithms return a ranked list of items, these may be evaluated for the number of relevant items recommended, for example, through Precision (Equation 2.9), Recall (Equation 2.10), and F-Measure (Equation 2.11), and Hit Ratio, and for the quality of the ranking, through Mean Reciprocal Rank (Equation 2.12), and Normalized Discounted Cumulative Gain (Equation 5.2).

$$Precision@k = \frac{relevant\_items@k}{k} \tag{2.9}$$

$$Recall@k = \frac{relevant\_items@k}{total\_relevant\_items} \tag{2.10}$$

$$F\_measure@k = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{2.11}$$

$$MRR = \frac{1}{n\_users}\sum_{i=1}^{n\_users}\frac{1}{rank_i} \tag{2.12}$$

$$DCG = \sum_{i=1}^{n}\frac{relevance_i}{\log_2(i+1))} \tag{2.13}$$

$$nDCG = \frac{DCG}{iDCG} \tag{2.14}$$

Precision@k provides a measure of the relevant items recommended in the top@k list, recall@k a measure of relevant items recommended in the top@k list, and f-measure provides a harmonic mean of precision and recall. The MRR evaluates in which position the first relevant item appears. The nDCG is an evaluation method that compares the ideal ranking of a test set (iDCG), with the ranking assigned by the recommendation algorithm (DCG - Equation 5.1) [170]. The DCG measures the relevance of an item based on its position in the recommendation list. The Hit Ratio evaluates the number of relevant items in a list of recommendations.

Another critical issue in evaluating a RS is the splitting method used for dividing the dataset into training and testing sets. The most used methods are hold-out and cross-validation. In the hold-out method, the dataset is divided into $\alpha\%$ for training and $1-\alpha\%$ for testing. In the cross-validation method, the dataset is divided into q equal sets, and in each evaluation, we use q-1 sets as training data and 1 set as testing data. Each evaluation has different dataset sets, ensuring that all the dataset is tested and avoiding over-fitting. This method does not require a validation set [18]. The validation set is only needed when cross-validation is used simultaneously for the selection of the best set of hyperparameters and for error estimation [55], which does not happen in many studies on RS [90, 161, 172]. For sequence-aware RS, the most common evaluation method is the leave-one-out, by hiding the last item in the sequence for testing. Leave the last out is the most appropriate method since sequence-aware RS aim to predict the next best item. Usually, for the validation, it is hidden the last but one item.

## 2.4 State-of-the-art

### 2.4.1 General Recommender Systems State-of-the-art

RS are a vast and diverse field of study. Along the years, there have been a growing number of research articles, which translate into several surveys about RS [14, 34, 40, 44, 98, 184]. Burke (2002) [44] surveyed the existent hybrid RS at the time. The authors of the survey concluded that less than half of the 41 possible hybrids RS had been explored, which is expected since RS were at the beginning of their journey. Advancing towards

2005, Adomavicius and Tuzhilin (2005) [15] described the state-of-the-art until that period, emphasising the limitations of RS, such as limited content analysis, over-specialisation, cold start for new users and new items, and the sparsity of the rating matrix.

Due to the growing use of CF approaches, Su and Khoshgoftaar (2009) [184] published a review on the topic, describing CF methods, challenges, and evaluations metrics. The conclusions drawn from this survey were that CF is a widely used technique to provide recommendations through memory-based methods, model-based methods or combined with other techniques in a hybrid RS.

In Bobadilla et al. (2013) [40] we see a shift in the source of the data being used to provide the recommendations. In the Era of Web 2.0, there is by this time social information about the personal interests of the users, for example, from Facebook and Twitter, that can be used to improve and personalise the recommendations. The authors also suggest that in the future, RS will use the information provided by the integrated devices on the Internet (Internet Of Things [219]). They also draw readers attention to the new lines of research with RS, including the possibility of use RS to help in data visualisation and exploration.

Since RS have been so widely used in many different fields, surveys about the application of RS in specific fields started to arise, such as the use of RS in the recommendation of scientific papers [34], and the use of RS in Health Informatics [242]. Beel et al. (2016) [34] surveyed more than 200 research papers about RS, considering a period of 16 years. Most of these RS applies to the recommendation of books, education, academic alert services, expert search, venue recommendations, academic events, patents, and even plagiarism detection. From this survey, the authors concluded that CB had been the most used approach to provide the recommendations in the research papers RS field. Most of the RS use implicit ratings due to the lack of explicit ratings from users.

The rest of this section provides an insight into the most cited RS research published between 2015 and 2020, inclusive. The articles were selected, considering that they had 100 or more Google Scholar citations until July 2021. They were original studies with a new algorithm or method in the field of RS. They had as goal solving one RS challenge. Table 2.3 shows the selected articles, highlighting the problems studied, the approaches and the datasets used in each case.

Table 2.3: Most recent articles in Recommender Systems. Problem: CS - cold start; SD - Sparse Data; ImpR - Improve Recommendations; Scale - scalability; other. Approach: CF - collaborative filtering; CB - content-based; hybrid. (NN): the approach includes the use of Neural Networks. (KG): the approach includes the use of knowledge graphs.

| Authors | Year | Challenge | Approach | Dataset |
|---|---|---|---|---|
| Pereira and Hruschka [150] | 2015 | CS | Hybrid | MovieLens, Jester, Netflix |
| Guo et al. [75] | 2015 | CS | CF | Flixster, FilmTrust and Epin-ions |
| Wang et al. [206] | 2015 | SD | Hybrid (NN) | CiteULike, Netflix |
| Thong et al. [192] | 2015 | ImpR | CF | Medical Records |
| Martinez-Cruz et al. [131] | 2015 | CS | Hybrid (KG) | Epinions |
| Zahra et al. [228] | 2015 | Scale | CF | MovieLens, FilmTrust, Book-Crossing, LastFM |
| Al-Hassan et al. [22] | 2015 | ImpR | Hybrid (NN) | tourism services |
| Hernando et al. [85] | 2016 | Scale | CF | MovieLens, Netflix |
| Zhang et al. [231] | 2016 | SD | Hybrid (NN) | MovieLens, IntentBooks |
| Cheng et al. [48] | 2016 | SD | Hybrid (NN) | Google Play Apps |
| Wu et al. [223] | 2016 | ImpR | CF (NN) | MovieLens, Netflix, Yelp |
| Kim et al. [103] | 2016 | SD | Hybrid (NN) | MovieLens, Amazon |
| Gong and Zhang [71] | 2016 | ImpR | CF (NN) | microblog |
| Song et al. [179] | 2016 | ImpR | CF (NN) | News |
| Wu et al. [220] | 2017 | ImpR | CF (NN) | Netflix, IMDB |
| Zhang et al. [232] | 2017 | ImpR | CF | movieLens |
| Wei et al. [215] | 2017 | CS | Hybrid (NN) | Netflix |

( To be continued)

| Authors | Year | Challenge | Approach | Dataset |
|---|---|---|---|---|
| Zheng et al. [240] | 2017 | SD | CF (NN) | Yelp, Amazon, Beer |
| He et al. [82] | 2017 | SD | CF (NN) | MovieLens, Pinterest |
| Zhang et al. [238] | 2017 | ImpR | CF | Yelp |
| Xue et al. [224] | 2017 | ImpR | CF (NN) | MovieLens, Amazon movies, Amazon music |
| Li et al. [114] | 2017 | ImpR | CF (NN) | Yoochoose, Diginetica |
| Okura et al. [140] | 2017 | ImpR | Hybrid (NN) | Yahoo! JAPAN's homepage |
| Beutel et al. [36] | 2018 | ImpR | Hybrid (NN) | Youtube |
| Ebesu et al. [60] | 2018 | ImpR | CF (NN) | Epinions, citeulike-a, Pinterest |
| Wang et al. [207] | 2018 | CS | Hydrid (KG) | MovieLens, Book-Crossing, Bing-News |
| Ying et al. [227] | 2018 | Scale | Hybrid (NN, KG) | Pinterest |
| Chen et al. [47] | 2018 | ImpR | CF (NN) | Amazon |
| Kang and McAuley [97] | 2018 | ImpR | CF (NN) | Amazon games, amazon beauty, steam, movieLens |
| Wang et al. [208] | 2018 | ImpR | CB (NN, KG) | logs of Bing News |
| Wang et al. [209] | 2019 | ImpR, cold-star | CF (NN, KG) | MovieLens, Book-Crossing, Last.FM, Dianping.food |
| Fan et al. [67] | 2019 | ImpR | CF (NN, KG) | Ciao, Epinions |
| Wu et al. [221] | 2019 | SD | CF (NN) | Yelp, Flickr |
| Sun et al. [185] | 2019 | ImpR | CF (NN) | Beaut, MovieLens, Steam |

( To be continued)

| Authors | Year | Challenge | Approach | Dataset |
|---|---|---|---|---|
| Wu et al. [222] | 2019 | ImpR | CF (NN, KG) | Yoochoose, Diginetica |
| Tang [187] | 2019 | ImpR | CF (Quantum) | theoretical paper |
| He et al. [83] | 2020 | ImpR | CF (NN, KG) | Gowalla, Yelp2018, Amazon-Book |

Analysing Table 2.3, the most addressed problem is the correctness of the recommendations, with 22 out of 37 articles trying to improve the quality of the recommendations. The sparsity of data and cold start are discussed in 7 and 5 papers, respectively. The scalability of recommender algorithms is addressed in 3 articles.

As seen previously, RS intervene in a broad range of areas, and the articles in Table 2.3 are the reflection of the different sources of data. Nonetheless, movies datasets are the most used to implement and evaluate RS. Eighteen research articles used movies datasets, particularly MovieLens (13 articles) and Netflix (6 articles). After 2016, we can see an increase in the use of datasets from Amazon. The other datasets are from the fields of books, opinions, and tourists information. In this list of research articles, only one study uses data from scientific fields by using a dataset of medical records. The stark difference between the number of studies using movies, books, and e-commerce datasets for testing and evaluating new recommendation algorithms may be related to the fact that there are a large number of open-source datasets in these specific fields. In scientific fields, as we will see in the next section, public and available datasets may be a challenge.

About the recommendation approaches used in the studies, CF is the most used, with 25 articles developing new CF methods. This means that a significant number of RS are using the information about similar users' preferences to recommend the items.

Twelve articles developed hybrid approaches described bellow, all using feature combination to provide the recommendation:

- Pereira and Hruschka (2015) [150] combines CF recommendations with demographic information.
- Wang et al. (2015) [206] uses CF and explores auxiliary information with deep learning.

- Martinez-Cruz et al. (2015) [131] uses CF, since it mentions the use of trust between users, and CB through the implementation of ontologies.
- Al-Hassan et al. (2015) [22] uses semantic knowledge of items to enhance the CF recommendation quality.
- Zhang et al. (2016) [231] makes use of CF and knowledge-based approaches.
- Cheng et al. (2016) [48] combines CB with CF, using deep learning to find content information.
- Kim et al. (2016) [103] developed a method called convolutional matrix factorization (ConvMF) that integrates convolutional neural network for extracting contextual information of documents into probabilistic matrix factorization (PMF).
- Wei et al. (2017) [215] uses CB and CF combined approaches, with deep learning for feature extraction.
- Okura et al. (2017) [140] study uses representations of articles based on a denoising autoencoder (CB), generate user representations by using a recurrent neural network (RNN) (CF) and match and list articles for users based on inner-product operations.
- Beutel et al. (2018) [36] incorporate contextual data (CB) into a CF Recurrent neural network.
- Wang et al. (2018) [207] uses knowledge graph as the source of side information to complement the CF approach.
- Ying et al. (2018) [227] uses knowledge graphs (KG) and NN to create embedding for the items with resource to features, such as images and text. The embedding is used in CF algorithms.

Looking at Table 2.3, we can also verify the increasing use of KG in the most recent years. Eight of the articles use KG in their approaches to create the recommendations. KG are used in CB approaches to calculate the similarity between the items, in CF approaches to create connection paths between users and items, and in hybrid approaches, where KG are used both for finding the similarity between the items using their shared features for

creating connections between the users. Martinez-Cruz et al. (2015) [131] developed an ontology to characterize the trust between users using fuzzy linguistic modelling. Wang et al. (2018)[207] constructed KG to link the items through their features. Ying et al. (2018)[227] uses KG to create embedding for the items. Wang et al. (2018)[208] is a content-based RS that uses the KG to connect the information in news articles and to find similar articles. In Wang et al. (2019)[209] a KG is used to connect users and items, creating paths between them. Fan et al. (2019)[67] developed the GraphRec framework, which creates KG of user-user and user-item interactions for social recommendations. Wu et al. (2019)[222] focus on session-based recommendations. The sessions are modelled as knowledge graphs for improving the recommendations. He et al. (2020)[83] developed LightGCN, a simpler implementation of neural graphs collaborative filtering.

Together with KG, Neural Networks (NN) have been recognized as state-of-the-art approaches in RS, as can be seen in Table 2.3. These two approaches appear often associated since it is easy to include external sources of knowledge into NN to extract connections that would be much harder to find with simple CF methods. This conclusion is predictable since Big Data widely use NN [89], and most of the fields that use RS are producing large amounts of data every day. NN may help improving scalability problems.

This is the most recent scenario in the state-of-the-art recommendation systems. CF leads the rank compared to CB methods, probably because it does not need extra information about the items. KG and NN are gaining supporters compared to standard RS approaches, and the fields of movies are still the most used for testing and evaluating new recommendations algorithms. From the studies presented in Table 2.3, only one is assessed in a scientific fields [192].

### 2.4.2 Scientific fields Recommender Systems State-of-the-art

Next, we present the state-of-the-art for recommendation systems in scientific fields, such as Chemistry, Health, Life Sciences, and Astronomy. Table 2.4 shows in greater detail important research studies from the scientific field using RS. It provides information about the scientific field, what is being considered as users and items, the recommendation approach, and if the dataset used in the study is available for reproducibility proposes (if it is possible to download and use the dataset).

Table 2.4: Background studies about the use of recommender systems in scientific fields.

| Article | Year | Field | Dataset | Users | Items | Approach | Available |
|---|---|---|---|---|---|---|---|
| Owen et al. [146] | 2003 | Genetics | C. elegans DNA microarray | Genes | Genes | CF | No |
| Ng et al. [138] | 2007 | Genetics | Stanford Microarray and Gene Expression Omnibus | Pathway | Genes | CF | Yes |
| Torkaman et al. [195] | 2011 | Health | Flow Cytometry | Clinical hematologists | Disease | CF | No |
| Bostrøm et al. [43] | 2011 | Drugs | AstraZeneca R&D Mölndal stock reagents | Chemists | Reagents | CF | No |
| De Smet et al. [53] | 2013 | Genetics | INSDC & LTP | Bacteria and archaeal type strains | RNA | CB | No |
| Wiesner and Pfeifer [217] | 2014 | Health | Gold standard of human expert recommendations | Health Record System | Health info | CB | No |

( To be continued)

| Article | Year | Field | Dataset | Users | Items | Approach | Available |
|---|---|---|---|---|---|---|---|
| Ishihara et al. [93] | 2015 | Chemical compounds | Collection of diazepane derivatives | Chemical compounds | Free-Wilson-like | CF | No |
| Zhang et al. [237] | 2015 | Drugs | Drug Database | Customers | Drugs | CF | No |
| Macedo et al. [128] | 2016 | Health | Simulated patients' medical records | Healthcare professionals | Health info | CB | No |
| Hao and Blair [76] | 2016 | Health | NHANES | patients | Health info | CB | Yes |
| Corrado et al. [50] | 2016 | Genetics | Human RBP-RNA interactions | RNA binding proteins | RNA | CF | Yes |
| Chen et al. [45] | 2016 | Health | Electronic Medical Record | Healthcare professionals | Health info | CF | No |
| Savage et al. [167] | 2017 | Chemical compounds | Chemical reactions | Chemical compounds | Reactants | CF | No |
| Mustaqeem et al. [135] | 2017 | Health | Electronic Medical Record | Patients | Health info | Hybrid | No |
| Bocanegra et al. [41] | 2017 | Health | Health-related videos, SNOMED-CT, Bio-ontology | Health consumers | Health educational | CB | Yes |

( To be continued)

| Article | Year | Field | Dataset | Users | Items | Approach | Available |
|---|---|---|---|---|---|---|---|
| Gr ßer et al. [74] | 2017 | Health | Electronic Medical Record | Patients | Health info | CF | No |
| Chen et al. [46] | 2017 | Health | Electronic Medical Record | Healthcare professionals | Health info | CF | No |
| Fan et al. [66] | 2017 | Drugs | CNS side effects | Drugs | CNS side effects | CF | No |
| Ezzat et al. [65] | 2017 | Drugs | G-Protein Coupled Receptors (GPCR), Ion Channels (IC), Nuclear Receptors (NR) and Enzymes (E) | Drugs | Targets | CF | Yes |
| Peska et al. [152] | 2017 | Drugs | G-Protein Coupled Receptors (GPCR), Ion Channels (IC), Nuclear Receptors (NR) and Enzymes (E) | Drugs | Targets | CF | Yes |

( To be continued)

| Article | Year | Field | Dataset | Users | Items | Approach | Available |
|---|---|---|---|---|---|---|---|
| Ozsoy et al. [147] | 2018 | Drugs | DrugBank, PubChem & UMLS | Drugs | Disease | CF | No |
| Yang et al. [225] | 2018 | Nutrition | Users' info | Users | Meals | CB | No |
| Wang et al. [203] | 2018 | Drugs | Genomics data | Drugs | Disease | CF | No |
| Seko et al. [171] | 2018 | Chemical compounds | Inorganic databases | Inorganic compounds | Chemical relevant compositions | CB | Yes |
| Suphavilai et al. [186] | 2018 | Drugs | CCLE & GDSC | Cell-lines/patients | Drug response | CF | Yes |
| Pustozerov et al. [155] | 2018 | Health | Survey patients | Patients | Disease | CF | No |
| Iatraki et al. [91] | 2018 | Health | Electronic Medical Record | Patients | Health documents | Hybrid | No |
| Agapito et al. [16] | 2018 | Nutrition | Electronic Medical Record | Patients | Nutritional advice | Hybrid | No |
| Liu et al. [121] | 2018 | Drugs | CCLE & GDSC | Cell-lines/patients | Drug response | CF | Yes |
| Katzman et al. [99] | 2018 | Health | Electronic Medical Record | Patients | Treatments | CF | Yes |
| Wittich et al. [218] | 2018 | Biology | Vascular plants | Territory | Plant taxa | CF | Yes |

( To be continued)

| Article | Year | Field | Dataset | Users | Items | Approach | Available |
|---|---|---|---|---|---|---|---|
| Yasuo et al. [226] | 2018 | Drugs | DrugBank | Drugs | Protein | Hybrid | No |
| Kim et al. [104] | 2018 | Disease | TCGA-PRAD | Genes | Disease | CF | Yes |
| Hao et al. [77] | 2018 | Drugs | DrugBank | Drugs | Targets | CF | No |
| Srinivas et al. [183] | 2018 | Chemical compounds | CHEMBL | Chemical compounds | Targets | CF | Yes |
| Torrent-Fontbona and López [197] | 2019 | Health | Virtual subjects | Patients | Insulin | CB | No |
| Balvert et al. [27] | 2019 | Drugs | GDSC | Cell-lines/patients | Drug response | CF | Yes |
| Zeng et al. [229] | 2020 | Disease | | Genes | Disease | CF | Yes |
| Wang et al. [204] | 2020 | Drugs | GDSC | Cell-lines/patients | Drug response | CF | Yes |
| Lan et al. [110] | 2020 | Drugs | DrugBank | Drugs | Enzyme proteins | CF | Yes |
| Mustaqeem et al. [136] | 2020 | Disease | Electronic Medical Record | Patients | Disease | CF | No |
| Emdadi and Eslahchi [64] | 2020 | Drugs | CCLE & GDSC | Cell-lines/patients | Drug response | CF | Yes |
| Lim and Xie [118] | 2021 | Drugs | ChEA | Genes | Genes | CF | Yes |

( To be continued)

| Article | Year | Field | Dataset | Users | Items | Approach | Available |
|---|---|---|---|---|---|---|---|
| Ren et al. [159] | 2021 | Health | Electronic Medical Record | Patients | Search terms | Hybrid | No |
| Astronomy | | | | | | | |
| Mukund et al. [134] | 2018 | Astronomy | Open source logbook data from the Laser Interferometric Gravitational Observatory (LIGO) | Users | Queries | CB (NN) | Yes |
| Hinkel et al. [88] | 2019 | Astronomy | Hypatia Catalog | Host stars | Giant exoplanet | CB | Yes |
| Kerzendorf [101] | 2019 | Astronomical Literature | arXiv articles | Research articles | Research articles | CB | Yes |
| Malanchev et al. [129] | 2021 | Astronomy | Zwicky Transient Facility (ZTF DR3) | Astronomers | Anomalies | CB | Yes |
| Teimoorinia et al. [191] | 2021 | Astronomy | images from the MegaCam | Astronomical images | Label (Good, Bad) | CB | Yes |

Analysing Table 2.4, we can find diverse scientific fields where RS were used, such as Genetics, Health, Drugs, Chemistry, and outside biological and biomedical fields, Astronomy. The use of RS in these fields is substantially different from the use of RS in the studies presented in Table 2.3. Typically, a RS goal is to recommend the most appropriate items to a user. However, in scientific fields, the concept of item and user may be different. In datasets such as MovieLens, the recommendations algorithms have users (people) who manifested their preferences about an item. In scientific fields, the definition of user and item may be wider. We have studies recommending genes to genes [118, 138, 146], or diseases to genes [104, 229]. A field with a large number of studies is the Drugs field. Here we have as items targets, side effects, diseases, drugs responses, and proteins being recommended to drugs [27, 64, 65, 66, 77, 110, 121, 147, 152, 186, 203, 204, 226]. The recommendation of chemical compounds or to chemical compounds is also well represented in the Table 2.4 [93, 167, 171, 183].

In Astronomy, some studies have emerged in recent years. In [101], the author developed a tool whose goal is to find similar articles based only on text content from an input article. The dataset used to create the tool was collect from the ArXiv [1]. The author argues that the tool performs robustly and finds relevant articles that are not discovered by other platforms via citations, references or suggestions from SAO/NASA Astrophysics Data System (ADS)[2]. However, the authors do not provide any evaluation measure for the system, providing only isolated examples. [134] implemented a different approach for the recommendation. The authors developed a RS for Astronomical observatories. When a user introduces a query related to an instrument, the system recommends logs written by other researchers, providing positive and negative feedback. Recommending also the bad feedbacks allows that new researchers do not make the same mistakes as others. To test the system, they used open-source logbook data from the Laser Interferometric Gravitational Observatory (LIGO). The system's performance was tested using six months of logbooks by comparing the retrieved logbooks with actual relevant entries, with the system retrieving most of the entries correctly. Despite these promising results, the authors do not present the results using standard metrics, such as precision and recall. They do not provide a baseline for comparison, for example, how a random recommender would perform in the test set. In [88], the goal of the study is to recommend stars that may host giant planets, based on the elements found on the host

[1] https://arxiv.org/
[2] https://ui.adsabs.harvard.edu/

stars. The authors used the Hypatia Catalog as dataset and selected the namely, volatiles, lithophiles, siderophiles, and Fe features. Malanchev et al. [129] deals with the recommendation of anomalies in the dataset Zwicky Transient Facility (ZTF DR3), which astronomers then study. The feedback of the astronomers helps in the development of better recommendation systems. In [191], the goal is to attribute labels to Astronomical images in a dataset of images from the MegaCam instrument mounted on the Canada-France-Hawaii Telescope. Although this work is entitled "An astronomical image content-based recommendation system using combined deep learning models in a fully unsupervised mode", no RS is tested, and they only mention that the labelled dataset may be used for recommendation systems.

One of the biggest challenges of the recommendation systems in scientific fields is the lack of available datasets for developing, testing and evaluating the recommendation algorithms. From Table 2.4, 51% of the datasets used in the studies are not available. Most of them are private and protected, especially those involving health and patients. And even those available do not allow the creation of RS which recommend scientific items, such as a gene or a chemical compounds, to researchers, since the format of these datasets is not the standard $<$user,item,rating $>$.

### 2.4.3 State-of-the-art of ontology recommender systems

In RS, ontologies are used in a particular task not represented in the previous sections, i.e., ontologies for NER. This Section presents state-of-the-art studies related to ontologies and RS in the NER context. Table 2.5 shows studies that use NER and ontologies for improving recommendations, providing information about the field of study, the type of RS, the role of NER/NEL, the tool of NER/NEL, and the ontologies applied.

Table 2.5: Background studies about the use of NER and NEL in recommender systems.

| Article | year | Field | RS type | Role of NER/NEL | NER/NEL Tool | Ontologies |
|---|---|---|---|---|---|---|
| Qi and Dong [156] | 2011 | Videos | - | To extract content information from videos | - | - |
| Abel et al. [12] | 2011 | News | CB | NER is used in tweets and external sources of news articles for creating better users' profiles, improving the recommendations | - | - |
| Musto et al. [137] | 2014 | Movies | CB | To identify most relevant context found in text related to the movies, to create users' and items' profiles | DBpedia Spotlight, Wikipedia Mine, TAGME | DBpedia, Wikipedia |

( To be continued)

| Article | year | Field | RS type | Role of NER/NEL | NER/NEL Tool | Ontologies |
|---|---|---|---|---|---|---|
| Basile et al. [32] | 2014 | Books | CB | To identify most relevant context found in text related to the books, to create users' and items' profiles | TAGME | DBpedia |
| Domingues et al. [59] | 2015 | Agro-business web pages | CF | To extract entities from web pages to enrich the dataset | - | - |
| Manzato et al. [130] | 2016 | Agro-business web pages, movies | CF | To extract entities from web pages to enrich the dataset | REMBRANDT, Stanford NER | Wikipedia |
| Eftimov et al. [61] | 2017 | Dietary-related web pages, scientific text | - | To extract dietary concepts and recommendations from scientific sources | drNER | - |
| Limongelli et al. [119] | 2017 | Teaching resources | - | To extract and link entities in the transcript of educational resources | Dandelion NER | DBpedia |

( To be continued)

| Article | year | Field | RS type | Role of NER/NEL | NER/NEL Tool | Ontologies |
|---|---|---|---|---|---|---|
| Iovine et al. [92] | 2020 | Movies | CB | To find relevant entities mentioned in the user sentence in order to improve a dialog manager | - | Wikidata |

The fields in Table 2.5 varies, but all have in common the existence of text. Most of the studies use a CB recommendation approach, and the NER and the ontologies are used for extracting information from the text to improve the recommendations. DBpedia and Wikipedia are the most used ontologies. Some articles in Table 2.5 without RS type defined did not test RS, only mentioned that the datasets improved with NER may be used for RS.

The following studies do not use NER, but they somehow use ontologies. [116] created a RS for recommending English collections of books in a library. The authors developed PORE, a personal ontology RS, which consists of a personal ontology for each user and then applying a CF method. [177] also used an ontology for creating users' profiles for the domain of books. They calculated the similarity, not between the users' ratings, but based on the interest scores derived from the ontology. [172] developed a Trust–Semantic Fusion approach, tested on movies and Yahoo! datasets. Their approach incorporates semantic knowledge to the items' primary information, using knowledge from the ontologies.

[145] presented a solution for the top@k recommendations (list of size k with the most relevant items for a user, predicted by the recommendation algorithm) specifically for implicit feedback data. The authors developed the Spank - semantic path-based ranking. They extracted path-based features of the items from DBpedia and used L2R algorithms to get the rank of the most relevant items. They tested the method on music and movies domains. [22] developed a new semantic similarity measure, the Inferential Ontology-based Semantic Similarity. The new measure improved the results of a user-based CF approach based on the tourism domain tests. Most recently, [139] developed a hybrid RS tested on the movies domain. The method used Single Value Decomposition for dimensionality reduction for the item and user-based CF and ontologies for item-based semantic similarity, improving the CF results. They do not deal with implicit data.

Ontologies and other external knowledge sources have a great potential in RS. We can see the growing use of KG in state-of-the-art RS. Nevertheless, this increase is not noticed in RS for scientific fields.

In the next chapters, this thesis will tackle some of the challenges identified in this chapter: the lack of open-access datasets for recommending scientific items, the use of ontologies in CB RS and how they may be used to enhance CF approaches, the use of sequence-aware RS for recommending scientific entities, and how NER can improve the recommendations.

# 3

# Using research literature to generate datasets of implicit feedback for recommending scientific items

This Chapter intends to answer the Research Question 1: May the use of research literature mitigate the lack of recommendation dataset for developing, testing and evaluating recommendation algorithms in scientific fields? and it corresponds to the paper: *Barros, Marcia; Moitinho, André; Couto, Francisco M; Using Research Literature to Generate Datasets of Implicit Feedback for Recommending Scientific Items, IEEE Access, 7, 176668-176680, 2019, IEEE.*

In an age of information overload, we are faced with seemingly endless options from which a small number of choices must be made. For applications such as search engines and online stores, Recommender Systems have long become the key tool for assisting users in their choices. Interestingly, the use of Recommender Systems for recommending scientific items remains a rarity. One difficulty is that the development of such systems depends on the availability of adequate datasets of users' feedback. While there are several datasets available with the ratings of the users for books, music, or films, there is a lack of similar datasets for scientific fields, such as Astronomy and Life and Health Sciences. To address this issue, we propose a methodology that explores scientific literature for generating utility matrices of implicit feedback. The proposed methodology consists in identifying a list of items, finding research articles related to them, extracting the authors from each article, and

finally creating a dataset where users are unique authors from the collected articles, and the rating values are the number of articles a unique author wrote about an item. Considering that literature is available for every scientific field, the methodology is in principle applicable to Recommender Systems in any scientific field. The methodology, which we call LIBRETTI (LIterature Based RecommEndaTion of scienTific Items), was assessed in two distinct study cases, Astronomy and Chemistry. Several evaluation metrics for the datasets generated with LIBRETTI were compared to those derived from other available datasets using the same set of recommender algorithms. The results were found to be similar, which provides a solid indication that LIBRETTI is a promising approach for generating datasets of implicit feedback for recommending scientific items.

## 3.1 Introduction

In the last years, scientific literature has increased in size and complexity [42]. Scientific literature has several applications and purposes, but the main goal is to disseminate the work and the discoveries of researchers. Recommender Systems (RS) have been a useful help to that end, by improving the discoverability of research articles.

The goal of our article is to provide a methodology for generating datasets of implicit feedback, suitable for evaluating recommender algorithms in scientific areas, by going beyond the recommendation of topics and articles, and support the recommendation of scientific items. For the purposes of this work, we define **scientific item** as an entity belonging to the universe, that may be modeled, characterized by multiple features using a computational representation, and an object of research. Some examples of scientific items are genes, phenotypes, chemical entities, plants, diseases, stars, and groups of stars, such as Open Clusters and Galaxies.

RS are software tools that provide suggestions for items that are presumably of interest to a particular user [164], which have been used in the recommendation of a wide range of products, for example, movies, books, research articles, or e-commerce [34, 124, 178]. Some well-known platforms integrating RS are GroupLens[1], including MovieLens[2], Ama-

[1] http://grouplens.org
[2] http://grouplens.org/datasets/movielens/

zon[1], Netflix[2], and Google News[3]. Due to the wide applicability of RS, there has been a progressive interest in the research of new recommendation methods and algorithms. In the beginning the approaches were mostly based in similarity metrics, but now they evolved to machine learning and deep learning techniques [15, 33, 34, 38, 44, 124, 151, 153, 184, 194].

Recommender algorithms try to predict the interest of the users in each item/product, mostly based on information from their past behaviour. Explicit or implicit feedback from the users may provide this information. Explicit feedback means that the users wittingly indicate if they liked or not some item, for example, by rating an item in a five stars scale. On the contrary, implicit feedback is extracted from the activities of the users, for example, information about what items a user clicked on or purchased. Explicit or implicit information about the preferences of the users is the foundation for RS, allowing the creation of user/item ratings matrices. Depending on the approach, RS may be divided into Collaborative-filtering (CF), when using the similarity between the ratings of the users to provide the recommendations, and Content-based (CB), when using the similarity between the characteristics of the items, and hybrid, a combination of both CF and CB [25]. CF algorithms may be divided into two methods, memory-based and model-based [184]. Memory-based methods compare users patterns of ratings by calculating the similarity between the rows (users) or the columns (items) of the ratings matrix. Model-based methods use machine learning and data mining to predict the ratings, filling the user/item ratings matrix blank spaces. One of the most used Model-based method is matrix factorization, a method which leverages all row and column correlations in one shot to estimate the entire data matrix [106]. Whereas with Memory-based methods we may explain the recommendations with "similar users also liked this item", with Model-based methods it is not always simple to identify the reason why we are recommending an item.

Despite the dissemination of RS in many fields, for example, movies, music, and e-commerce, they are not being widely used in Science. The main reason is that it is not easy to gather information about the preferences of the users/researchers about an item/topic. Offline evaluation methods [173] for recommender algorithms require a dataset with information about the past interests of the users to compare the ratings that the recommender algorithms

[1] `http://www.amazon.com`
[2] `http://www.netflix.com`
[3] `http://news.google.com`

predicted with the real ratings. Most of the platforms holding log files about the users have privacy restrictions, keeping these files private and protected.

In Health Sciences there are a few recommender systems that recommended scientific items. Those that exist are mainly focused either on the recommendation of clinical information and research articles to health professionals or on the recommendation of health related content to patients [168, 199]. In addition, drugs, genes, diseases and their relations are also scientific items targeted by recent recommender systems studies [186, 203, 234]. Other studies focus on the recommendation of plants [218], and nutrition [69]. A common complaint in all studies is the lack of datasets for evaluating recommender systems.

Offline evaluation is suitable for measuring the accuracy of the predicted ratings, for example through Mean Absolute Error (MAE) and Root Mean Square Error (RMSE), and the accuracy of ranked lists of recommended items, for instance, through Precision (PRE), Recall (REC), F-measure (F1) and normalized Discounted Cumulative Gain (nDCG) [95]. MAE measures the difference between the value of the real rating of an item, and the value of the rating predicted by a recommender algorithm, for all $n$ items under analysis. The lower this value, the better the algorithm. For evaluating the predicted rankings the most used metrics are Precision, Recall, and F-measure. The values range between zero and one, and the algorithm is better if it achieves values closest to one. For a given number $k$ of recommended items, Precision is defined as the percentage of recommended items that are relevant for the user. The Recall is the percentage of the total relevant items for a user that has been recommended. For example, if a list of size 10 recommends 5 relevant items for a user whose total number of relevant items on that test set is 5, the Recall will be 100%, because the algorithm is recommending all the possible items the user was interested in. The F-measure is the harmonic mean of Precision and Recall, allowing the global evaluation of the recommender algorithm. The nDCG measure evaluates the quality of the raking. Higher rated items should appear first in the ranking. Offline evaluation requires the division of the dataset into a training set, used for training the system, and test set used for evaluating the system. This information will enable us to compare the rating predicted by the recommender method, with the real rating in the test set.

In most of the scientific and medical fields, evaluation datasets are unavailable, compromising the evaluation and application of RS. [143] acknowledged the problem above and proposed a solution. They created a dataset (SD4AI) suitable for testing and evaluating RS for scientific topics by scanning scientific literature for information. This dataset is about the

topic of Artificial Intelligence. It consists of 14,143 articles (the articles represent the users in a traditional RS), 18,502 topics related to Artificial Intelligence (which represent the items of a RS) and 1,389,094 ratings. The ratings are the relevance of the topic in the article. This dataset is used to recommend scientific topics and articles using a CF approach.

The goal of our work is to recommend specific items enclosed in the articles. To this end, we develop a methodology, we shall call LIBRETTI - LIterature Based RecommEndaTion of scienTific Items -, based on collecting information from research articles, which are a common artifact in all scientific fields. Our approach generates a <user,item,rating >dataset, where authors of research articles represent the users, and the scientific items they wrote about represent the items to recommend. The number of articles an author wrote about an item are the implicit ratings. These ratings represent the strength of the interest of an author for an item. The structure of the dataset is the same as in [143], however, the meaning of user, item and rating is significantly different. [143] recommend topics and articles based on topics and based on articles, whereas with LIBRETTI we are able to recommend scientific items (not topics) to real people (not articles) based on the interests of their peers.

Two interesting fields for testing our approach are Astronomy and Chemistry, because there are well defined lists of scientific items, and it is easy to find research articles related with each item using web services. In the case study of Astronomy, the list of items are open star clusters (in short, Open Clusters or OCs) [58]. The web services used are Simbad[1] [216] and SAO/NASA Astrophysics Data System (ADS)[2] [108]. Simbad is a database of astronomical objects, and ADS is a bibliographic system dedicated to Astronomy. For the case study in Chemistry, the items are Chemical Compounds (Chem) collected from the Chemical Entities of Biological Interest (ChEBI) [54]. This database also includes information about the articles related to each entity, providing the PubMed IDs of these articles. PubMed is a biomedical bibliographic system, and through its web service[3] it is possible to collect the meta-data of each article (e.g.: title, authors, year).

Our methodology is suitable for any scientific field provided there is a list of scientific items and there are research articles related to each item.

The main contributions of our work are:

[1] `http://simbad.u-strasbg.fr/simbad/`
[2] `https://ui.adsabs.harvard.edu/#`
[3] `https://www.ncbi.nlm.nih.gov/home/develop/api/`

1. A new methodology (LIBRETTI) to create datasets of implicit feedback through scientific literature, helping researchers to find scientific items of interest. The methodology is designed to be general, in principle applicable to any scientific field;

2. A novel dataset in the field of Astronomy for recommending Open Clusters of stars;

3. A novel dataset in the field of Chemistry for recommending Chemical Compounds.

In this article, we describe the creation of datasets for recommender algorithms using LIBRETTI and present a well-founded study of how such datasets behave with CF algorithms. By applying known and tested recommender algorithms to our datasets, we compare our results with the results obtained for other public datasets: SD4AI and Movielens 100k (ML-100k).

We performed the evaluation of the datasets using the methods implemented in the Collaborative Filtering for Java (CF4J) library [144], which was designed for CF research experiments. Although its main function consists in testing new recommender algorithms, we used the algorithms offered in CF4J to evaluate how they perform with the datasets generated by our work.

The Python implementation of the LIBRETTI methodology for both case studies are available at `https://github.com/lasigeBioTM/cARM` and at `https://github.com/lasigeBioTM/CheRM`, as well as the full datasets used in this study.

## 3.2 Background

RS have been widely used to recommend items such as movies [75, 85, 150, 228], music [24, 228], or books [228, 231], i.e., items that in one way or another will benefit the owner of the platform where the RS is implemented. RS have also been used to recommend scientific articles. [34] surveyed more than 200 articles about RS for research literature, throughout 16 years. According to the authors, most of RS for scientific literature were applied to books, education, academic alert services, expert search, venue recommendations, educational events, patents, and even plagiarism detection. This survey concluded that CB had been the most used approach to provide the recommendations in the field of RS for research articles, with most of the RS using implicit ratings due to the lack of explicit ratings. However, the survey

does not present any work whose goal was to recommend scientific items besides documents, neither the use of authors as users of a RS.

In scientific fields, the use of RS is spreading. Table 3.1 shows in greater detail important research studies from the biomedical field using RS. It provides information about the field, what is being considered as users and items, the recommendation approach, if the dataset is considered public and its availability (if it is possible to download and use the dataset). A closer analysis shows us that the interest in RS have been growing in these fields, CF is the most used approach, and the most tested field is health in general. Only few of the datasets used in the research studies presented in Table 3.1 are public and available. In Biomedicine, the recommendation of Chemical Compounds does not seem to be a common practice. We have only two examples ([93, 171]). In [93], the authors use CF techniques for recommending Free-Wilson-like fragment to Chemical Compounds. The dataset is not public nor available. [171] aimed at discovering new inorganic compounds from all chemical combinations, using CB methods. The dataset is public, even though it is not a dataset with <user,item,rating >format.

Table 3.1: Background studies about the use of recommender systems in Bio-medicine, collected from Pubmed.

| Year | Article | Field | Users | Items | Approach | Pub/Not pub | Availability |
|---|---|---|---|---|---|---|---|
| 2003 | [146] | Genetics | Query of genes | Genes | CF | Public | Not available |
| 2007 | [138] | Genetics | Pathway | Genes | CF | Public | Not available |
| 2011 | [195] | Health | Clinical hematologists | Leukemia types | CF | Not public | Not available |
| 2011 | [43] | Drugs | Chemists | Reagents | CF | Not public | Not available |
| 2013 | [53] | Genetics | Bacteria and archaeal type strains | 16S rRNA gene sequences | CB | Public | Not available |
| 2014 | [217] | Health | Personalized Health Record System users | Personalized health information artifacts | CB | Not public | Not available |
| 2015 | [93] | Chemical compounds | Chemical compounds | Free-Wilson-like fragment | CF | Not public | Not available |
| 2016 | [45] | Health | Healthcare professionals | Clinical orders (e.g., labs, imaging, medications) | CF | Not public | Not available |

( To be continued)

| Year | Article | Field | Users | Items | Approach | Pub/Not pub | Availability |
|---|---|---|---|---|---|---|---|
| 2016 | [128] | Health | Healthcare professionals | Surveillance levels and scientific literature | CB | Not public | Not available |
| 2016 | [76] | Health | Patients | Clinical features | CF | Public | Available |
| 2016 | [50] | Genetics | RNA binding proteins | RNA targets | CF | Public | Available |
| 2017 | [135] | Health | Cardiac patients | Disease prediction and medical recommendations | Hybrid | Not public | Not available |
| 2017 | [41] | Health | Health consumers | Health educational websites from MedlinePlus | CB | Public | Available |
| 2017 | [74] | Health | Patients | Therapy | CF | Not public | Not available |
| 2017 | [46] | Health | Healthcare professionals | clinical orders (e.g., labs, imaging, medications) | CF | Not public | Not available |
| 2017 | [225] | Nutrition | Patients | Meals | CB | Not public | Not available |
| 2018 | [203] | Drugs | Drug | Disease | CF | Public | Not available |

( To be continued)

| Year | Article | Field | Users | Items | Approach | Pub/Not pub | Availability |
|---|---|---|---|---|---|---|---|
| 2018 | [171] | Chemical compounds | Inorganic compounds | Chemical relevant compositions | CB | Public | Available |
| 2018 | [186] | Drugs | Cell-lines/patients | Drug responses | CF | Public | Available |
| 2018 | [155] | Health | Patients with gestational diabetes mellitus | Blood glucose control | CF | Not public | Not available |
| 2018 | [16] | Nutrition | Patients | Nutritional advises | Hybrid | Not public | Not available |
| 2018 | [99] | Health | Patients | Treatments | CF | Public | Available |
| 2018 | [197] | Health | Patients | Insulin | CB | Public | Not available |

In another scientific field, Astronomy, there are recent studies with the goal of recommending research articles. For example, ADS implemented on its improved platform a service that recommends articles related to the one the user is currently reading, however they do not provide information about the recommender algorithms used [13]. [100] is another example of a system recommending astronomical articles. The author developed a tool that finds similar articles based only on text content from an input article (CB algorithm). The dataset used to develop the tool was collected from ArXiv[1]. The author argues that this tool works robustly, finding relevant articles that are not discovered by other platforms via citations, references or suggestions from ADS. However, they do not provide any quantitative evaluation measure for the system, providing only isolated examples, and without information about the ratings of the users. [133] implemented a different approach, by recommending opinions of other users, instead of an item/object. The authors developed a RS for astronomical observatories, that when a user introduces a query related to an instrument, the system recommends logs written by other researchers, providing positive and negative feedback. Reporting the negative feedback allows that new researches do not make the same mistakes as others. To test the system they used an open source logbook data from the Laser Interferometric Gravitational Observatory (LIGO). The performance of the system was tested using six months of logbooks, by comparing the retrieved logbooks with actual relevant entries, with the system retrieving most of the entries correctly. Despite the promising results, the authors do not present the results using standard metrics, such as Precision and Recall, and neither provide a baseline for comparison, for example, how a random recommender would perform in the test set.

More recently, [143] approached the lack of evaluation datasets for recommender algorithms in Science using a solution based on scientific literature. Their approach consists in extracting the main research topics from a dataset of articles, creating a dataset of <article, topic, cardinality>, where the cardinality is the weight of the topic in the article. This dataset is equivalent to a dataset of <user,item,rating>. The goal is to recommend topics related to the articles, and articles related to each topic. One of the contributions of that work was an evaluation dataset in the field of Artificial Intelligence (SD4AI).

Our proposal goes a step further in the RS field, mitigating the lack of datasets. Unlike previous works, LIBRETTI recommends not the research articles themselves, but the objects and items mentioned in the articles, such as clusters of stars, Chemical Compounds, diseases.

[1] https://arxiv.org/

The set of items depends on the scientific field, but as long as they are mentioned or linked to scientific articles our methodology can deal with them. Besides the methodology presented, we also generated datasets for recommending Open Clusters of Stars, and Chemical Compounds.

## 3.3 Methodology

The general view of our methodology, LIBRETTI, for creating datasets for recommending scientific items is represented in Figure 3.1. The pipeline is as follows:

(i) Identification of a list of scientific items by experts indication;

(ii) Identification of a corpus of research articles related to each item. This may be achieved by using Named Entity Recognition (NER) to identify the items in the articles, or by using external sources of knowledge, such as Pubmed or ADS, where there is already structured information linking the item to the article;

(iii) Extraction of the authors from each article;

(iv) Generation of the <user,item,rating >dataset. The users are unique authors from the articles, and the rating values are the number of articles a unique author wrote about an item;

(v) Evaluation of recommender algorithms using the dataset.

This methodology can be employed in any scientific field with well-defined items, and a corpus where they are mentioned. The next section describes the consolidation of the methodology in Astronomy and Chemistry.

### 3.3.1 Study cases

For testing LIBRETTI we used information from two fields: Astronomy and Chemistry. The consolidation of the methodology for each field is described in the next sections.

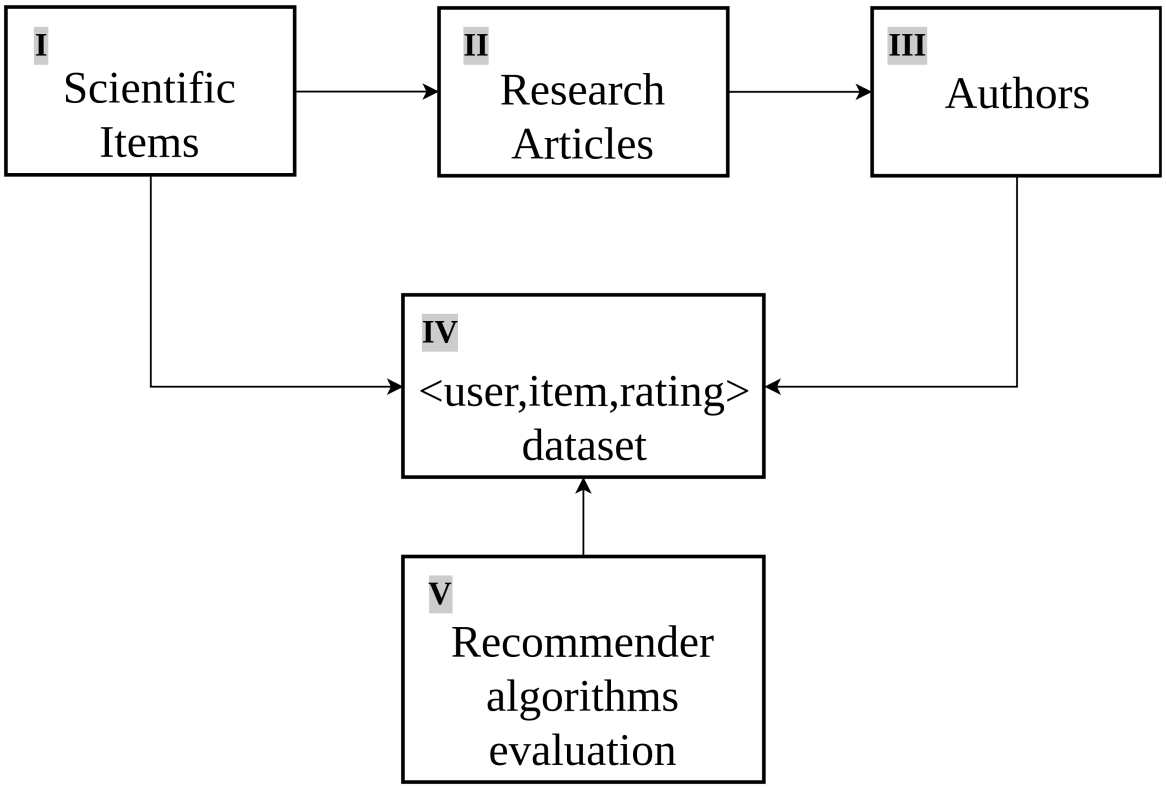


Figure 3.1: General view of the methodology LIBRETTI for creating an evaluation dataset for scientific fields using the scientific literature to extract the implicit ratings.

#### 3.3.1.1 Astronomy

For the case study using astronomical data, we selected a list of objects from a Catalogue of Open Clusters [58], with 2166 OCs and 13 features. OCs are assortments of stars formed from the same molecular cloud and with approximately the same age. Some attributes of these OCs are the position (galactic latitude and longitude), Diameter, Distance, Age, and Name.

To achieve a <user,item,rating >dataset, where users are authors of scientific research articles and the items are OCs, we followed the steps described bellow (see Figure 3.2):

1. For each cluster attribute “Name”, we searched the unique Simbad ID (unique identifier used by Simbad for each object);

2. Through Simbad ID, using ADS API[1], we searched all the articles for each cluster,

[1] `https://github.com/adsabs/adsabs-dev-api`

between the years of 1998 and May 2018;

3. For each paper, we extracted the authors, title, year, DOI and bibcode (unique identifier of an article);

4. For each author, we extracted the Name, Short Name, and Affiliation;

5. Next, we identified the unique authors;

6. Finally, we counted how many articles each unique author wrote about each Open Cluster of our list;

7. In this step we used the recommender algorithms provided by the CF4J library with the dataset created in the previous step to access the accuracy of the predicted ratings and accuracy of the given recommendations.

Step 1 required text processing to correct the names of 649 clusters from the catalogue because they were not suitable for searching on Simbad, i.e., searching the clusters by name was not retrieving any Simbad ID. In this regard, it was necessary to identify the non-matching names (usually due to using an alternative designations) and to correct the spelling. That was done by gathering all names that retrieved null in the first search in step 1, and by finding patterns in the first part of the name. For example, we found 107 names beginning with ASCC + a number. We corrected all these 107 names to [KPR2005] + a number. This happens because the ambiguity of the names: some clusters may have more than one name and not all synonyms are in Simbad. For step 2, before storing the information of the article (authors, title, year, DOI and bibcode), the methodology searches the database for similar bibcodes. If the bibcode already exists, the article is not introduced in the database, storing only the information that this article is also related to the OC under analysis. Step 5 identified the unique authors by finding all authors with the same ShortName and by considering this ShortName as a unique author/user. Step 6 created the <user,item,rating >dataset by counting how many articles a unique author wrote about each OC. The result of this step was a dataset for recommender algorithms for Astronomical OCs (Astronomical Ratings Matrix - ARM).

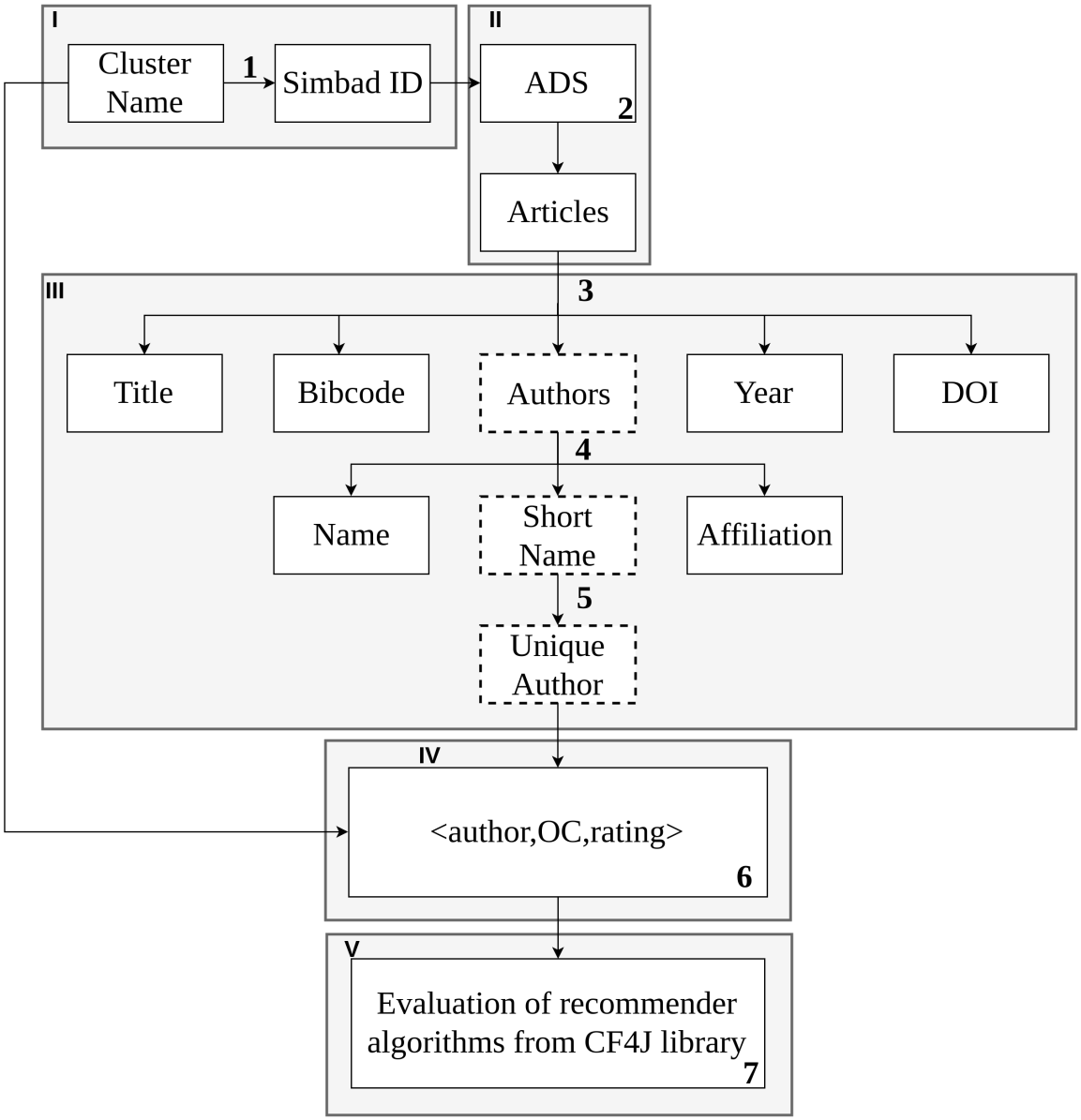


Figure 3.2: Specification of the general methodology described in Figure 3.1 for a case study in Astronomy, using as scientific items Open Clusters of Stars (OCs).

#### 3.3.1.2 Chemistry

For the case study in Chemistry, the items are Chemical Compounds extracted from ChEBI. Figure 3.3 shows the steps followed for creating a dataset for recommending Chemical Compounds:

1. From the ChEBI database, we selected all the compounds with 3 stars. For each ChEBI ID, we extracted the PubMed IDs for the articles that are identified in ChEBI as related to that compound;

2. For each PubMed ID, we extracted the information for each article throught PubMed API[1];

3. For each article, we extracted the authors, title, year and DOI;

4. For each author, we extracted the Name. The steps 5, 6, and 7 are the same as in the Astronomical case study, which allows us to create CheRM - ChEBI Ratings Matrix, a dataset for the recommendation of Chemical Compounds.

The correspondence between the general methodology (Figure 3.1) and its application to the study cases of Astronomy (Figure 3.2) and Chemistry (Figure 3.3) is I - 1; II - 2; III - 3, 4, 5; IV - 6; V - 7.

Besides the full ARM dataset and CheRM, we created a subset of ARM and a subset of CheRM by removing all the users with less than 20 rated items (ARM-20 and CheRM-20), to mimic Movielens datasets, where users are only included if they have 20 or more rated items [80]. For these study cases, there was no need to apply NER or any elaborated text-mining techniques since we already have external sources of knowledge with structured information that link the items and the articles. However, in the future, we intend to use these techniques to extract the items and information about them directly from the text of scientific articles.

### 3.3.2 Evaluation setup

[1] https://www.ncbi.nlm.nih.gov/home/develop/api/

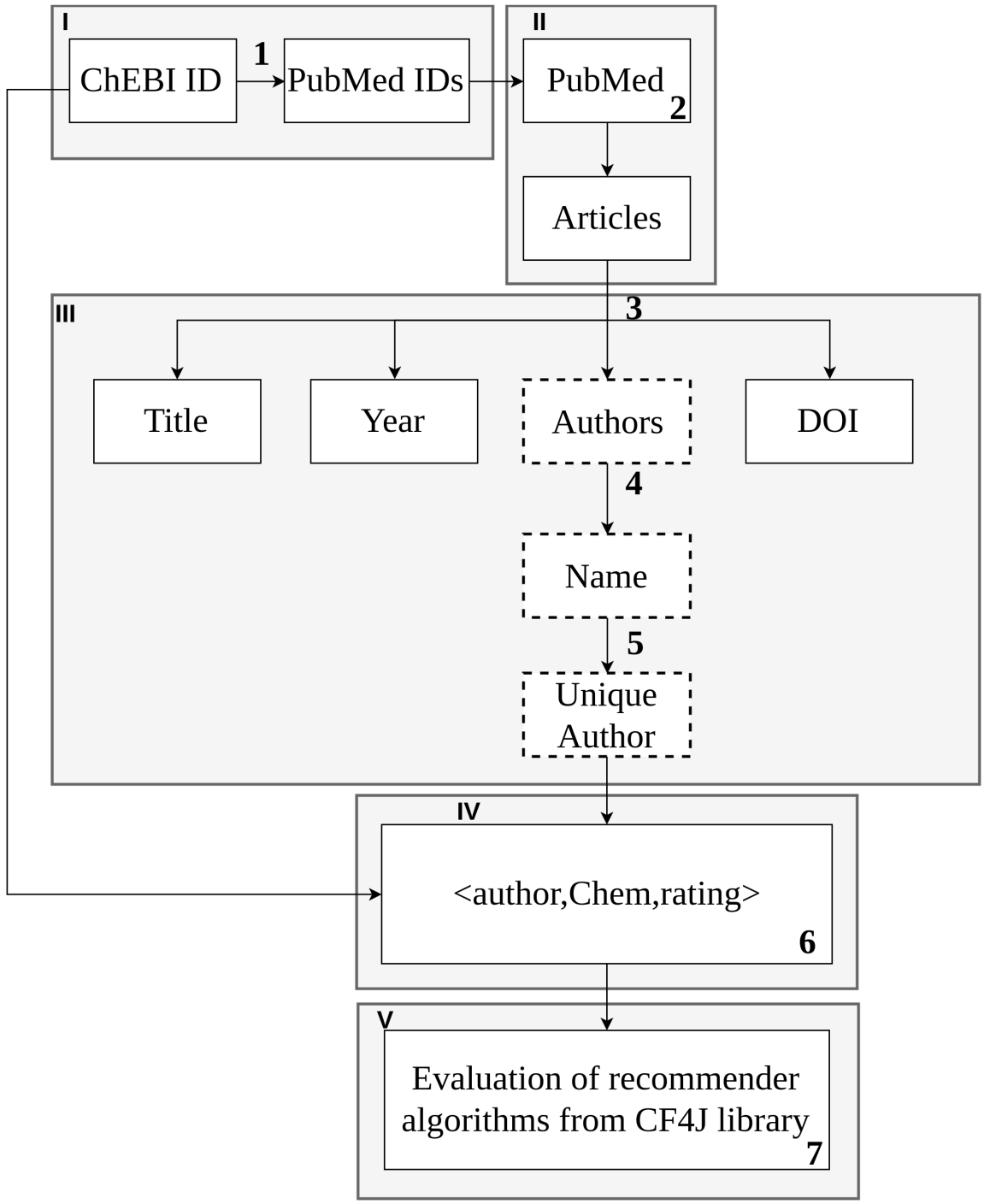


Figure 3.3: Specification of the general methodology described in Figure 3.1 for a case study in Chemistry, using as scientific items Chemical Compounds.

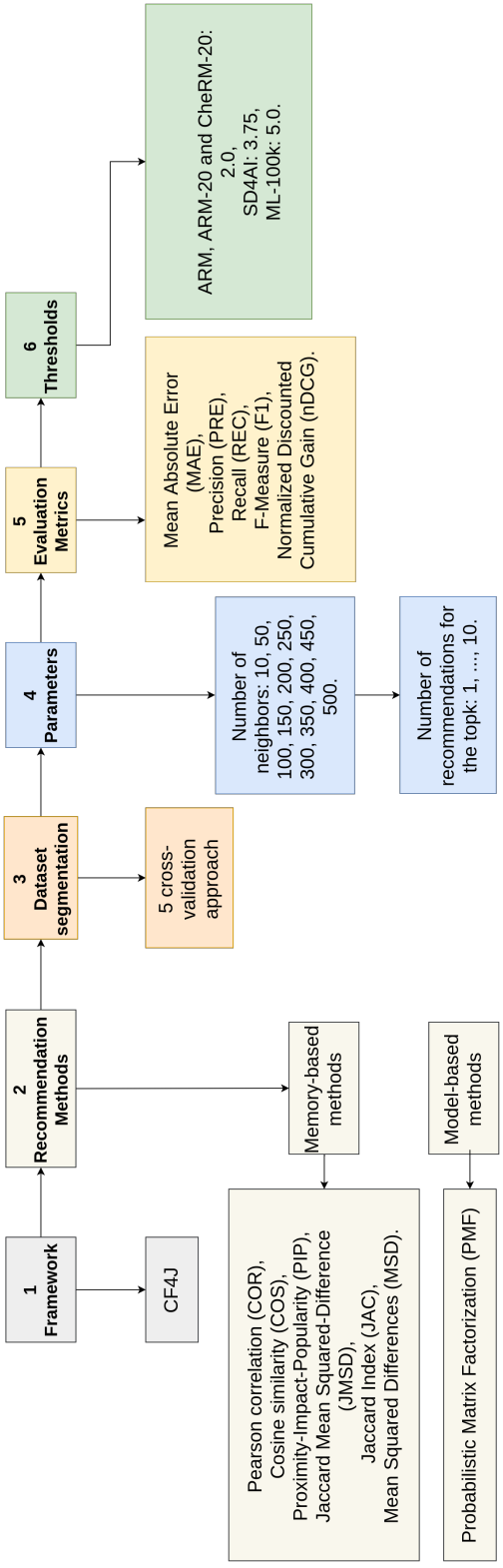


Figure 3.4: Evaluation setup.

For testing if the ARM and CheRM datasets built with LIBRETTI (see Figure 3.2 and Figure 3.3) are suitable for recommending scientific items (Open Clusters and Chemical Compounds, respectively), we followed the setup described below. We applied the same setup to other datasets, namely SD4AI [143] and the dataset from Movielens with 100k ratings (ML-100k) [80]. By following the next steps (Figure 3.4), this study is entirely replicable.

1. Selection of the evaluation framework. Several libraries exist for evaluating recommender algorithms such as LensKit [62], CF4J [144], and Mahout [126]. In this work we adopt CF4J for the evaluation of our dataset for its simplicity of use and for providing well tested recommender algorithms. CF4J also allows to directly compare our results with the results obtained in [143].

2. Selection of the recommendation methods. CF4J provides a wide range of CF recommender methods, from both memory-based and model-based methods. For this work we selected a k-nearest neighbors algorithm (a memory-based method), with the following similarity metrics: Pearson correlation (COR), Cosine similarity (COS), Proximity-Impact-Popularity (PIP), Jaccard Mean Squared-Difference (JMSD) [39], Jaccard Index (JAC), Mean Squared Differences (MSD). For model-based method, we selected a matrix factorization algorithm, the Probabilistic Matrix Factorization (PMF). With these methods we achieve a wide representation of CF algorithms.

3. Segmentation of the dataset for training and testing. In this step we selected a 5 cross-validation approach (20% for the test set and 80% for the training set).

4. Selection of the cross-validation parameters:

   (a) Number of neighbors for Memory-based methods: 10, 50, 100, 150, 200, 250, 300, 350, 400, 450, 500;

   (b) Number of recommendations for the top@k: 1, ..., 10.

   (c) For the PMF recommender algorithm the parameters used are described in Table 3.2. These are the optimal conditions achieved by testing different values.

5. Selection of the evaluation metrics. The algorithms were evaluated for MAE, PRE, REC, F1, and nDCG. With these metrics we evaluate the accuracy of the predicted

ratings, the relevance of the recommended items, and the quality of the recommended rankings.

6. Selection of thresholds (minimal rating value for considering a recommended item as relevant for the user, used in the calculation of the Precision, recall and f-measure) for the different datasets being tested. ARM, ARM-20, CheRM-20: threshold 2.0; SD4AI: threshold 3.75; Ml-100k: threshold 5.0.

Table 3.2: Parameters used in the PMF algorithm for the ML-100k, ARM-20, CheRM-20 and SD4AI datasets.

| Dataset | Latent Factor | Iterations |
|---|---|---|
| ML-100k | 1 | 50 |
| ARM-20 | 3 | 50 |
| CheRM-20 | 1 | 150 |
| SD4AI | 16 | 150 |

## 3.4 Results

In this section we describe the results obtained from the application of LIBRETTI to the Astronomy and Chemistry use cases, and the performance of the algorithms in the different datasets.

### 3.4.1 Dataset Description

Following LIBRETTI (Figure 3.1) applied to the astronomical case study described in Section 3.3.1 (Figure 3.2), we created a database with 2,166 items, 12,378 articles, and 83,208 authors, resulting in 17,006 unique authors, when grouped by equal ShortName. From the 2,166 items, 64 were excluded because no Simbad ID was found. The dataset created from our database has a size of 17,006 rows $\times$ 2,102 columns, with 179,269 ratings, which means that our user/item ratings matrix has a level of sparsity of 99.5%. The sparsity level matches the sparsity levels of rating matrices presented by other studies [143, 148, 205]. For the Chemistry case study, we have 22,307 Chemical Compounds (with distinct ChEBI

ID), 66,655 articles and 345,494 authors. The final dataset of <Author,Chem,Rating>has 22,299 Chemical Compounds, 193,106 unique authors and 456,681 ratings.

Table 3.3 shows the dimensions and statistics about the datasets of ARM, ARM-20, CheRM, CheRM-20 and also for SD4AI and ML-100k.

Table 3.3: Dimensions of the datasets evaluated in this study for **nUsers** (number of users), **nItems** (number of items), **nRat** (number of ratings), **minRat** (minimal rating), **maxRat** (maximum rating), **nThreshold** (number of ratings greater or equal to the defined threshold), **sparsity**, **mean**, **SD** (Standard deviation), **mode**, and **median**.

| Dataset | nUsers | nItems | nRatings | minRat | maxRat | nThreshold | Sparsity | Mean | SD | Mode | Median |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **cheRM** | 193106 | 22299 | 456681 | 1 | 62 | 28967 | 99.98 | 1.08 | 0.45 | 1 | 1 |
| **cheRM-20** | 2193 | 16437 | 117020 | 1 | 62 | 7600 | 99.67 | 1.08 | 0.41 | 1 | 1 |
| **ARM** | 17006 | 2102 | 179269 | 1 | 89 | 49325 | 99.49 | 1.72 | 2.02 | 1 | 1 |
| **ARM-20** | 1493 | 2101 | 106104 | 1 | 89 | 34258 | 96.61 | 1.92 | 2.37 | 1 | 1 |
| **ML-100k** | 943 | 1682 | 100000 | 1 | 5 | 21201 | 93.69 | 3.52 | 1.12 | 4 | 4 |
| **SD4AI** | 14143 | 18502 | 1389094 | 1 | 160 | 216746 | 99.46 | 2.38 | 2.61 | 1 | 1.75 |

Figures 3.5 and 3.6 shows the relevant statistical information of ARM and CheRM datasets, respectively. The maximum rating value for ARM is 89 (a single author wrote 89 articles featuring a cluster), corresponding to user 14308 and item "Melotte 22" also known as the Pleiades (simbad ID:675533). For CheRM the maximum rating is 62, corresponding to user 164989 and to the item ChEBI:101096 (ethoxzolamide).

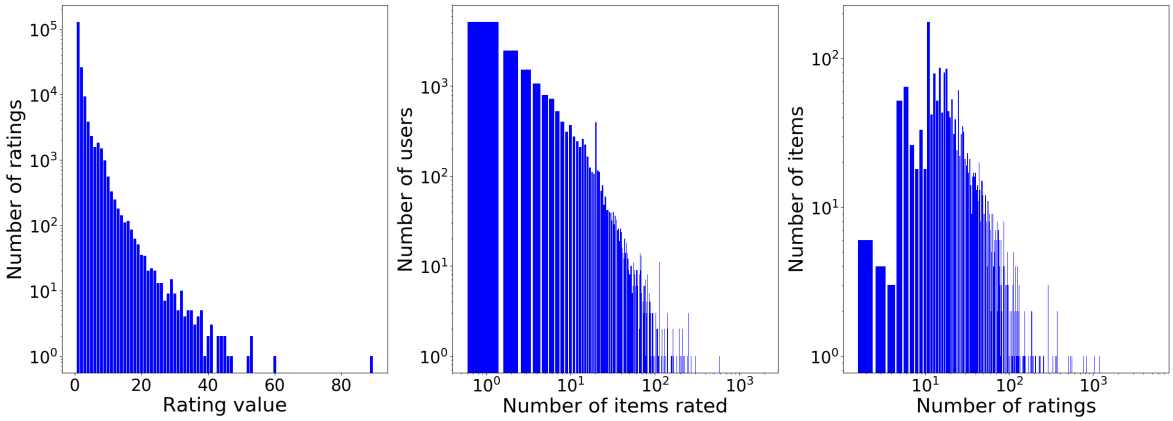


Figure 3.5: Analysis of ARM dataset. Left: Distribution of rating values; Center: Number of rated items by user; Right: Number of ratings by item.

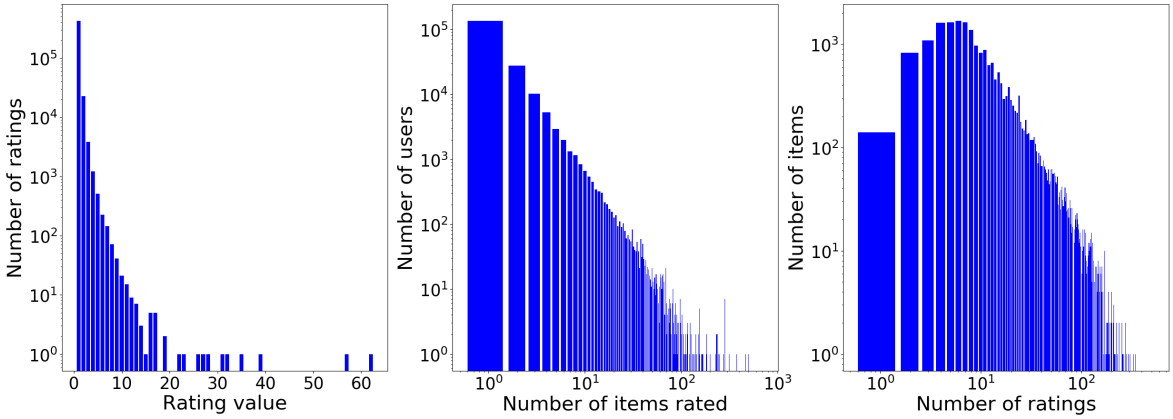


Figure 3.6: Analysis of CheRM dataset. Left: Distribution of rating values; Center: Number of rated items by user; Right: Number of ratings by item.

The distribution of the rating values by number of ratings is represented on the left graphics of Figures 3.5 and 3.6. The minimal rating for both datasets is 1, and it corresponds to 72% of the ratings for ARM and 93% for CheRM, meaning that the majority of the authors wrote only one article about the items in study.

Table 3.4: Recommender algorithms top results for each evaluation metric, for ARM, SD4AI, ARM-20, CheRM-20 and ML-100k datasets.

| | ARM | | SD4AI | | ARM-20 | | CheRM-20 | | ML-100k | |
|---|---|---|---|---|---|---|---|---|---|---|
| **M** | **Algo** | **Value** | **Algo** | **Value** | **Algo** | **Value** | **Algo** | **Value** | **Algo** | **Value** |
| MAE | JMSD | 0.593 | COR | 0.562 | JMSD | 0.903 | MSD | 0.100 | PIP | 0.754 |
| PRE | COR | 0.371 | PIP | 0.641 | JAC | 0.599 | PIP | 0.158 | JAC | 0.356 |
| REC | PIP | 0.936 | PIP | 0.793 | PIP | 0.893 | PIP | 0.609 | PIP | 0.705 |
| F1 | COS | 0.770 | PIP | 0.601 | JMSD | 0.600 | COS | 0.740 | PIP | 0.425 |
| nDCG | PIP | 0.845 | PIP | 0.769 | PIP | 0.838 | PIP | 0.836 | PIP | 0.773 |

The total number of items rated by user is represented on the center graphics of Figures 3.5 and 3.6. For example, for ARM, 5207 authors have only one item rated (cold start problem), and for CheRM this value is 136,391 authors. In our context, this means that 30% of the authors in ARM only wrote about one of the cluster of stars of our list and 70% on the authors in CheRM only wrote about one Chemical Compound of our list. The right graphics of Figures 3.5 and 3.6 show the number of ratings by item. There are no items with only one rating for ARM, with the minimal number of ratings being 2, for a total of 6 items. There are 176 items with 11 ratings each, and this is the most frequent number of ratings. The most rated item is “Melotte 22”, with ratings from 5287 users. For CheRM, there are 140 Chemical Compounds with only one rating, and the item with more ratings is CHEBI:465284 (ganciclovir) with ratings from 529 authors.

### 3.4.2 Dataset Validation

To elucidate about what is being recommended with ARM, Figure 3.7 provides an example of what the PIP algorithm recommends to user 1206 in a top 10 ranked list. Thus, for this user using PIP, the ranked list of recommended items is [175, 187, 1104, 1139, 1850, 152, 1573, 2002, 2012 and 866], which corresponds to the OCs named [Melotte 20, IC 348, IC 2602, NGC 3532, Roslund 5, NGC 1039, NGC 6494, NGC 7092, Trumpler 37, and NGC 2571], respectively. The OCs underlined are the ones correctly recommended, i.e., relevant for this user (according to the previously defined threshold of 2.0). For this user, the Precision is 0.60, and the Recall is 0.86 since this user has 6 relevant items in the test set. For CheRM, instead of OCs, we are recommending Chemical Compounds from ChEBI.

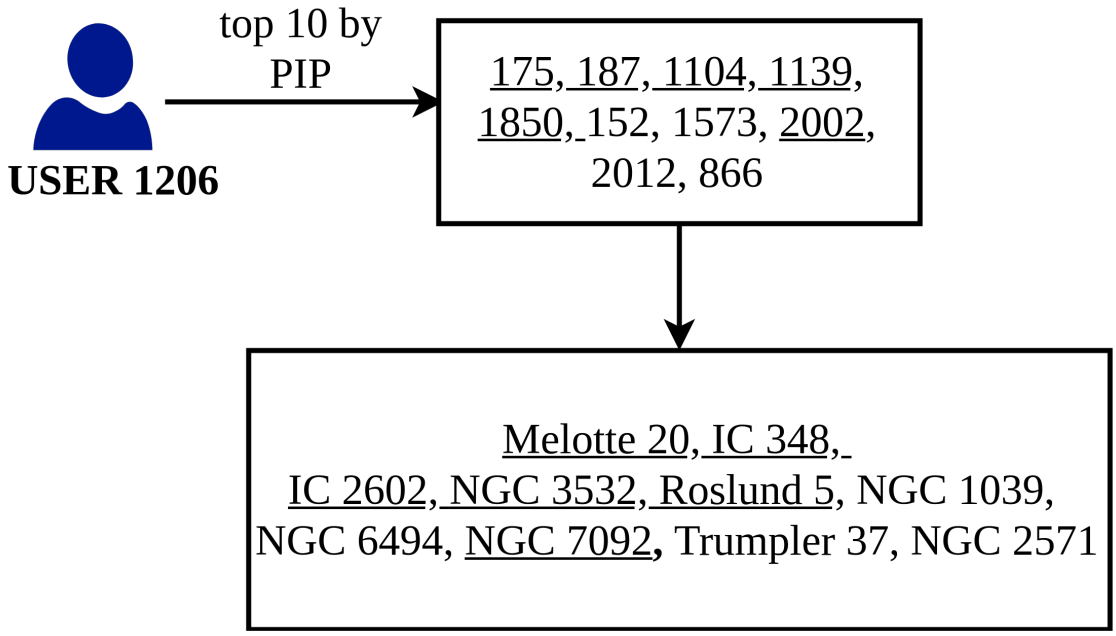


Figure 3.7: Example of the top 10 recommendations of Open Clusters to the user 1206, calculated by the PIP recommender algorithm. The OCs underlined are the ones from the top 10 that are relevant to the user 1206.

Table 3.4 shows the top results obtained for ARM, ARM-20 and CheRM-20, as well as for SD4AI and ML-100k, for the measures (**M**) MEA, PRE, REC, F1, and nDCG. The table presents the maximum value for each measure (**Value**), and the algorithm where it was obtained (**Algo**). The algorithms in question are COR, COS, PIP, JMSD, JAC, and MSD. The results of PMF are presented separately for a better comparison of Memory-based and Model-based algorithms. It was not possible to get the results for the full CheRM dataset since CF4J cannot process datasets of such large dimensions, thus we used CheRM-20 for a fairer comparison with ML-100k and ARM-20. ARM achieved better results for Recall, F-measure and nDCG than SD4AI. For ARM-20 the results of the Precision are better than for ARM. Due to its dimensions (Table 3.3), ARM-20 is more comparable to ML-100k, and its results for Precision, Recall, F-measure, and nDCG are higher. PIP is the recommender algorithm that achieved the best results for most of the evaluation measures in all datasets. The Precision for ARM is the value that presents a higher difference for the highest Precision achieved with SD4AI. MAE is similar in all datasets, however this measure is not directly comparable due to the different range of ratings values of the evaluation datasets.

Figures 3.8, 3.9 and 3.10 show in more detail the results of Precision, Recall, and nDCG, respectively, obtained in the different datasets with the different algorithms. Analysing the plots, we see that the datasets present similar behavior for the same algorithms. PMF results are presented in Table 3.5. ARM-20 benefits from this algorithm only for Recall and nDCG. For ML-100k and SD4AI, PMF is the recommender algorithm with the best results. Thus, based on these results, we can say that using CF4J, Memory-based algorithms work better than Model-based algorithms for the ARM dataset.

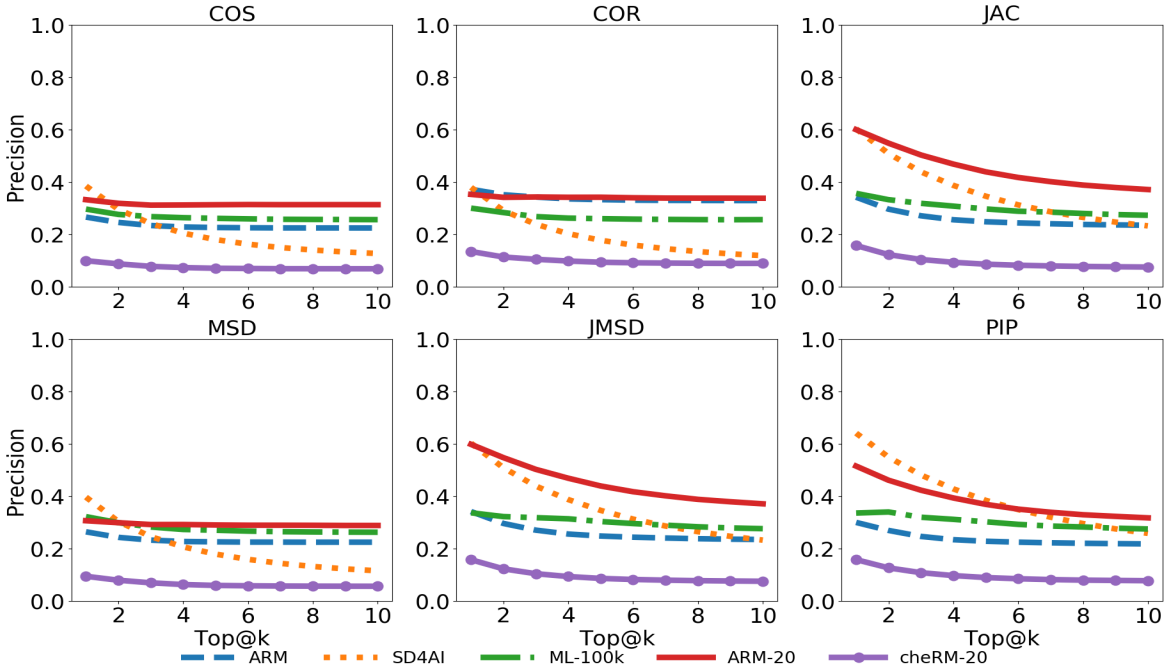


Figure 3.8: Precision at k (k between 1 and 10), for Pearson Correlation, Cosine Similarity, Jaccard Index, Jaccard Mean Squared Difference, Mean Squared Difference, and Proximity Impact Popularity, for ARM, ARM-20, CheRM, ML-100k and SD4AI datasets

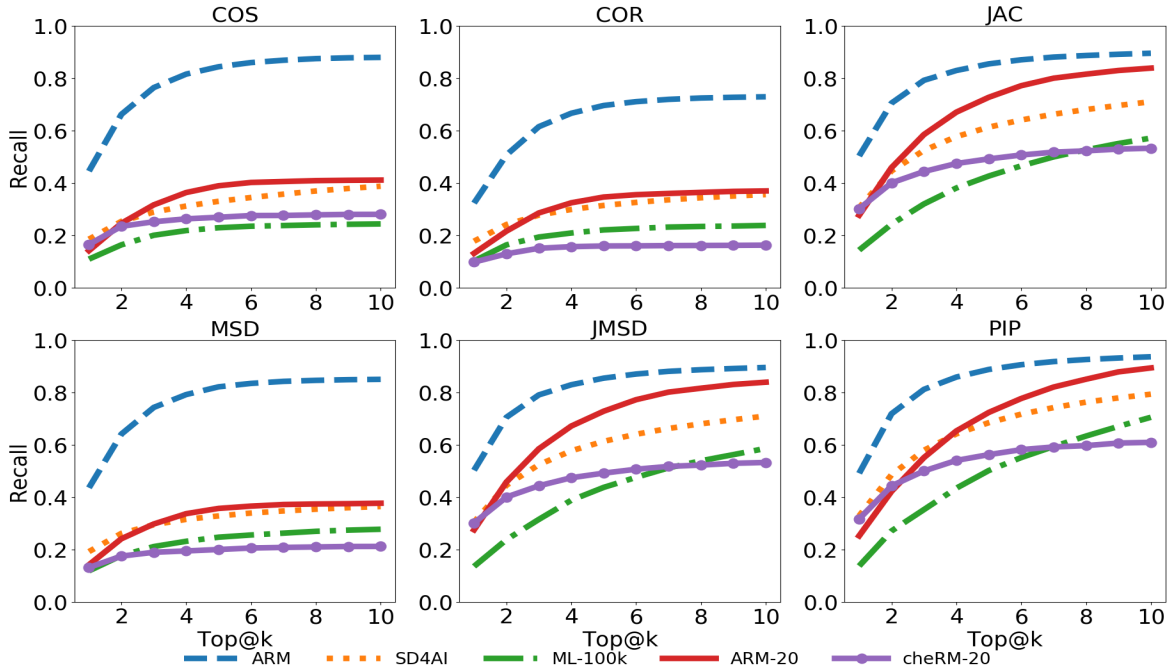


Figure 3.9: Recall at k (k between 1 and 10), for Pearson Correlation, Cosine Similarity, Jaccard Index, Jaccard Mean Squared Difference, Mean Squared Difference, and Proximity Impact Popularity, for ARM, ARM-20, CheRM, ML-100k and SD4AI datasets

Table 3.5: Results for the PMF recommender algorithm for the datasets ARM-20, CheRM-20, SD4AI, and ML-100k.

| **Measure** | **ARM-20** | **CheRM-20** | **ML-100k** | **SD4AI** |
|---|---|---|---|---|
| MAE | 0.906 | 0.200 | 0.756 | 0.686 |
| PRE | 0.501 | 0.149 | 0.479 | 0.669 |
| REC | 0.923 | 0.893 | 0.774 | 0.886 |
| F1 | 0.569 | 0.726 | 0.466 | 0.609 |
| nDCG | 0.886 | 0.955 | 0.832 | 0.815 |

The results for each dataset are not directly comparable since they use similar but not equal settings (e.g.: minimum and maximum rating, thresholds), however they provide sound indication of LIBRETTI effectiveness.

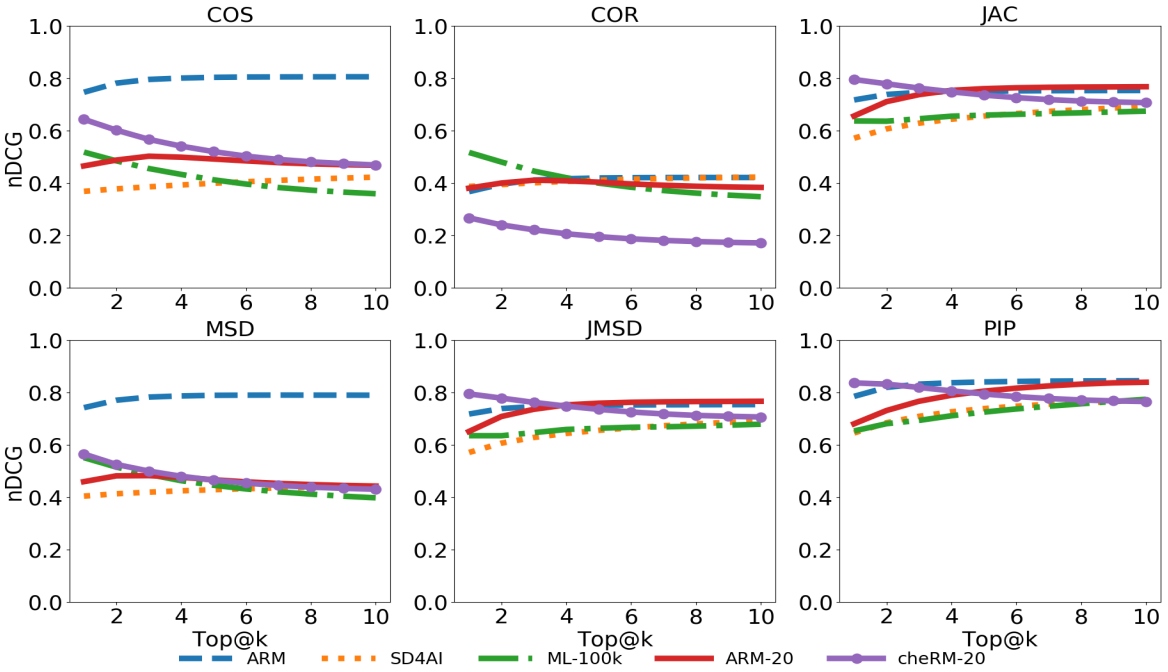


Figure 3.10: nDCG at k (k between 1 and 10), for Pearson Correlation, Cosine Similarity, Jaccard Index, Jaccard Mean Squared Difference, Mean Squared Difference, and Proximity Impact Popularity, for ARM, ARM-20, CheRM, ML-100k and SD4AI datasets

## 3.5 Discussion

The lack of datasets for deploying or evaluating recommender algorithms for scientific data exploration is a major drawback delaying their use and development in this area. The proposed methodology, LIBRETTI, is a solution for the lack of ratings, taking advantage of the comprehensive list of scientific publications available for all research areas.

From the results presented in Section 3.4.1 we can say that ARM, ARM-20 and CheRM-20 are similar to other datasets often used in the field of RS, such as movies datasets, with similar values of data sparsity. Compared with SD4AI, ARM has less items, however, we achieved almost the same number of users. This is a positive point because we will have more users to search for similarity. A disadvantage of ARM is its high percentage of users who rated only a few items. For instance, when we remove the users who rated less than 20 items to create ARM-20, the number of users is reduced to less than 8%. In the case of CheRM-20, the number of users is reduced to 1.13% of the original dataset. This may be mitigated by using NER to extract more items from each article, items that may not be

identified in the external sources of knowledge that we used (SIMBAD, ADS and ChEBI). Like this, we will have more items rated for the same number of users. Despite that, the results with ARM-20 are strong, as may be seen in Table 3.4 and 3.5. For Precision, Recall, F-measure and nDCG, ARM-20 results are higher than the results of ML-100k. ML-100k is a dataset widely used for evaluating recommender algorithms, thus these results support our hypothesis that ARM is a viable solution in assessing recommender algorithms in scientific fields.

The results for CheRM-20, particularly for precision, are lower than the results for the other datasets in this study. This may be explained by the fact that CF4J is a framework more suitable for datasets of explicit data, where we can define a threshold for the rating, defining an item as relevant/not relevant. The datasets developed through LIBRETTI methodology are implicit and even the minimal rating, 1, is relevant. When we move the threshold to 2, we are losing, in the case of CheRM, 93% of the actually relevant ratings (see Table 3.3). Thus, for example, if the RS recommends 5 items, whose real ratings are 1, the Precision will be zero, since all the ratings are bellow the defined threshold. If we define the threshold as 1, the Precision will always be one, since CF4J only recommends items from the testset that we have a real rating. For example, if a user in the testset rated 10 items, and we want the top 5, CF4J ranks only these 10 items and recommends 5 of them. As the threshold is 1, the Precision is 1 because all items are relevant.

An advantage of the datasets created with LIBRETTI is that they may be used as direct input data for CF platforms, mitigating the sparsity problem. The pure cold start problem of new users, which do not have any rated item, is not overcome by our datasets. However, with a few ratings we can easily find similar users. The cold start for new items is also a challenge in CF. Our dataset may help solving this problem by introducing into the recommendation platforms implicit ratings for these unrated items.

The datasets created with LIBRETTI can also be used for testing and evaluating recommender algorithms. The dataset is filled with real people (the authors of the articles), who in one moment of their research had interest for that item they mentioned. For example, if we were evaluating which is the best algorithm for recommending OCs, analysing Figures 3.8, 3.9 and 3.10, for Precision it would be JMSD, and for Recall and nDCG it would be PIP. Another advantage is that LIBRETTI is scalable, and not limited to a small number of items. The most limiting point related to the scalability of the method is the access restrictions that

may be imposed by the external sources. For example, the ADS API only allows 5000 requests per day. Another advantage of LIBRETTI is that the database creation process runs offline. Thus, it does not interfere with the retrieval of the recommendation to the user, and it is easy to keep updated, with regular crawling for new articles.

The application of LIBRETTI for creating ARM and CheRM is fully available at `https://github.com/lasigeBioTM/cARM` and `https://github.com/lasigeBioTM/CheRM`, as well as the datasets used in this study.

## 3.6 Conclusions

The main goal of this work was to provide a validated methodology for generating datasets of implicit feedback suitable for recommending scientific items using CF approaches. The proposed methodology, LIBRETTI, consists in identifying a list of items/objects, finding research articles mentioning each item, extracting the authors from each article, and finally creating a <user,item,rating >dataset where users are unique authors from the collected articles, and the rating values are the number of articles a unique author wrote about an item. We used Astronomy and Chemistry as case studies and compared the obtained datasets (ARM, ARM-20 and CheRM-20) with SD4AI and ML-100k. Considering the results obtained, we believe that LIBRETTI paves the way to a widely applicable and an effective solution for testing and evaluating the use of recommender algorithms in scientific areas and for the recommendation of not studied items for the researchers.

# 4

# Hybrid Semantic Recommender System for Chemical Compounds in Large-Scale Datasets

This Chapter answers to the Research Question 2: Does the use of semantic similarity between the Chemical Compounds calculated through ontologies for creating a Content-based (CB) algorithm improve the results of state-of-the-art collaborative-filtering algorithms for implicit feedback recommendation datasets? and it corresponds to the paper: *Barros, Marcia, Andre Moitinho, and Francisco M. Couto. "Hybrid semantic recommender system for chemical compounds in large-scale datasets." Journal of cheminformatics 13.1 (2021): 1-18.*

The large, and increasing, number of chemical compounds poses challenges to the exploration of such datasets. In this work, we propose the usage of Recommender Systems to identify compounds of interest to scientific researchers. Our approach consists of a hybrid recommender model suitable for implicit feedback datasets and focused on retrieving a ranked list according to the relevance of the items. The model integrates collaborative-filtering algorithms for implicit feedback (Alternating Least Squares and Bayesian Personalized Ranking) and a new content-based algorithm, using the semantic similarity between the chemical compounds in the ChEBI ontology. The algorithms were assessed on an implicit dataset of chemical compounds, CheRM-20, developed according to the LIBRETTI methodology, with more than 16.000 items (chemical compounds). The hybrid model was able to

improve the results of the collaborative-filtering algorithms, by more than ten percentage points in most of the assessed evaluation metrics.

## 4.1 Introduction

Chemical entities/compounds, defined as "physical entities of interest in chemistry including molecular entities, parts thereof, and chemical substance"[4], are growing in number and complexity, generating large datasets, challenging for the researchers to explore deeply. Recommender Systems (RS) may be a feasible solution for this challenge by identifying new entities to explore, for example, by suggesting entities not yet studied by the researchers based on their past investigation projects. However, the recommendation of chemical compounds of interest has not been widely explored [93, 171]. One challenge to include RS in compound databases is the lack of available datasets with the preferences of the researchers about the chemical compounds for assessing the RS. For example, it is not easy to explicitly know if a specific researcher had interest in the study of a chemical or not. More recently, alternatives have emerged with the development of datasets consisting of data collected from implicit feedback [29, 143]. These datasets do not contain the explicit interests of the users, as other famous datasets, such as Movielens [79]. Instead, this information is extracted from their activities, mostly from the scientific literature, which remains the main method for disseminating scientific work.

Datasets of explicit or implicit feedback require different recommender algorithms, especially because implicit feedback has significant downgrades, such as the lack of negative feedback and unbalanced ratio of positive vs. unobserved ratings [102, 161]. When dealing with implicit feedback datasets, the solution involves applying learning to rank (L2R) approaches. L2R consists in, given a set of items, identify in which order they should be recommended [160].

In RS, the main approaches are Collaborative-filtering (CF) and Content-Based (CB) [165]. CF uses the similarity between the ratings of the users, and CB uses the similarity between the features of the items. CF is divided into two methods, memory-based and model-based [184]. Memory-based methods deal with the recommendation problem by finding the most similar users based on the ratings of the items. If two users tend to rate the same items in the same way, they will probably like the items seen by each other. Model-based methods use machine learning and data mining for predicting the ratings or for assigning a

score to each item by filling the rating matrix blank spaces (unknown ratings). One of the most used methods is matrix factorization since it leverages all row and column correlations in one shot to estimate the entire data matrix [106]. With model-based methods, it is more difficult to explain the recommendations.

CF approaches cannot deal with new items or new users in the system, i.e., items and users without ratings (cold start problem). CB does not suffer from the cold start problem for new items since this approach only needs the features that characterize them to compare with the features of the items that the user already saw or liked. Thus, even if the new item does not have a single rating in the entire dataset, it may still be recommended. However, CB needs a list of features for the items, which varies from field to field. To deal with CF and CB challenges, we can develop hybrid RS, which are the assembling of CF and CB. One of the most common forms of creating hybrids is by a weighted technique, where the scores of the different algorithms are combined into a unique final score [17].

One of the challenges of CB approaches is related to which features to use for finding similar items. Some items have obvious features. For example, when our items are movies, the features used to find similar items may be the genre, director, and authors. In other fields, the task of finding features for the items is not that obvious. Thus, one of the tools used by CB for this purpose is ontologies [190], which provide controlled vocabularies of terms and definitions to represent the entities of a specific field of study [28, 198].

The notion of ontology is not new and has long been used for classifying and describing concepts. At the time of the rising of the semantic web, ontologies were adapted to computational reasoning and knowledge sharing since their structured format (triplets of subject, predicate and object) makes them ideal for computer processing. More recently, ontologies were adapted to the biological/biomedical domain. Some examples of well-known bio-ontologies are the Chemical Entities of Biological Interest (ChEBI) [2, 81], the Gene Ontology (GO) [9, 49], and the Disease Ontology (DO) [7, 169]. Bio-ontologies are particularly important for providing a unique identifier for biomedical entities. The name of biomedical entities may change over time, and different researchers may refer to them differently. One of the advantages of the ontologies is storing lists of these descriptors. Considering, for example, the chemical entity caffeine [3]. This entity is identified in the ontology with the primary name **caffeine**, primary ID **CHEBI:27732** and it has an extended list of synonyms:

- 1,3,7-trimethyl-2,6-dioxopurine

- 1,3,7-trimethylpurine-2,6-dione
- 1,3,7-Trimethylxanthine
- 1,3,7-trimethylxanthine
- 1-methyltheobromine
- 3,7-Dihydro-1,3,7-trimethyl-1H-purin-2,6-dion
- 7-methyltheophylline
- anhydrous caffeine
- cafeín
- caféine
- CAFFEINE
- Caffeine
- caffeine
- Coffein
- guaranine
- Koffein
- mateína
- methyltheobromine
- teína
- Thein
- theine

Thus, when a researcher is interested in scientific articles about Koffein, we can use the ontology for identifying all its synonyms and retrieve all the articles that mention them instead of just limiting the search to the given descriptor. Another significant advantage of the ontologies is that we can relate the entities through their semantic similarity, a measure based on the ontology's semantic structure. Figure 4.1 shows the knowledge graph adapted from ChEBI for the chemical compound *caffeine*. As we can see in the graph, the relations are defined based on the semantics of the entities, for example, *caffeine* **is a** *purine alkaloid*. We can use these relations to calculate how much two entities are semantically similar, for example, considering their common ancestors.

Several works have used the semantic similarity between the entities of an ontology. In [68], the authors developed a hybrid method for classifying chemical compounds based on structural and semantic similarity. This work concluded that using semantic similarity improves the classification of the chemical compounds and the best results were obtained when the weight of semantic similarity was higher than two thirds (71%) and the weight of the structural similarity less than one third (29%). More recently, [212] used the structural similarity and the ChEBI semantic similarity assembled into a hybrid for predicting compounds subtracts suitable for membrane transporters. Other studies used the semantic similarity of ChEBI entities for recognition and confirmation of chemical compounds found in research documents [73, 109]. In our work, we propose using the ontologies as a source of features that characterize the scientific items to find similar items for recommendation.

The field of RS is broad, and its approaches are applied to several domains, such as movies [202], books [193], and e-commerce [176]. In the Chemistry domain, RS have been generally used in studies related to drugs, for example, for new drugs design [43], and for finding candidate drugs for diseases [77]. [43] used RS for recommending reagents for new drugs, based on the experience of other chemists. The dataset used in this study, despite interesting, is not available. [77] applied RS techniques for recommending targets to drugs. The datasets used has the format of target-drug pairs, but it does not contain any information about the researcher choices. Most recently, [180] used RS approaches to discover new antiviral drugs, extracting compounds from ChEMBL [1], a database of molecules with drug-like properties. The dataset used has the format of compound-viral species-interaction value. The authors explain how the dataset was created, but they do not provide the dataset. Other RS applications in Chemistry may be found in [93], which describes the use of CF methods for creating possibilities for new chemical compounds. The dataset is not available. [171]

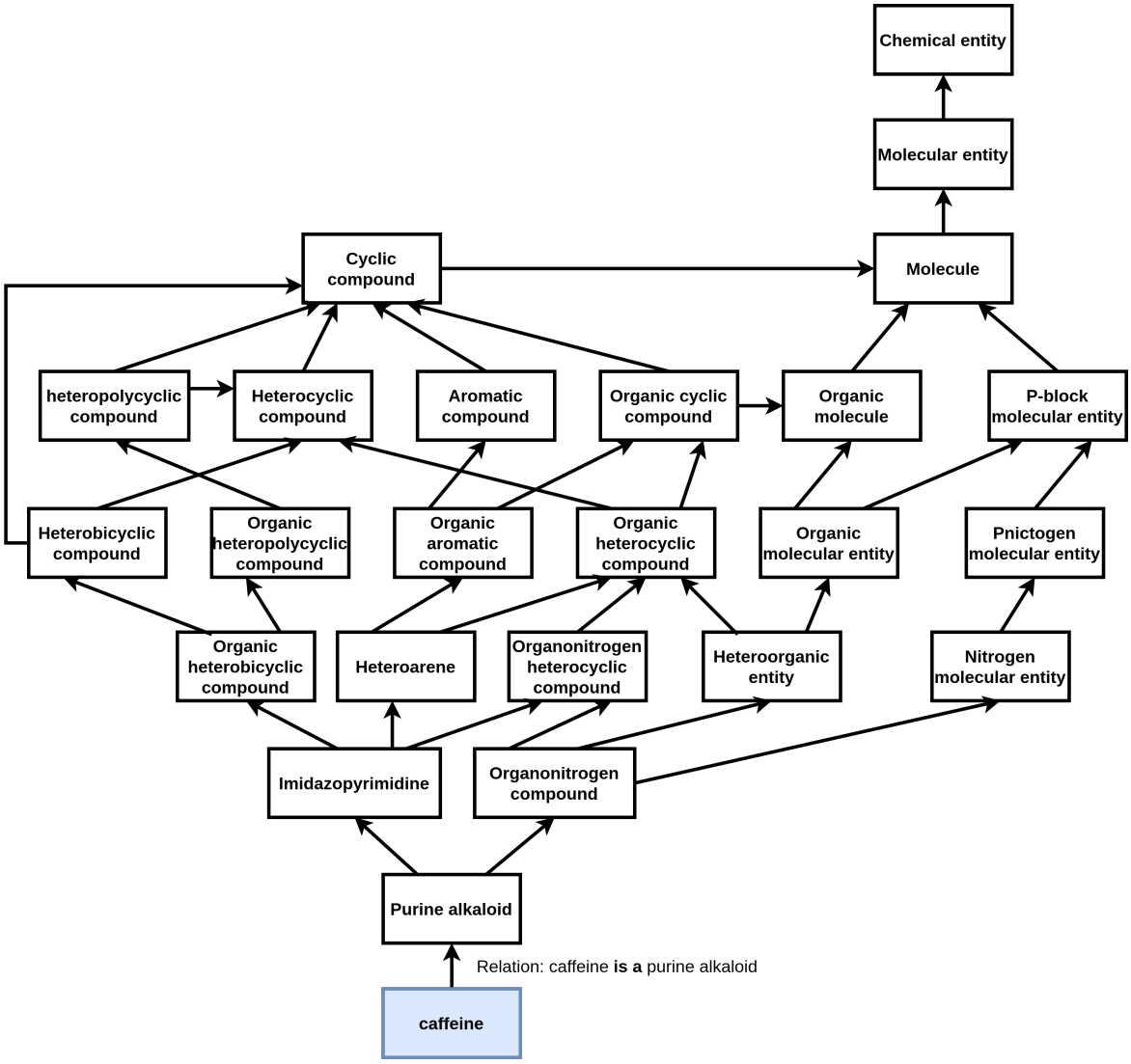


Figure 4.1: Knowledge graph for the entity caffeine, adapted from ChEBI.

uses RS techniques also for the discovery of new inorganic compounds. The authors used the features of chemical relevant compositions to predict if a certain composition is a good candidate to inorganic compound. If the system predicts a composition as being a new compound, it recommends this composition to further studies. The authors provide some additional material, but not the final dataset used in the RS. Once again, this study does not use a dataset of user, item, rating, and it does not have any information about the preferences of the researchers.

None of the previous studies reported the use of ontologies, as opposed to the studies presented below, in which the use of ontologies enhanced the CF approaches. [116] created a RS for recommending English collections of books in a library. The authors developed PORE, a personal ontology Recommender System, which consists of a personal ontology for each user and then applying a CF method. [177] also used an ontology for creating users' profiles for the domain of books. They calculated the similarity, not between the ratings of the users, but based on the interest scores derived from the ontology. [172] developed a Trust–Semantic Fusion approach, tested on movies and Yahoo! datasets. Their approach incorporates semantic knowledge to the items' primary information, using knowledge from the ontologies.

[145] presented a solution for the top@k recommendations (list of size k with the most relevant items for a user, predicted by the recommendation algorithm) specifically for implicit feedback data. The authors developed the Spank - semantic path-based ranking. They extracted path-based features of the items from DBpedia and used L2R algorithms to get the rank of the most relevant items. They tested the method on music and movies domains. [22] developed a new semantic similarity measure, the Inferential Ontology-based Semantic Similarity. The new measure improved the results of a user-based CF approach, based on tests on the tourism domain. Most recently, [139] developed a Hybrid RS tested on the movies domain. The method used Single Value Decomposition for dimensionality reduction for the item and user-based CF, and ontologies for item-based semantic similarity, improving the CF results. They do not deal with implicit data.

For datasets of implicit feedback, there are two CF algorithms which have been particularly popular, Alternating Least Squares (ALS) [90] and Bayesian Personalized Ranking (BPR) [161]. ALS is a latent factor algorithm that addresses the confidence of a user-item pair rating, which goal is to minimize the least squares error of the observed ratings by factorizing the ratings matrix in user and item matrix. ALS has the advantage of being easily parallelized. Some recent studies focused on speeding up the implementation of this algorithm [78, 115]. Another study developed a recommender system for movies based on ALS using Apache Spark [23]. BPR is also a latent factor algorithm, but it is more appropriate for ranking a list of items. BPR does not just consider the unobserved user-item pairs as zeros but also discerns the preference of a user between an observed and an unobserved rating. Several studies have been using BPR in the recommendation of items from implicit feedback datasets. [37] presented a deep neural network model based on Stack Denoising

Auto-Encoder and BPR. [239] proposed a social distance-aware BPR model for social network recommendations. [236] presented a solution for the recommendation of restaurants, based on deep learning and BPR, for multi-source datasets of implicit feedback.

Here we present a new hybrid semantic recommender model for recommending chemical compounds that uses semantic similarity and deals with implicit feedback data, of which a prototype has been presented in [30]. The system here presented is now capable of dealing with thousands of items, and the results represent an improvement over top@k in several evaluation metrics. The hybrid model has two modules, one CF and one CB. The CF module addresses the implicit feedback datasets by applying ALS or BPR, and the CB module explores the semantic similarity of the chemical compounds. The Hybrid model combines the outcomes of the CF and CB modules.

The main contributions of this work are:

- a recommender framework for recommending chemical compounds;
- a new CB semantic recommender algorithm named ONTO based on ontologies;
- a new Hybrid recommender algorithm for datasets of implicit feedback;
- a dataset with the semantic similarity between more than 16.000 chemical compounds;
- a faster semantic similarity calculation for DiShIn library.

The framework developed for this work, as well as all the data, is available at `https://github.com/lasigeBioTM/ChemRecSys`.

## 4.2 Methods

### 4.2.1 Workflow of the proposed model

In this work we propose a Hybrid recommender model, featuring two modules: CF and CB. Figure 4.2 shows the general workflow of the model.

The input data used in this model, better described in Experiments Section, has the format of <user,item,rating>. The unrated set represents the items we want to rank to provide the best recommendations in the first positions to a user. The rated set are the items the users already rated. Since we will split the data into train and test, let's call training set to the rated

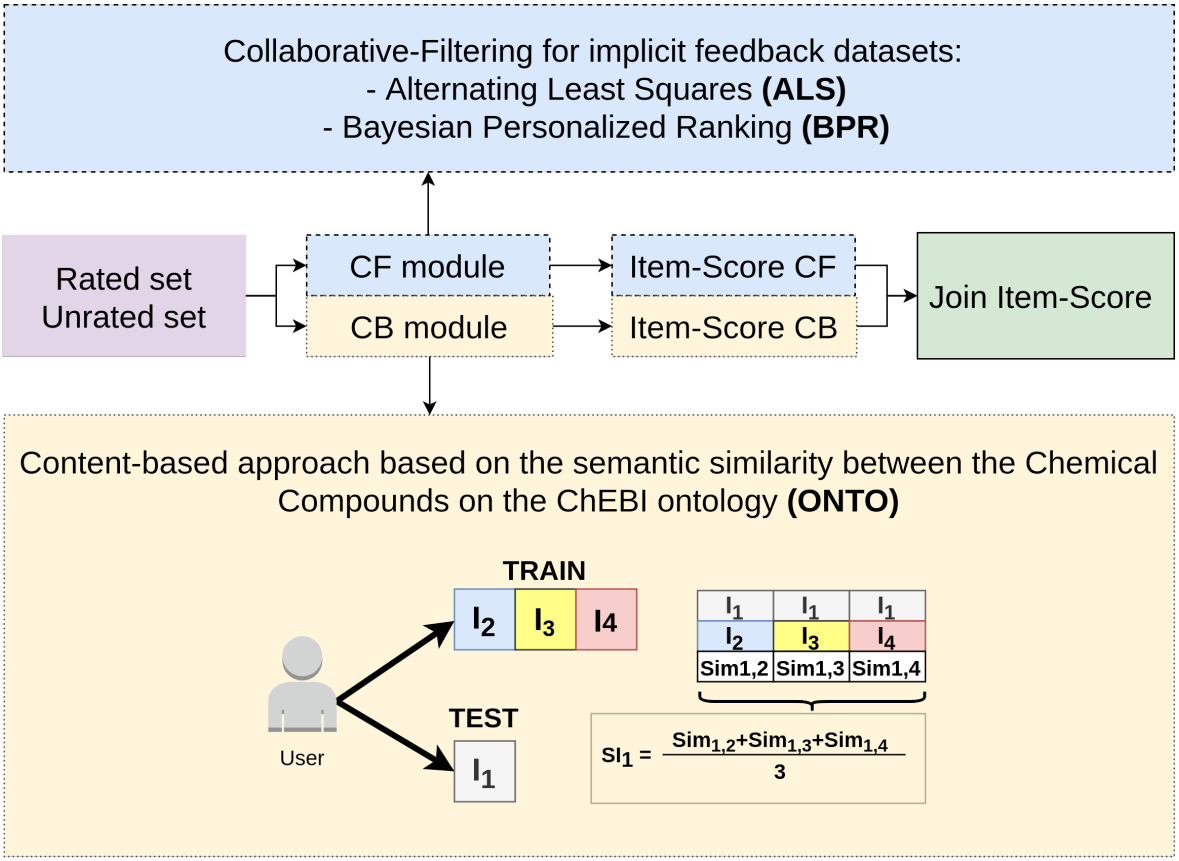


Figure 4.2: Workflow of the Hybrid semantic recommender model.

set and testing set to the unrated set. Both training and testing sets are the input for the CF and CB modules. Using CF algorithms for implicit feedback datasets, the CF module gives a score for each item in the test set. The CB module uses semantic similarity for providing a score for the items in the test set. In the last step, the scores from CF and CB modules are combined and sorted in descending order.

For the CF module, we selected two CF recommender algorithms for recommending data collected from implicit feedback, Alternating Least Squares (ALS) [90] and Bayesian Personalized Ranking (BPR)[161], both implemented in the library *Fast python collaborative filtering for implicit datasets* (implicit)[10]. These algorithms and the implementation in the implicit library are suitable for the type of dataset we are using and they were already used with similar datasets, i.e., recommendation datasets of implicit feedback, especially for recommending music playlists [200, 201]. ALS and BPR are used separately in the CF

module. The goal is to verify which combination of CF(ALS or BPR)/CB achieves the best recommendations results. The CF module outputs a score, $S_{CF}$, for each test item.

To the CB module, we developed a new algorithm, called ONTO, which is based on the semantic similarity between the items in the ChEBI ontology. This module assigns a score $S_{CB}$ to each item in the test set, calculating the semantic similarity between each item in the train and the test sets, as shown in Figure 4.2. The semantic similarity allows measuring how close two entities are in a semantic base. When using ontologies, the semantic similarity may be measured, for example, by calculating the shortest path connecting the nodes of two entities. For calculating the similarity, we used DiShIn [6, 51], a tool for calculating semantic similarities between the entities represented by an ontology. DiShIn provides three similarity measures: Resnik [163], Lin [120], and Jiang and Conrath (JC) [96]. All the previous measures are based on the information content of the entities, given by the probability of the entity appears in the ontology, and in the shared information content, calculated from the common ancestors. Resnik and Lin are real similarity measures, whereas JC is a distance measure, posteriorly converted to similarity. Lin and JC have a range between zero and one. The higher the value, the more similar the entities are. The ONTO algorithm is described in Algorithm 1.

```
Data: train = [I2, I3, I4], test=[I1]
Result: List of scores for each item in Test
test_scores = [ ];
for i in test do
    score_i = [ ];
    for b in train do
        score_i.append(sim(i,b))
    end
    test_score.append(score_i.mean())
end
```

**Algorithm 1:** ONTO algorithm.

ONTO receives as input two lists of items, train and test. The train data are the items we know the user already saw. The test data contains the items we want to know if suitable for recommending to a user. Thus, for each item in the test set, the ONTO algorithm finds the similarity to each item in the train set and calculates the mean of the similarities, as expressed by Equation 4.1.

$$S_{\text{CBI1}} = \frac{Sim_{1,2} + Sim_{1,3} + ... + Sim_{1,n}}{m} \tag{4.1}$$

In Equation 4.1, $S_{\text{CBI1}}$ is the score for item 1, which is a test item, calculated through the ONTO algorithm, and $\text{Sim}_{1,2}$, $\text{Sim}_{1,3}$, $\text{Sim}_{1,n}$ are the semantic similarities between item 1 and items 2, 3, ..., n, respectively. 2, 3 and n are train items, and m is the number of train items.

Whereas the CF module uses all the ratings from the train set to train the model, CB module only takes into account the ratings of each user. ONTO algorithm does not use any real rating of the test items when calculating the score for each item in the test set, thus we do not have the problem of introducing bias in the results.

The final score for each item in the test set in the Hybrid model is the ensemble of the scores obtained from the CF algorithms, ALS or BPR, and the score obtained by the ONTO algorithm [17]. We used a weighted method, weighting the components heuristically according to two different metrics. Metric1 is represented in Equation 4.2 and it multiplies the scores from CF and CB approaches. Metric2 is represented in Equation 4.3 and it calculates the mean of the scores.

$$Metric1 = S_{\text{CFI1}} \times S_{\text{CBI1}} \tag{4.2}$$

$$Metric2 = \frac{S_{\text{CFI1}} + S_{\text{CBI1}}}{2} \tag{4.3}$$

$S_{\text{CFI1}}$ is the score obtained for item 1, depending on the CF algorithm that we are using (ALS or BPR for our case study), and $S_{\text{CBI1}}$ is the score for item 1 obtained with the CB algorithm. Metric2 (Equation 4.3) is a more standard approach, however, Metric1 (Equation 4.2) allows that items that are really outstanding in one of the algorithms are recommended. Our goal is to prove that by combining both modules, we can improve the results of each module separately.

## 4.3 Experiments

For this work, we used a preexisting dataset, called CheRM-20, which was created by [5, 29]. The CheRM-20 is a recommendation dataset with the standard format of <user,item,rating>. According to the authors, the dataset was developed using a methodology called LIBRETTI,

which allows the creation of standard recommendation datasets by using research literature for extracting implicit feedback for the researchers. Thus, in CheRM-20, the users are authors from research papers, the items are chemical compounds, which may be linked to ChEBI ontology, and the ratings are the number of items an author wrote about a chemical. With CheRM-20, we have access to information about the researchers' past interests for chemical compounds, which allows us to develop recommender algorithms for predicting which chemical compounds the researchers may be interested now, based on their past ratings and the ratings of their similar peers.

CheRM-20 has 16.437 items, 2.193 users, and 117.020 ratings. All the users in the dataset have rated at least 20 items, i.e., the researchers considered in this dataset wrote articles about at least 20 of the 16.437 chemical compounds, even if only one article per item. This condition imposes a minimum number of items per user and it serves the sole purpose of when splitting the dataset into train and test, both datasets have a minimum number of items, providing a fair evaluation. This is a recurrent practice in other recommendation datasets, such as MovieLens [79]. On the contrary, there is no limitation for the minimum number of authors rating an item, which is an advantage because an item with only one rating (only one author wrote one paper about this chemical compound) has still the possibility of being recommended. Since this dataset's rating was collected from implicit feedback, we will use algorithms suitable for this kind of data, such as ALS and BPR.

Table 4.1 shows the variation of algorithms evaluated in this study. For CF, we tested ALS and BPR, separately. We tested different latent factors, achieving the best results for this data with 150 factors. For CB, we tested the ONTO algorithm, using three different similarity measures: Lin, Resnik, and JC. The Hybrids were developed in combinations of the CF and CB approaches, using the two different metrics for calculating the final score of each item in the test set, Metric1 - Equations 4.2 and Metric2 - Equation 4.3.

We used offline methods for evaluating the performance of the algorithms for the top@k, with k varying between 1 and 20, with steps of 1 [174]. From the vast range of metrics for evaluating recommender algorithms, we selected classification accuracy metrics and rank accuracy metrics, since they allow us to evaluate the algorithms for the relevant and irrelevant items recommended in a ranked list, and for the ability of an algorithm to recommend the items in the correct order. We use Precision, Recall (classification accuracy metrics), MRR, and nDCG (rank accuracy metrics) for this study. All the selected evaluation metrics range between 0 and 1, with values closest to 1 better. For the segmentation of the dataset into

Table 4.1: Variation of the algorithms evaluated.

| CF | CB | Metric | Algorithm |
|---|---|---|---|
| ALS | - | - | ALS |
| BPR | - | - | BPR |
| - | ONTO_JC | - | ONTO_JC |
| - | ONTO_LIN | - | ONTO_LIN |
| - | ONTO_RESNIK | - | ONTO_RESNIK |
| ALS | ONTO_JC | Metric1 | ALS_ONTO_JC_m1 |
| ALS | ONTO_JC | Metric2 | ALS_ONTO_JC_m2 |
| ALS | ONTO_LIN | Metric1 | ALS_ONTO_LIN_m1 |
| ALS | ONTO_LIN | Metric2 | ALS_ONTO_LIN_m2 |
| ALS | ONTO_RESNIK | Metric1 | ALS_ONTO_RESNIK_m1 |
| ALS | ONTO_RESNIK | Metric2 | ALS_ONTO_RESNIK_m2 |
| BPR | ONTO_JC | Metric1 | BPR_ONTO_JC_m1 |
| BPR | ONTO_JC | Metric2 | BPR_ONTO_JC_m2 |
| BPR | ONTO_LIN | Metric1 | BPR_ONTO_LIN_m1 |
| BPR | ONTO_LIN | Metric2 | BPR_ONTO_LIN_m2 |
| BPR | ONTO_RESNIK | Metric1 | BPR_ONTO_RESNIK_m1 |
| BPR | ONTO_RESNIK | Metric2 | BPR_ONTO_RESNIK_m2 |

training and testing sets, we used a 5 cross-validation approach, by splitting users and items into five folds. In each iteration we draw 20% of the users and 20% of the items as test data, and 80% as train data. We did not use a validation set, since it is not required when using a cross-validation approach. This split and evaluation method is used in several recommender system studies [90, 161, 172].

All the positive ratings in the test set are considered relevant items for the user, i.e., an item with a rating of 5 is not more relevant than an item with a rating of 1. If an author wrote one paper about one chemical compound, we consider this chemical relevant for the author. We considered the unrated items as negative ratings, i.e., not relevant for the users. For the ONTO algorithm, we also assessed how using the n most similar items affects the results, with n varying from 1, 5, 10, 15, 20, 25, 30, and all of the items.

The semantic similarity between the chemical compounds was calculated offline, using the DiShIn. Despite DiShIn robustness, the framework was not fit for a large number of items. Thus, we implemented a new functionality, Light DiShIn, which allowed us to speedup the calculation of the similarities and the feasibility of the ONTO algorithm. Light DiShIn was implemented based on Pandas [11], which is a python Framework for manip-

ulating datasets, and the use of multiprocessing, introducing the use of multiple cores for processing the similarities. Table 4.2 and Figure 4.3 show the results of the speedup in latency (Equation 4.4 [84]) of Light DiShIn when compared with the original DiShIn. The number of similarities calculated (n similarities) is 1, 30, 60 and 180, and both systems calculated Resnik, Lin, and JC similarity metrics.

$$Speedup_{Latency} = \frac{Latency1}{Latency2} \tag{4.4}$$

Table 4.2: Evaluation of the speedup latency from original DishIn to Light DiShIn. The latency is measured in seconds and n similarities is the number of similarities calculated in each iteration of the test.

| n similarities | Original DiShIn | Light DiShIn | Speed up |
|---|---|---|---|
| 1 | 0.77 | 1.66 | 0.46 |
| 30 | 20.36 | 1.79 | 11.34 |
| 60 | 41.43 | 1.83 | 22.59 |
| 90 | 62.72 | 2.07 | 30.22 |
| 180 | 121.72 | 2.39 | 50.82 |

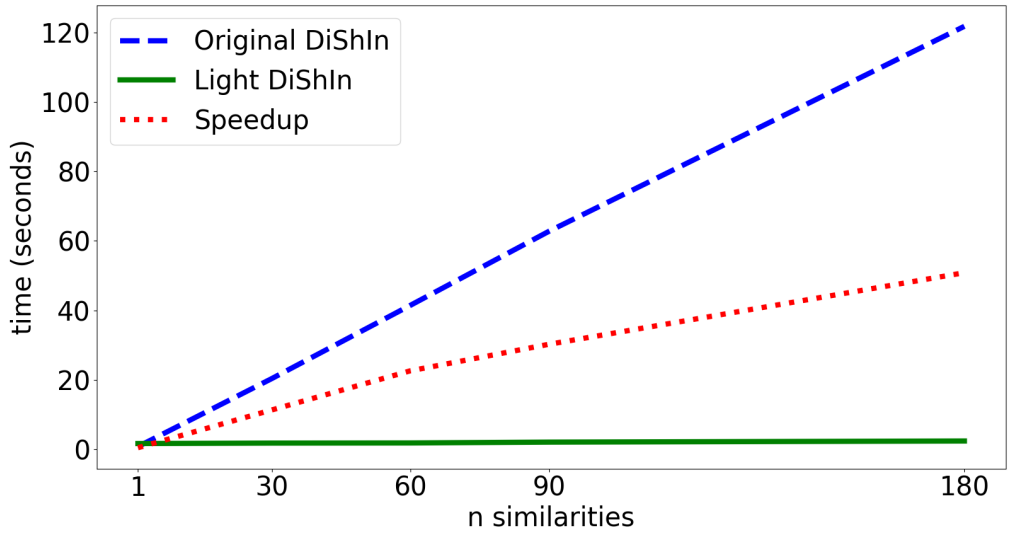


Figure 4.3: Speedup of Light DiShIn with respect to the Original DiShIn.

According to the results, for calculating the similarity between two entities (n similarities = 1), the original DiShIn is faster. Though, when increasing the number of entities and the number of similarities for calculation, the Light DiShIn is much faster than the original DiShIn, whose calculation time seems to be exponential. In our tests, the speedup latency from original DishIn to Light DiShIn achieves values of 50 times faster. For calculating the 131.538.810 similarities between the entities used for this work, we estimated that the original DiShIn would take 3.2 years. The similarities for 16.437 chemical compounds, 131.538.810 similarities, were calculated in less than a week and stored into a mySQL database for the measures Lin, Resnik and JC. This database is used by the ONTO algorithm for faster retrieving the semantic similarities of all items in the test and in the train sets. The introduction of Light DiShIn allows the viability of the execution of the ONTO algorithm, described in Algorithm 1.

## 4.4 Results and Discussion

We present the results of this study in Figures 4.4, 4.5, 4.6, and 4.7 for Precision, Recall, MRR, and nDCG, respectively, through the form of heat-maps, for all the algorithms in Table 4.1. The heat-maps show the results from top@1 to top@20, obtained using the five most similar items when calculating the scores for the ONTO algorithm, since these were the best results obtained. Following the heat-map, the more purple, the better the results. The Hybrids, both with ALS and BPR, achieved the best values for all the represented metrics. The best precision was obtained with ALS-ONTO-LIN-m2 (0.63 - top@1), improving ALS results by seven percentage points. The best recall was obtained with ALS-ONTO-JC-m2 (0.55 - top@20), improving ALS results by six percentage points.

BPR had lower results than ALS for all the evaluated metrics. However, when combining BPR with ONTO, the improvement is more significant from BPR to BPR-ONTO than from ALS to ALS-ONTO. Precision had an improvement of 13 percentage points, and recall had an improvement of six percentage points. From these results, we may conclude that the combination of ALS with ONTO achieves the highest results, but the hybrids with BPR undergo more significant increases when compared to BPR alone. These results of precision and recall show that the Hybrid algorithms are including more relevant items in the list of recommendations.

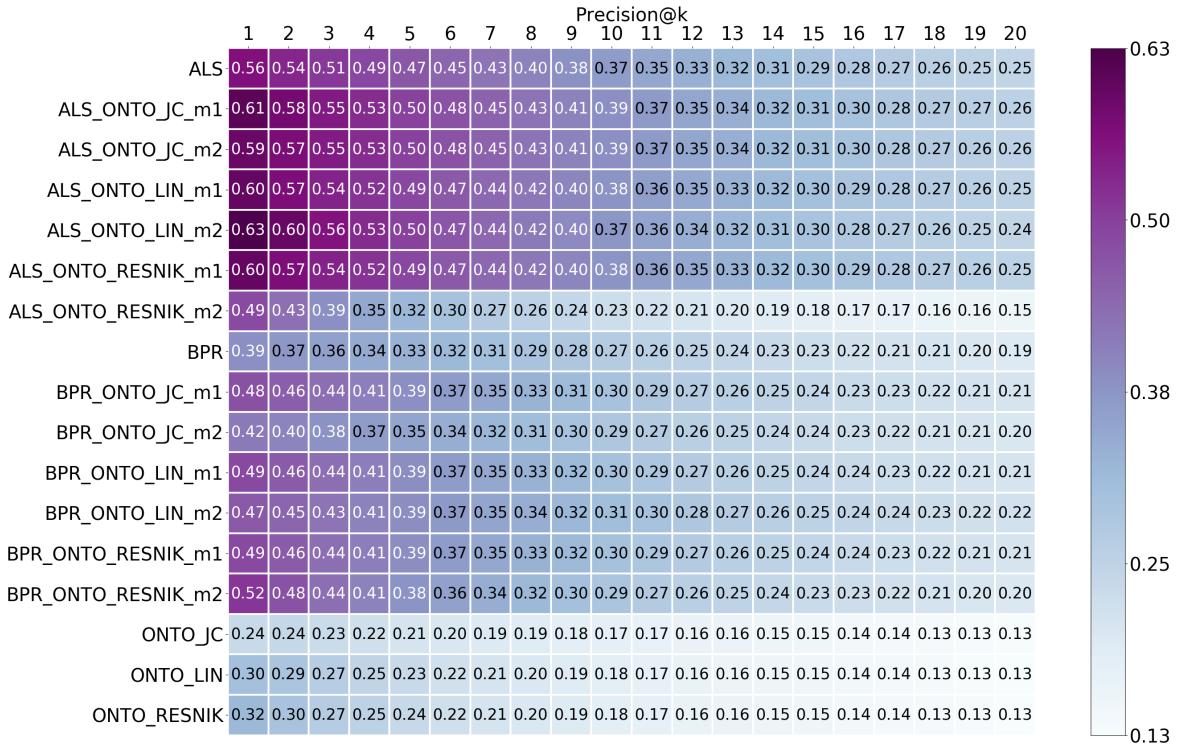


Figure 4.4: Precision results from top@1 to top@20, for ALS, BPR, ONTO and the Hybrids obtained using the 5 most similar items when calculating the scores for the ONTO algorithms.

Looking at the ranking quality metrics MRR and nDCG in Figures 4.6 and 4.7, ALS-ONTO-LIN-m2 obtains the best MRR (0.68 - top@15), with a growth of seven percentage points from ALS to ALS-ONTO-LIN-m2. ALS-ONTO-JC-m2 have the best nDCG (0.70 - top@9,10,11), more seven percentage points than ALS. For BPR, the increase was 14 percentage points for MRR and 13 percentage points for nDCG. These results of MRR and nDCG indicate that the Hybrid algorithms are effective in rearranging the ranked list of recommendations.

Analysing Figures 4.4, 4.5, 4.6 and 4.7, the ONTO algorithms alone have the lowest results in all evaluation metrics. Nevertheless, they follow the trend of the other algorithms, and when measuring these metrics for the top@20, the results are similar. ONTO has the advantage of being a CB algorithm; consequently, it does not have the problem of cold start for new items. ALS and BPR cannot be used if the item in the test set is not in the train set at least once (at least one author in the train set wrote about this chemical compound).

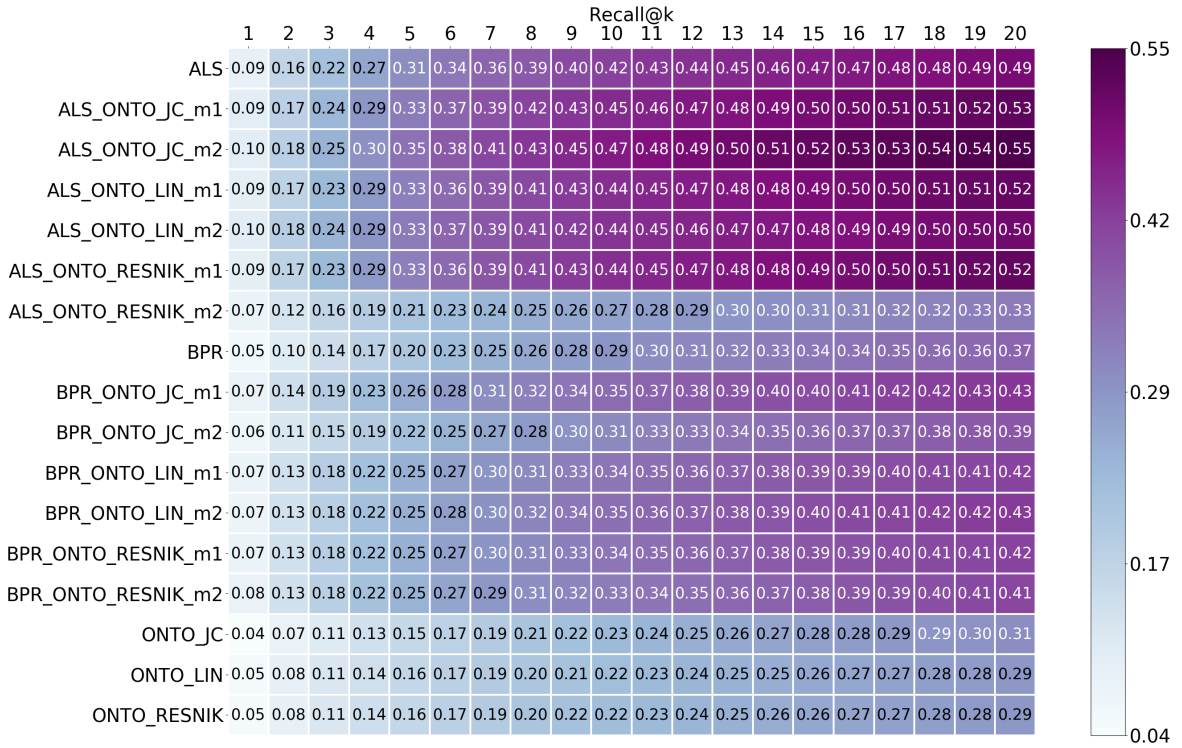


Figure 4.5: Recall results from top@1 to top@20, for ALS, BPR, ONTO and the Hybrids obtained using the 5 most similar items when calculating the scores for the ONTO algorithms.

-

However, ONTO algorithm requires the existence of all the entities in an ontology. In this case, the chemical compounds must be represented in ChEBI. When applying the ONTO algorithm to a database which does not have the ChEBI ID for the entities, we may use Named Entity Linking (NEL) methods, such as the Relation Extraction for Entity Linking (REEL) [166], which links entities recognized in the literature to the ChEBI ontology.

ONTO-LIN and ONTO-RESNIK achieved almost the same results; however, the Hybrids created with the two metrics have quiet different results. The Hybrids with ALS created through Metric1 (Equation 4.2) achieved similar results for both ONTO-LIN and ONTO-RESNIK. For Metric2, the Hybrids with ONTO-LIN are better (Equation 4.3). The ranges of the scores may explain this. Whereas LIN has a range between 0 and 1, and ALS is also returning scores inferior to 1, the same is not true for ONTO-RESNIK, since the Resnik similarity metric has an infinite upper limit. Thus, when using Metric2 for calculating the

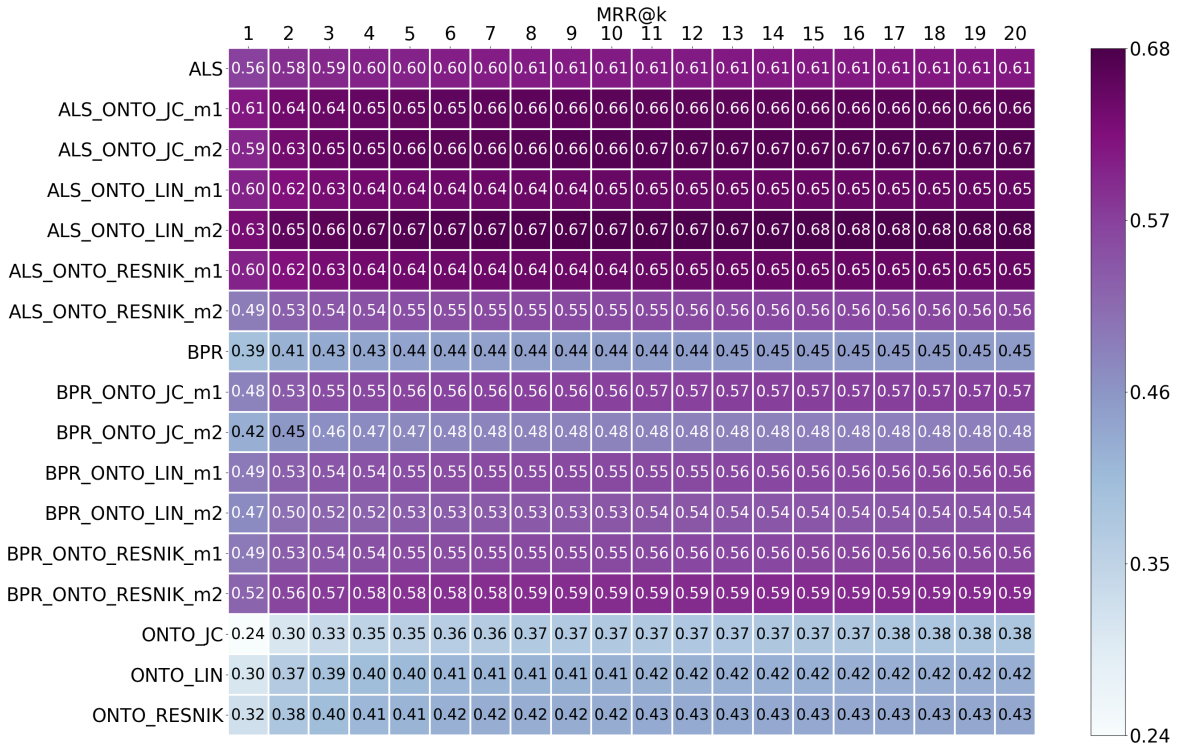


Figure 4.6: MRR results from top@1 to top@20, for ALS, BPR, ONTO and the Hybrids obtained using the 5 most similar items when calculating the scores for the ONTO algorithms.

-

final score for an item, the scores from ONTO-RESNIK have a much greater influence on the mean of the scores than the ones from ALS (<1).

For BPR, we verified that the Hybrid with ONTO-RESNIK with Metric1 achieved similar results to the ones obtained with ONTO-LIN. With Metric2, the Hybrid with ONTO-RESNIK is better than with ONTO-LIN. Due to BPR's particularity, which always increments 1 to the scores, all scores for the items from this algorithm are higher than one. Between ALS and BPR, ALS achieved the best results. Since BPR is an algorithm for ranking, it was expected to obtain better results. We believe this is because the dataset has a large number of ratings equal to one, and many items have the same relevance (difficult to rank).

We will now see how the number n of most similar items is also influencing the results of the ONTO algorithm, as well as the results for the Hybrids. Figure 4.8 shows the variation in the Precision@1, Recall@20, MRR@20 and nDCG@20 with different n most similar items in the ONTO-RESNIK algorithm and for the Hybrids ALS-ONTO-RESNIK-m1, ALS-

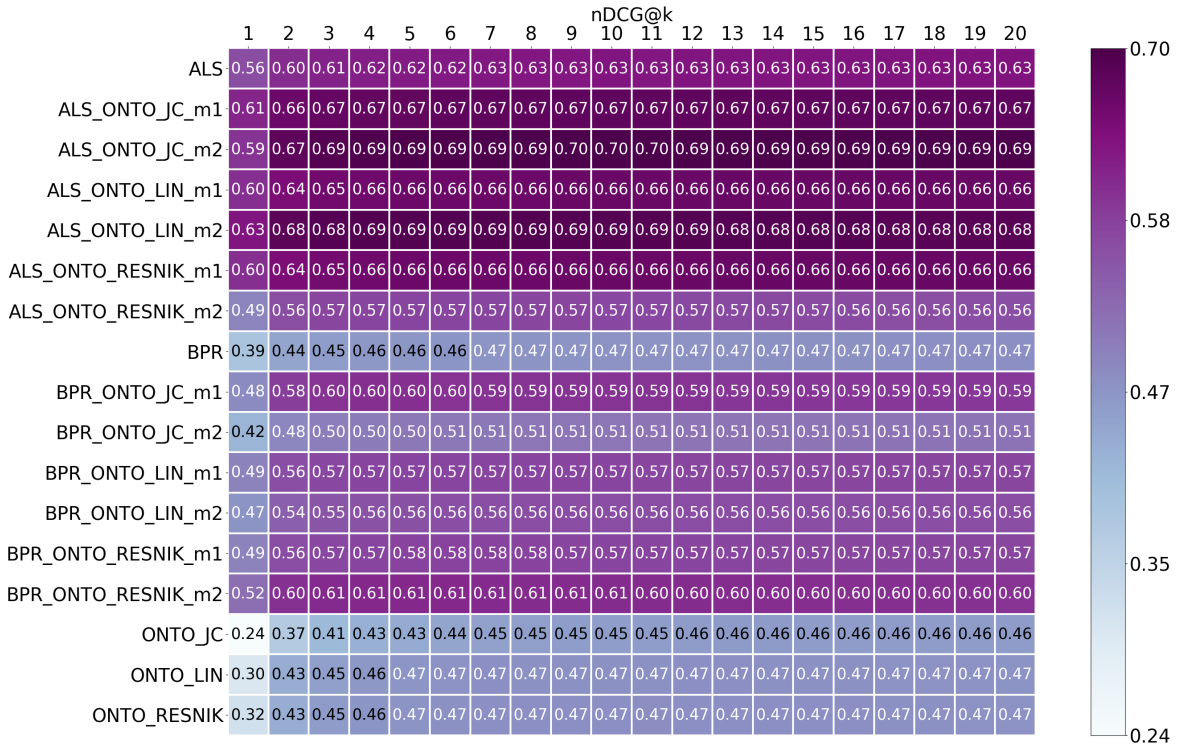


Figure 4.7: nDCG results from top@1 to top@20, for ALS, BPR, ONTO and the Hybrids obtained using the 5 most similar items when calculating the scores for the ONTO algorithms.

-

ONTO-RESNIK-m2, BPR-ONTO-RESNIK-m1, and BPR-ONTO-RESNIK-m2. ALS and BPR are also represented for better visualization of the improvement of the Hybrids. The small variations of ALS and BPR along the y axis are due to the stochastic nature of the evaluation methods.

Following Figure 4.8, the best results for ONTO-RESNIK in all the evaluation metrics are achieved using the five most similar items for calculating the scores of the items in the test set. Using a higher n, the quality metrics decrease for all the evaluation metrics. These results also affect the Hybrid algorithms, lowering the quality metrics with the increase of n. ALS-ONTO-RESNIK-m1 is the best for all evaluation metrics. Looking at the plots in Figure 4.8, we can notice a slightly descendent curve with the increase of the n most similar items. For example, the value for MRR@20 for ALS-ONTO-RESNIK-m2 is 0.6484 for n=5 and 0.6460 for n=10. This small difference may be because ALS has a much stronger influence

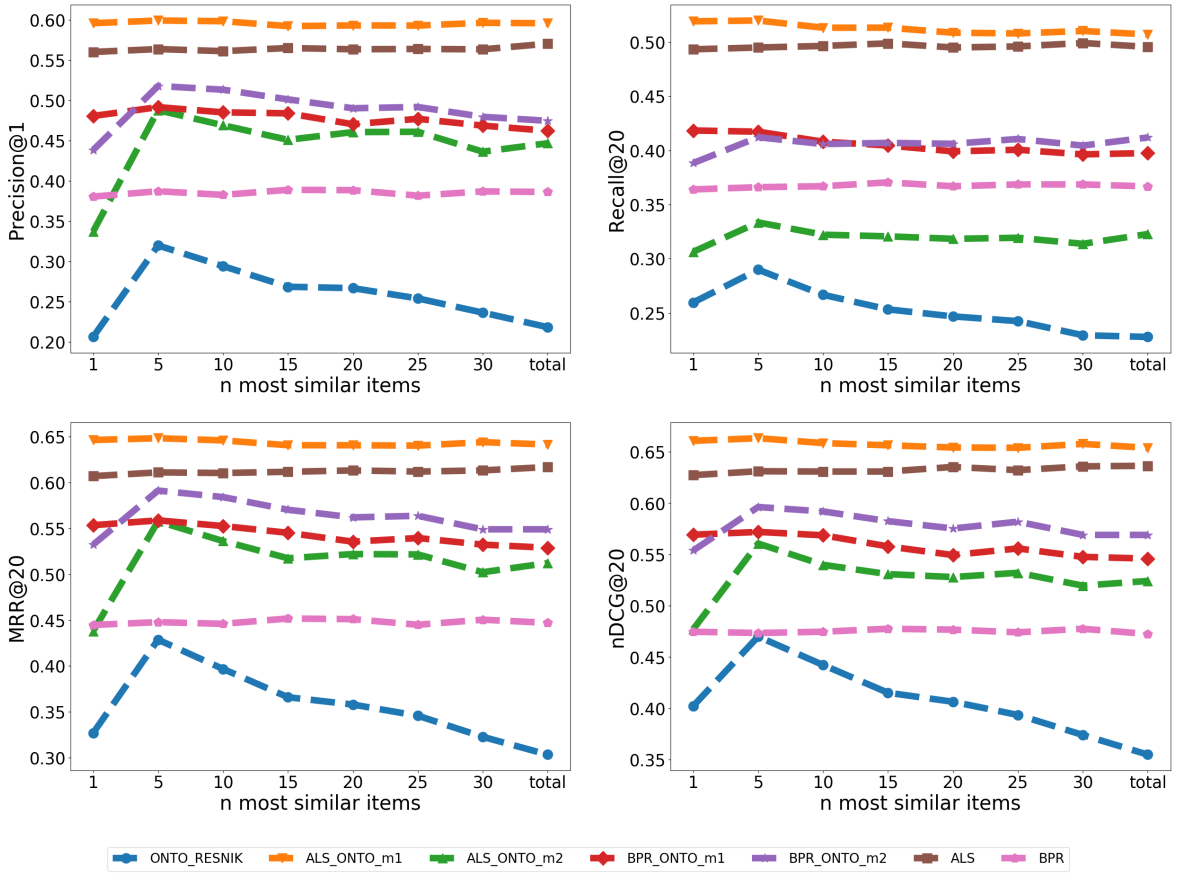


Figure 4.8: Variation of Precision@1, Recall@20, MRR@20 and nDCG@20 with different n most similar items in the ONTO-RESNIK algorithm.

on the final score than ONTO-RESNIK. As previously noticed, ALS-ONTO-RESNIK-m2 suffers a decrease when compared with ALS. This is justified by the different ranges of the scores for each algorithm, visibly affecting ALS-ONTO-RESNIK-m2 by the variation of n. BPR follows the trend of ALS results, with the difference that BPR-ONTO-RESNIK-m2 generally achieved best results than BPR-ONTO-RESNIK-m1.

The results for the variation of the algorithms with the n most similar items for LIN and JC metrics are represented in Figures 4.9 and 4.10, respectively. The analysis of the plots suggests the same behavior as the one for Resnik metric, i.e., the best results are achieved with n=5, and they degrade with the increase of n.

The following example presented in Table 4.3 shows the influence of the ONTO-RESNIK algorithm in the order of the items in the ranked list of recommendations. The Table shows

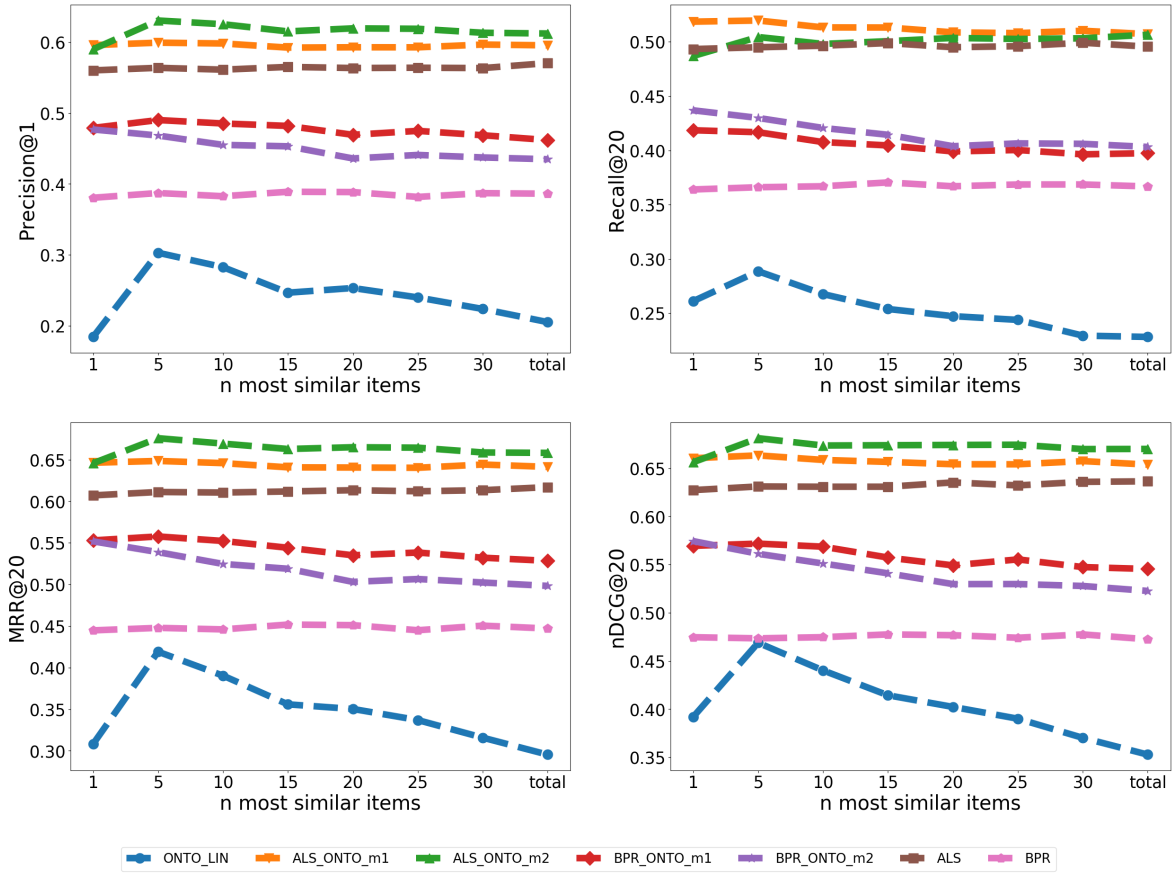


Figure 4.9: Variation of Precision@1, Recall@20, MRR@20 and nDCG@20 with different n most similar items in the ONTO-LIN algorithm.

the top@20 recommended items with the algorithms ONTO-RESNIK, ALS, BPR, ALS-ONTO-RESNIK-m1 ALS-ONTO-RESNIK-m2, BPR-ONTO-RESNIK-m1 and BPR-ONTO-RESNIK-m2, for a user with ID 174228. This user has 4 relevant items in the test set, (ChEBI ID/name: 85291 (N,1,2-trioleoyl-sn-glycero-3- phosphoethanolamine (1-)), 85292 (N-stearoyl-1,2-dioleoyl-sn-glycero-3- phosphoethanolamine (1-)), 137008 (N-acyl-1-[(1Z)-alkenyl]-sn-glycero-3- phosphoethanolamine (1-)) and 140452 (1-[(1Z)-octadecenyl]-2-oleoyl-sn-glycero-3-phosphate (2)) i.e., items in the test set with a rating higher than zero. The relevant items recommended by each algorithm are represented in gray cells. Additional info for all the chemical compounds mentioned in this text may be found in the Section Additional file 1.

For the example presented in Table 4.3, the best algorithms were ALS, ALS-ONTO-

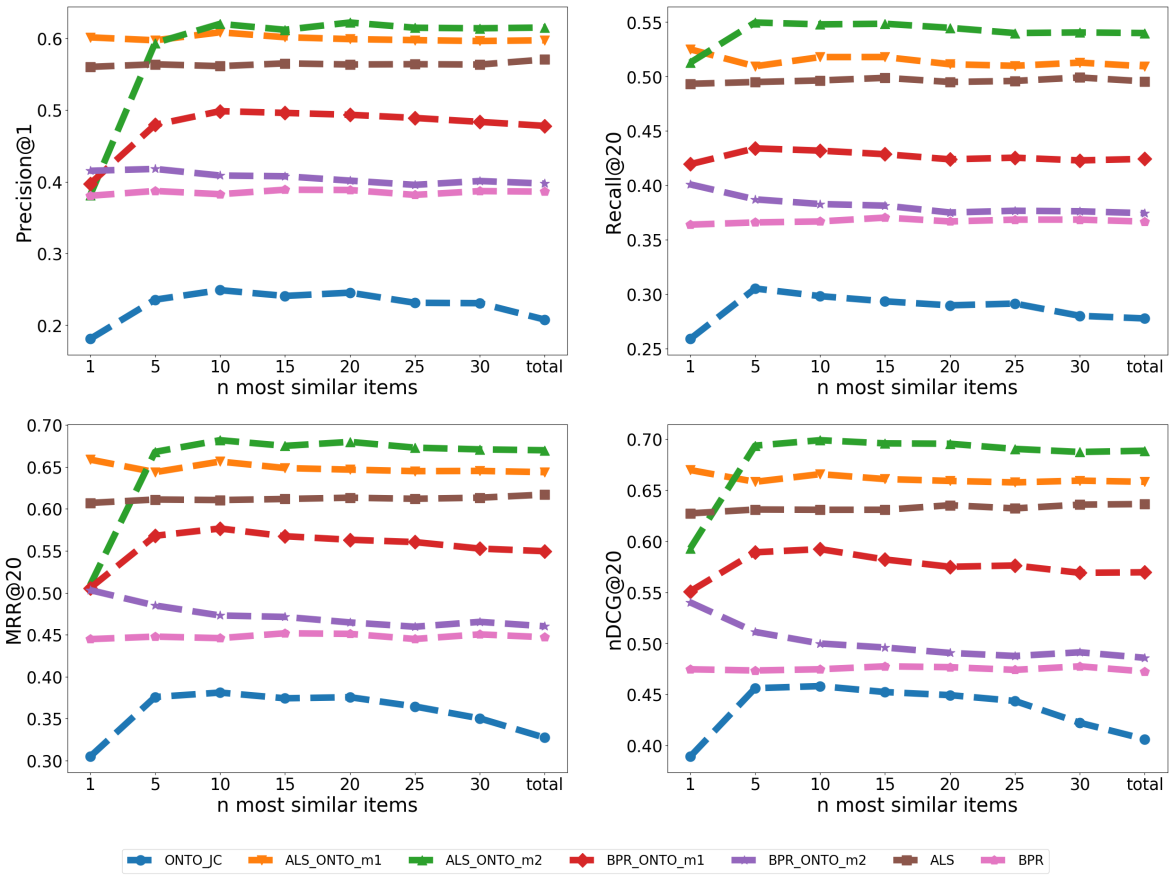


Figure 4.10: Variation of Precision@1, Recall@20, MRR@20 and nDCG@20 with different n most similar items in the ONTO-JC algorithm.

RESNIK-m1, and BPR-ONTO-RESNIK-m2, following the trend of our general results presented in Figures 4.4, 4.5, 4.6, 4.7 and Figure 4.8. Figure 4.11 shows the results for the Precision-Recall curve for all the algorithms in Table 4.1. This Figure shows that ALS-ONTO-m1 achieved the best results in the recommendation of the most relevant compounds.

Table 4.3: Influence of the ONTO-RESNIK algorithm in the top@20 list of recommendations for user 174228. This user has as relevant items the following ChEBI IDs: 85291, 85292, 137008 and 140452. The gray cells represent the relevant items recommended by each algorithm.

| ONTO-RESNIK | ALS | BPR | ALS-ONTO-m1 | ALS-ONTO-m2 | BPR-ONTO-m1 | BPR-ONTO-m2 |
|---|---|---|---|---|---|---|
| 85291 | 85292 | 23527 | 85292 | 85292 | 85292 | 85292 |
| 85292 | 85291 | 87818 | 85291 | 85291 | 85291 | 85291 |
| 85175 | 140452 | 72719 | 140452 | 85175 | 69120 | 140452 |
| 119 | 27847 | 6610 | 17697 | 119 | 140452 | 119 |
| 271436 | 175901 | 52347 | 5769 | 2904 | 137350 | 271436 |
| 2904 | 49668 | 72715 | 65495 | 271436 | 140243 | 6438 |
| 132187 | 87837 | 72754 | 27847 | 132187 | 132325 | 79079 |
| 79079 | 5769 | 69120 | 137411 | 79079 | 128770 | 132187 |
| 6438 | 17606 | 85292 | 49668 | 6438 | 69121 | 2904 |
| 140452 | 60453 | 140443 | 90983 | 140452 | 41214 | 69120 |
| 87764 | 87839 | 69340 | 132795 | 132725 | 82669 | 85175 |
| 132738 | 60747 | 132325 | 60999 | 132738 | 5635 | 137350 |
| 132725 | 76108 | 64499 | 30659 | 87764 | 63919 | 140243 |
| 65778 | 76097 | 140191 | 138802 | 78884 | 68249 | 128770 |
| 78884 | 60999 | 41214 | 138806 | 65778 | 69110 | 65778 |
| 76952 | 30659 | 91001 | 66917 | 141568 | 74912 | 69121 |
| 16108 | 31718 | 91000 | 37998 | 73275 | 140182 | 63919 |
| 77692 | 138802 | 133759 | 28850 | 138274 | 68236 | 69110 |
| 16125 | 138806 | 85291 | 66756 | 76952 | 130073 | 140182 |
| 31623 | 90983 | 67448 | 66755 | 16108 | 66394 | 130073 |

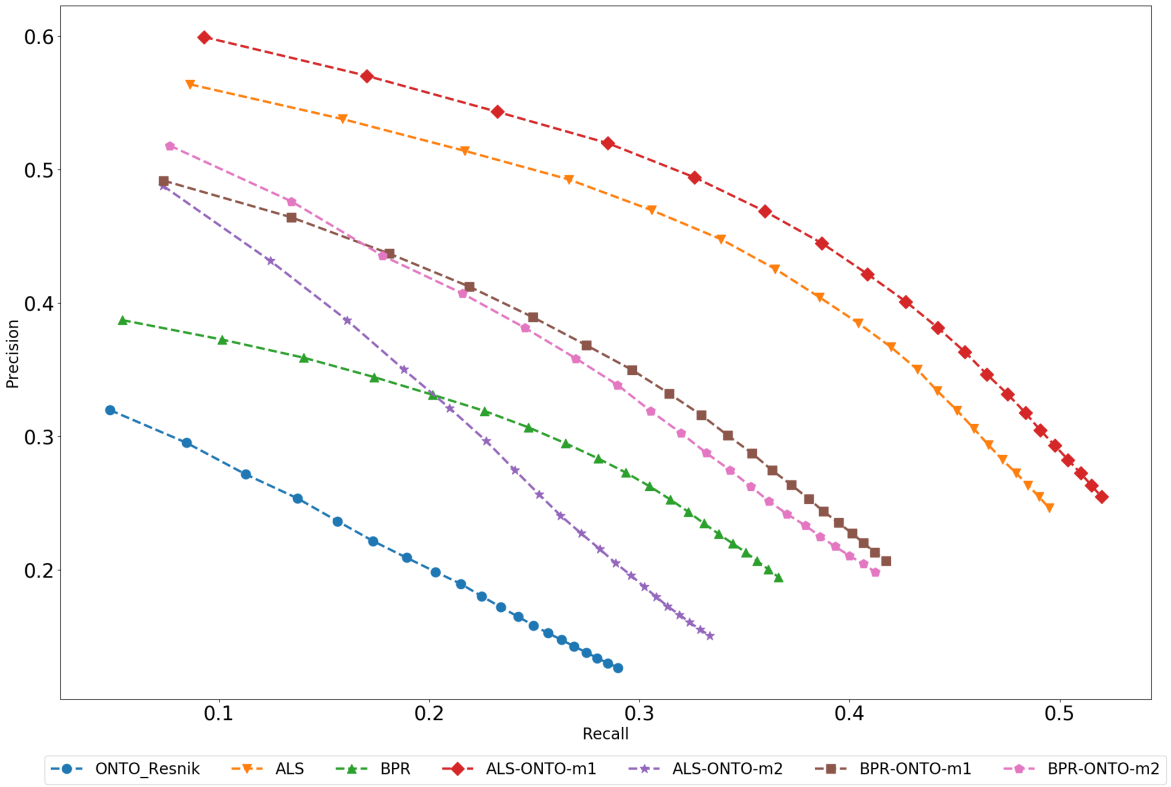


Figure 4.11: Precision-Recall curve for the algorithms ONTO-RESNIK, ALS, BPR, ALS-ONTO-m1, ALS-ONTO-m2, BPR-ONTO-m1, and BPR-ONTO-m2.

When combining ONTO-RESNIK with ALS using the Metric1, the recommended items are the same, showing that for this case, ALS has a stronger influence in the final results. When combining ONTO-RESNIK with ALS using the Metric2, it results in the recommendation of less relevant items in the first positions of the list. The Hybrid of ONTO-RESNIK and BPR using Metric1 or Metric2 improves the number of relevant items recommended in the first positions for both BPR and ONTO-RESNIK. Based on these results, we may conclude that combining the ONTO algorithms with ALS or BPR, the most relevant items are rearranged for better positions in the Hybrids, improving the chances of recommending useful content for the users in the first positions of the recommendations. Thus, the results support our hypothesis that by using a CB algorithm based on the semantic similarity between the chemical compounds for creating Hybrids with CF algorithms, improves the recommendation of relevant items.

Considering that the size of the test set for this user was larger than 3000 items and the

algorithms recommended three of the four relevant items in the first positions, one may say that RS are a solution for identifying chemical compounds of interest for scientific researches in large lists of these entities.

When using Model-based CF methods, it is not easy to justify why an item is recommended. However, our semantic approach finds a justification for the recommendations. Lets focus on Table 4.3, with the example for user 174228. The ChEBI IDs for the chemical compounds in the training set for this user were 134355, 137009, 137010, 137016, 137017, 138092, 138094, 138096, 140451, 61232, 62064, 62537, 71466, 78097, 78940, 85277, 85293, 85294, 85295, 85296, 85297, 85298, 85299, 85301, 85302, 85303, 85304, 85334 and 85335. The ONTO algorithm finds the semantic similarity between each item in the testing set (more than 3.000 items) and these items in the training set. The score for each item in the testing set is the mean of the similarity values. Thereby, for example, for item 85291, the score of ONTO-RESNIK is 4.67, being this the higher score for all 3.000 items in the test set. Interestingly, the score for item 85292 is also 4.67, which is justified by the fact that both items 85291 and 85292 are descendants of the item 62537, and share the same amount of common ancestors. This means that the items 85291 and 85292 share the most similarity with the items that we already know the user liked.

From a semantic and chemical point of view, both 85291 and 85292 are children of Organophosphate oxoanion (58945), which is an organic phosphoric acid, as well as a large number of compounds in the training set of this user - 62537, 78097, 85277, 85293, 85294, 85295, 85296, 85297, 85298 and 85334. Thus, it makes sense that both are recommended to this user, and by the test set, these are true positives, because we know the user had interest in these compounds. Another large group of items in the training set of this user are Bronsted bases (molecular entity capable of accepting a hydron from a donor) - 71466, 85299, 85301, 85302, 85303, 85304. The compound recommended by the ONTO algorithm in the third position (85175) is also a Bronsted base, thus, highly similar to these items in the training set. However, this compound is a false positive from the evaluation point of view, i.e., we don't know if the user already had interest in this compound. Nevertheless, and based on the training set, if we recommend this item to the user, she/he will probably have interest in its study. This analysis is not possible for the CF algorithms. However, with the hybrids, we can also relate the items semantically and guide the user to study new compounds. For example, ALS-ONTO-m1 recommends in the fourth position the item 17697 (N-acetylserotonin). Despite this compound not being in the list of relevant items for this user, it is semantically

similar to 85299 and 71466, which are from the group of Bronsted bases, and may be useful for this user research.

The only item in the list of relevant items which is not recommended by any algorithm is the 137008 (false negative). The reason this happens in the CF algorithms is because this item has a low number of users associated to it (3 users had interest in this item, the mean is 7 users by item). The ONTO algorithm is not able to retrieve this item in the list of recommendations due to a limitation of the DiShIn. The ID 137008 is a secondary ID for the compound 140403 (name: N-acyl-1-[(1Z)-alkenyl]-sn-glycero-3-phosphoethanolamine(1-)) and DiShIn is not able to calculate the similarity for the secondary IDs because it only works with primary IDs.

Table 4.4 presents another example of recommendation using the ONTO-RESNIK algorithm, for the user 33142. In this example, we show the relevant items recommended and the relevant items not recommended in the top@20 list.

The relevant items recommended (77367, 77380, 84078, 84082) have a high semantic similarity with the items in the training set of this user. All the four are structural derivatives of oligosaccharide and carbohydrate. In the list of relevant items not recommended, we also have an item with these characteristics (77629); however, the score of this item is lower than the score of the last recommended item in the top@20, and that's why it is not recommended. The other two items (59484 and 134230) do not share high semantic similarity with the train, explaining why they are not recommended.

Considering the results, the hybrid semantic recommender system presented in this work is suitable for the recommendation of chemical compounds of interest for researchers dealing with large scale datasets. The use of a hybrid approach not only improved the results of the individual module, but also provides recommendations of chemical compounds based on the interests of similar peers (CF) and being able of justifying the recommendation (CB).

The model described in this paper may also be applied to other databases in which it is possible to measure the semantic similarity between the entities. Consider the DrugBank [8], a major database of drugs, largely used in the pharmaceutical field. DrugBank, similarly to ChEBI, has chemical compounds, such as Acetaminophen. It provides detailed information about the chemicals, about their identification, pharmacology, or interactions, for example. It is also created in a hierarchical format, having a Chemical Taxonomy, which provides information such as **Super Class**, **Class**, **Sub Class**, and **Direct Parent**. This structure allows the calculation of semantic similarity between the chemicals, as shown in [141]. The ONTO

Table 4.4: Results of ONTO-RESNIK for the user 33142. The table presents the training items for this user, the relevant items in the testing set, the scores of these items calculated using the ONTO-RESNIK algorithm and the top@20 recommendations, and respective scores. In gray are the relevant items which were recommended (77367, 77380, 84078, 84082) and in red the relevant items which were not recommended in the top@20 (59484, 77629, 134230).

| Training | Relevant | Score | Top@20 | Score |
|---|---|---|---|---|
| 60561 | 59484 | 2.18 | 134258 | 7.59 |
| 62642 | 77367 | 6.82 | 61755 | 7.59 |
| 62664 | 77380 | 6.74 | 84082 | 7.59 |
| 62996 | 77629 | 6.60 | 84078 | 7.59 |
| 62997 | 84078 | 7.59 | 59949 | 7.48 |
| 62998 | 84082 | 7.59 | 90930 | 7.29 |
| 77314 | 134230 | 4.33 | 66139 | 7.29 |
| 77374 | | | 60381 | 6.87 |
| 77378 | | | 77367 | 6.82 |
| 77381 | | | 90775 | 6.82 |
| 77382 | | | 77380 | 6.74 |
| 77384 | | | 62471 | 6.65 |
| 77385 | | | 61847 | 6.65 |
| 77598 | | | 87452 | 6.65 |
| 77613 | | | 87799 | 6.65 |
| 77625 | | | 61713 | 6.65 |
| 77626 | | | 61329 | 6.65 |
| 77627 | | | 61334 | 6.65 |
| 77628 | | | 62534 | 6.65 |
| 84081 | | | 67164 | 6.65 |
| 84084 | | | | |

algorithm can then be applied using these similarity measures for providing the recommendation, and combine it with other recommender algorithms such as ALS or BPR.

## 4.5 Conclusion

A major challenge in the identification of new chemical compounds is the increasing number of entities added to repositories. In this work, we presented a solution to this problem in the form of a recommender system. Our approach consists of a Hybrid recommender

model for recommending ranked lists of chemical compounds. The Hybrid model has two modules, one using a CF approach and the other a CB approach. In the CF module, we used ALS or BPR, specific algorithms for implicit feedback datasets. The CB module consists of a new algorithm called ONTO, based on the semantic similarity of the chemical compounds in ChEBI ontology. The hypothesis presented was that by combining the scores obtained by each module, we would improve the results of both modules separately. The Hybrids between ALS and ONTO were the ones with the best results for all the evaluation metrics, improving the results by more than ten percentage points. The obtained results support our hypothesis since the results for the Hybrids algorithms are higher when compared with the individual algorithms. Even though ALS and BPR are better than the ONTO versions of the CB approach, when combined, the ONTO algorithm rearranges the positions of the items, recommending more relevant items in the first positions of the rank. Thus, with this work, we contributed with a recommender framework for chemical compounds, a new CB semantic recommender algorithm based on ontologies, a new Hybrid recommender algorithm for datasets of implicit feedback, a dataset with the semantic similarity between more than 16.000 chemical compounds, and also a faster method for calculating the similarities between large numbers of entities. We believe that this work is suitable for other fields of study, thereby, for future work, we intend to assess the ONTO algorithm, as well as the Hybrids, with entities from other ontologies, such as GO and DO. We would like to improve the results for precision and recall, for example by performing Named Entity Recognition (NER) in the articles from where the CheRM-20 dataset was created, to have more items related to each user. Other hypotheses are testing other similarity metrics, and using the relations between the compounds to provide the recommendations.

# 5

# SeEn: A sequential enrichment approach for sequence-aware recommendations

This chapter addresses the Research Question 3: Will the semantic enrichment of sequences of items with the n most similar items improve the results of state-of-the-art sequence aware recommendations algorithms?

The recommendation of items based on the sequential past users' preferences has evolved in the last years, mostly due to deep learning approaches, such as BERT4Rec. However, in scientific fields, recommender systems for recommending the next best item is not widely used. The main goal of this work is to improve the results for the recommendation of the next best item in scientific domains, using sequence aware datasets and algorithms. In the first part of this work, we present the adaptation of a previous method (LIBRETTI) for creating sequential recommendation datasets for scientific fields. The results were assessed in Astronomy and Chemistry, with the creation of two datasets for recommending open clusters of stars and chemical compounds, respectively. In the second part of this work, we propose a new approach whose goal is to improve the datasets, not the algorithms, to obtain better recommendations. The new hybrid approach is called sequential enrichment (SeEn), which consists of adding to a sequence of items the n most similar items after each original item. The results show that the enriched sequences obtained better results than the original ones. The Chemistry dataset improved approximately seven percentage points and the Astronomy dataset by 16 percentage points, for Hit Ratio and Normalized Discounted Cumulative Gain.

## 5.1 Introduction

Everyone appreciates a recommendation for a desired item, a good movie, or an interesting book. Why would it be different for researchers? An investigator studying the effect of chemical compounds in the creation of new drugs would be more than glad if a system recommended the next best match to their studies preferences. The number of new scientific entities grows every day, requiring new tools for knowledge extraction. Recommender systems (RS) approaches suits these situations since they can deal with large quantities of data and also provide personalized recommendations, according to the researchers' preferences [196]. The main challenge is that there are few studies in recommender systems for scientific fields, primarily due to the lack of open-access datasets.

The recommendation of items has been a topic of interest in many fields, such as music, movies, e-commerce, and even scientific fields as Chemistry and Astronomy. In some cases, the sequence of user/item interaction is important since the next item of interest may depend on the previous ones. Despite a large number of studies on sequence-aware recommendation systems (RS) [157], their use in scientific fields is not broad.

RS are by definition software tools and techniques that provide suggestions for items that are most likely of interest to a particular user, mostly used in the recommendation of movies, music, and e-commerce. There are two major approaches in RS, collaborative-filtering (CF) and content-based (CB) [165]. CF uses only the users' preferences as input for the recommendations, calculating the similarity between users. If John Smith and Jane Smith read the same article, they are similar users. Suppose Jane Smith reads a second article, it will be recommended to John Smith. The example refers to memory-based CF. Instead of directly calculating the similarity between the users, CF may be model-based, using machine-learning, for example, matrix factorization and deep learning, for predicting the ratings of unseen items. This approach has some challenges. It cannot deal with items without any rating or users who have not rated any item (cold start for new items and new users, respectively). In CB approaches, the recommendations do not depend on the similarity of the users but on the similarity of the items. If Jane Smith read an article, CB algorithms will recommend to her similar articles to the one she read without involving the preferences of other users. CB solves the problem of cold start for new items. However, for calculating the similarity between the items, we need a characterization of each items specified by a set of features. If the

item is a movie, the features may be the genre, actors, and director. Then, we can use similarity metrics, such as cosine similarity, or machine-learning methods, for example, clustering approaches, to group the items by similarity. A particular type of similarity is the semantic similarity shared by the items. For calculating the semantic similarity of the items, we may use ontologies, which are vocabularies hierarchically organized [28, 190]. Ontologies are widely used in Health and Life Sciences, with a large number of bio-ontologies being made available and maintained in the last few years, such as the Chemical Entities of Biological Interest (ChEBI) [81], the Gene Ontology (GO) [49], and the Disease Ontology (DO) [169]. Bio-ontologies are important since they help the researchers to identify an entity unequivocally, and they also enable the computation of the semantic similarity between the entities. Hybrid CF-CB approaches are used to get the best of both CF and CB. One of the methods used is the completion of the unknown ratings by calculating the similarity between the items that the user already rated and the unrated items (CB). The completed matrix is then used in CF approaches for finding the most similar users and providing the recommendations [117].

All the recommendation approaches presented in the previous paragraph depend on information about the users' preferences, usually in the form of ratings. These ratings may be explicit, for example, through a stars classification system, or implicit, where the users' preferences are collected from their activities, such as "user u watched movie b". Open-access datasets with the users' preferences are common in the fields of movies, TV shows, music and e-commerce. For movies we have Movielens [79] and Netflix [35] datasets. In music, we find datasets provided by Spotify[1], and for e-commerce, Amazon[2] has been relentless in the promotion of these datasets, which translates in a large number of algorithms applied to these fields.

Standard and open-access datasets with information about users' preferences are scarce in scientific fields, such as Chemistry and Astronomy. Thus, if we wish to develop an algorithm for recommending chemical compounds, we may lack access to a dataset with information about the past preferences of a group of users. Given this limitation, in Barros et al. [29] we developed a new methodology called LIterature Based RecommEndaTion of scienTific Items (LIBRETTI) whose goal is the creation of <user, item, rating>datasets, related with

[1] https://www.kaggle.com/yamaerenay/spotify-dataset-19212020-160k-tracks/tasks?taskId=961

[2] http://snap.stanford.edu/data/web-Amazon.html

scientific fields. These datasets are created based on the major resource of knowledge available in Science: scientific literature. The users are the authors of the publications, the items are the scientific entities (for example chemical compounds or diseases), and the ratings are the number of publications where the author mentioned the entity.

Typical recommendation datasets have matrix format, with items in the columns, users in the rows, and the ratings being the pairs <user, item>. However, some situations require knowledge about the order in which the items were seen, especially in scientific fields, where the scientific entities raise different degrees of interest to the researchers along the time. For example, according to Pubmed[1], the chemical compound Paracetamol[2] had a spike in the number of research articles in 2020, as shown in Figure 5.1.

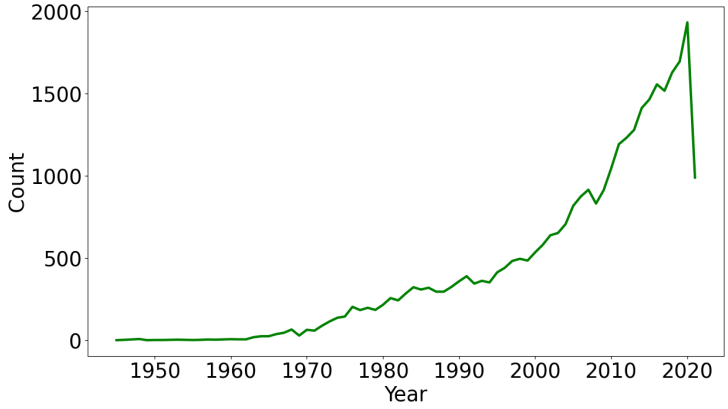


Figure 5.1: Paracetamol research articles by year in Pubmed.

Sequence-aware recommendations arise to solve the problem where the order of the items is important to provide the recommendation of the next best item. Sequence-aware recommendations have been developed and applied for movies, music, e-commerce, but to the best of our knowledge, not in scientific fields. There are already algorithms dealing with sequential recommendations. There are some common baselines, e.g. selecting the most popular, and k-nearest-neighbors approaches. We also have non-deep learning approaches, such as matrix factorization and Markov chains [175]. Most recently, deep learning approaches have emerged as state-of-the-art for sequence-aware recommendations, such as, GRU4Rec [87], CASER [188], SASRec [97] and BERT4Rec [185]. The last one outperformed all the other

[1] https://pubmed.ncbi.nlm.nih.gov/?term=paracetamol
[2] https://www.ebi.ac.uk/chebi/searchId.do?chebiId=CHEBI:46195

algorithms. BERT4Rec is based on the famous BERT model from Google. Its major difference from other deep learning algorithms is that it is bidirectional, reading the sequences from left to right and right to left. The first step of BERT4Rec is an embedding layer, where it combines the position and the item, and then several transformer layers. The transformer method is a deep learning model for Natural Language Processing (NLP), based on multi-head self-attention and another layer of position-wise feedforward. BERT4Rec has several transformer layers, and they are connected bidirectionally. For training, a percentage of the items are masked in the sequence. The output has the probability for the next items.

In many fields, such as movies and TV shows, it is possible to simulate implicit sequential datasets by using the timestamp associated with the ¡user, rating¿ pair, and converting the ratings to binary [185]. In science, the available datasets do not have this information, and and even our datasets created using the LIBRETTI methodology do not consider a timeline for the user's interaction with the items. In this work, we recreated the LIBRETTI methodology to create new datasets aware of the sequence of the interaction between user and item, thus we may use sequence aware recommendation algorithms for recommending the next best item for a researcher. The methods will be assessed in the fields of Chemistry and Astronomy for recommending chemical compounds and open clusters of stars, respectively.

Besides creating new sequential recommendation datasets, in this work, we also present a new methodology for sequence-aware recommendations in the fields of Chemistry and Astronomy, focused on the enrichment of the dataset, not on the improvement of the algorithm. The proposed methodology, called Sequence Enrichment (SeEn), employs a hybrid approach by adding to a sequence of items the n most similar items after each original item. The new sequence is then passed as input for state-of-the-art sequence-aware recommendation algorithms with the goal of improving the results when compared with the not enriched or original sequence.

As seen previously, the sequence of the user-item interaction is essential in scientific domains, thus the goal of this study is to prove that sequence-aware datasets are better for recommending the next best item in scientific fields.

The main contribution of this work are:

- A sequential dataset in the field of Chemistry, for the recommendation of chemical compounds;

- A sequential dataset in the field of Astronomy, for the recommendation of open clusters of stars;

- A new hybrid data-driven approach (SeEn) for sequence-aware recommendations.

## 5.2 Related work

There are a large number of studies regarding sequence-aware recommendations. Related to this work, we will present studies whose goal was to improve BERT4Rec recommendations.

In [94] the authors developed a framework called CITIES, whose main goal was to improve the recommendation of tail items. The architecture consists on three layers: item embedding, sequence modeling, and recommendation layer. BERT4Rec was used in the sequence modeling layer. The framework improved the recommendation of tail items when compared with BER4Rec, however, it did not improve the recommendation of head items.

[241] developed the S3-Rec framework, based on mutual information maximization. The approach uses the attributes of the items into a new embedding layer, which is then passed into a bidirectional self-attention layer, such as BERT4Rec. The results outperformed BERT4Rec, but it has the extra cost of a new layer.

[210] created the framework HyperRec, which is based on short-term item correlations in a hypergraph, correlated by the purchase time. The embedding of the items created through the hypergraphs and the short-term user intent are then fused, and passed into a self-attention model, such as BERT4Rec. HyperRec improved the results when compared with BERT4Rec. However, HyperRec does not have a CB component and will not be able to recommend new items.

[127] proposed a sequence-to-sequence (seq2seq) strategy, instead of the usual sequence-to-item (seq2item) strategy. Seq-to-seq is used in parallel with seq2item for extracting extra information from the datasets. The approach outperformed the tested baselines, including BERT4Rec. This method does not use information about the content of the items.

## 5.3 Methods

### 5.3.1 Datasets

For this work, we created two datasets from different scientific fields, one from Chemistry, where the items are chemical compounds, and another from Astronomy, where the items are Open Clusters of Stars. Both datasets were created according to the LIBRETTI methodology [29] modified to create sequences of items by user, ordered by the year of publication of the paper mentioning each item. Figure 5.2 shows the original scheme of LIBRETTI presented in Barros et al. [29] VS the new sequential module. In both modules, LIBRETTI requires a list of scientific items and articles where the items are mentioned. Then, we extract the authors from the articles, and create datasets of user (author), item (scientific entity), and rating (number of articles where the author mentioned the entity) for the original LIBRETTI. In the sequential module, the ¡user, item¿ interactions are ordered by the publication year of the article. The rating is always 1.

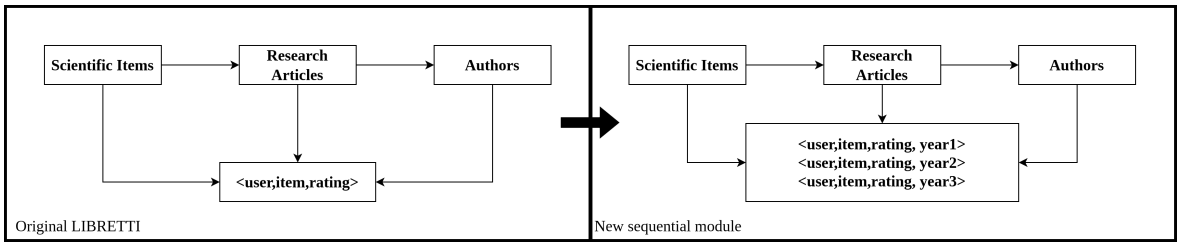


Figure 5.2: Scheme of the original LIBRETTI methodology vs new sequential module.

The Chemistry dataset, called chemicals Recommendation Matrix (**chERM**) is a dataset whose items are chemical compounds represented in the chEBI ontology. The first chERM dataset was created in [29], and it was already used in some works [30, 31] for testing new algorithms for recommending chemical compounds. The original chERM dataset has the format of ¡user, item, rating¿, the users being authors of research articles, the items being chemical compounds, and the ratings the number of articles where a user mentioned the item. In the new chERM dataset (chERMSeq), the items are organized by year for each user, as represented in Figure 5.7: Original chERMSeq. In these studies, the dataset chERMSeq was used to evaluate a new hybrid recommender algorithm based on the semantic similarity of the chemical compounds, calculated through the ChEBI ontology.

The Astronomy dataset, called astronomical Recommendation Matrix (**aRM**), is a dataset of Open Clusters of Stars, whose items were collected from the Dias catalogue of open clusters [57]. The method for creating this dataset is the same used in [29], except that the initial list of Open Clusters was updated. The new aRM dataset (aRMSeq) was created with the same method as chERMSeq.

Unlike datasets such as Movielens, chERMSeq and aRMSeq did not need to be converted to binary ratings, since they are already implicit feedback datasets, whose rating values are 1 (author mentioned entity in article), or 0 (author did not mention entity in article).

### 5.3.2 Sequential Enrichment Approach

The recommendation of the next best item for a user is still a challenge. Sequential datasets are usually of implicit feedback, highly sparse, and with no negative feedback. In this work, we propose a solution for the datasets' sparsity by introducing a hybrid sequential enrichment approach based on the similarity of the items.

Figure 5.3 shows the general pipeline of the SeEn approach. It consists in introducing after each item in a sequence its n most similar items. This allows reducing the sparsity of the dataset. The new enriched sequence is then passed into sequence-aware recommender algorithms. We hypothesise that using the SeEn approach will improve the results of state-of-the-art algorithms.

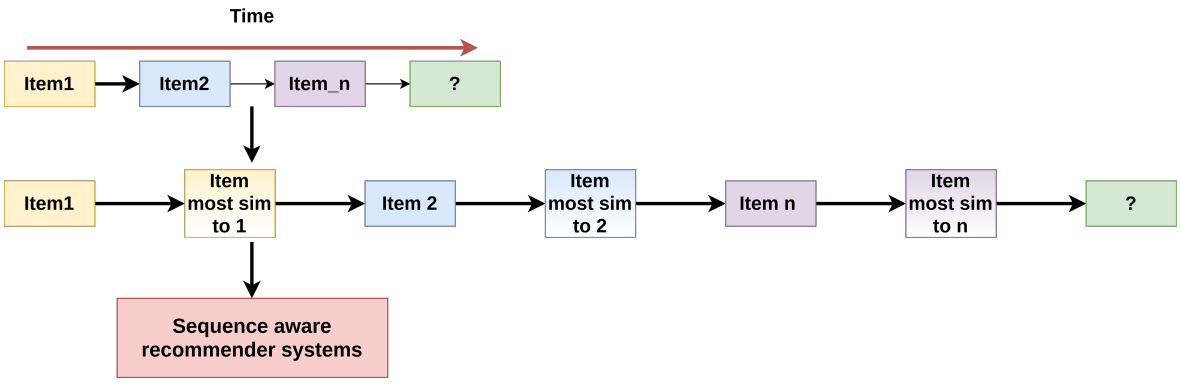


Figure 5.3: SeEn: Sequential enrichment approach general scheme.

The input of SeEn requires a recommendation dataset, where each user has a sequence of items with which the user already interacted, ordered by interaction time, for example, by

year or timestamp. After each original item, the method introduces the n most similar items to the original into the sequence. For calculating the similarity, we need a knowledge source with the features of the items, which will depend on the field of study.

If we have numerical features, we may directly apply similarity metrics, such as cosine or Jaccard, for finding the most similar items. These metrics calculate the similarity between two vectors [149]. In other cases, we may use semantic similarity for finding the most similar items. The semantic similarity may be measured based on the semantic structure of an ontology, allowing to have the closeness in meaning between the entities [51]. Some known metrics are Resnik [163], Lin [120], and Jiang and Conrath (JC) [96].

### 5.3.3 Evaluation

This work is divided into two evaluation phases. First, we want to identify the best algorithm and prove that using sequential datasets to recommend the next best item results in better recommendations than when not considering the interaction sequence. Second, we want to evaluate if enriching the datasets with the n most similar items improves further the results.

For the first phase of the evaluation, we used the following algorithms for testing both chERMSeq and aRMSeq datasets:

- **The most popular (Most-Pop)** - The most popular recommendation algorithm is a basic algorithm that considers the items with the larger number of ratings and recommends the top@k to the user. The sequence of the items is not relevant.
- **Alternating Least Squares (ALS)** - ALS is a latent factor algorithm, specific for implicit feedback datasets, that addresses the confidence of a user-item pair rating, which goal is to minimize the least squares error of the observed ratings by factorizing the rating matrix in user and item matrix. The order of the items is not relevant.
- **BERT4Rec** - BERT4Rec is a sequence-aware recommendation algorithm with state-of-the-art results in this field. The sequence of the items is relevant.

Table 5.1 shows the algorithms tested with which datasets. The Most-Pop and ALS algorithms were tested with the chERMSeq and aRMSeq datasets, but in these cases, the order is not relevant. BERT4Rec was tested with the chERMSeq and aRMSeq not sequential,

i.e., each user's sequences were shuffled. BERT4Rec was also evaluated with the sequential chERMSeq and aRMSeq datasets.

Table 5.1: Evaluation of sequential datasets: algorithm and version of the dataset.

| Algorithm | Dataset |
|---|---|
| Most-Pop | chERMSeq |
| | aRMSeq |
| ALS | chERMSeq |
| | aRMSeq |
| BERT4Rec | chERMSeq not seq |
| | aRMSeq not seq |
| | chERMSeq |
| | aRMSeq |

To guarantee the quality of the datasets, we limited the minimum number of user/item interactions to 20. In aRMSeq, we also determined the maximum number of the sequence to 800, since one of the Transformers layer limitations of BERT4Rec is the maximum size of the sequence [112].

For the second phase of the evaluation, we tested the SeEn approach. Table 5.2 shows the proceedings experiments. The selected algorithm was the BERT4Rec given its higher performance. The datasets used were the chERMSeq and the aRMSeq. Both were tested in their original sequential form and adding to the sequence the one, five, and ten most similar items, as shown in the Sequential Enrichment Approach Section. We also tested adding random items to the sequence in the same proportion, thus we may evaluate the difference between adding random items or items selected according to the similarity. The original and the enriched sequences datasets were then used for training models with BERT4Rec [185].

Table 5.2: Evaluation of the SeEn approach.

| Dataset | Algorithms | SeEn |
|---|---|---|
| chERMSeq<br>aRMSeq | BERT4Rec | Original<br>Sim + 1<br>Sim + 5<br>Sim + 10<br>Rand + 1<br>Rand + 5<br>Rand + 10 |

For both the evaluation phases, the evaluation method was the leave-one-out, by hiding the last item in the sequence for test and the second-last for validation. We guaranteed that the last item was always the same, whether we were using the shuffled dataset or not. This is a typical method used for evaluating sequence aware recommender systems since the goal is to predict the next best item. The evaluation metrics were the hit ratio (HR) (Equation 5.3) and the Normalized Discounted Cumulative Gain (nDCG) (Equation 5.2) at one, five, and ten. The hit ratio gives us the number of relevant items in a list of recommendations. In this case, the hit ratio will always be one or zero for each user because we only have one relevant item per user; thus, the item is, or it is not in the top@k recommendations. The nDCG is an evaluation method that compares the ideal ranking of a test set (iDCG), with the ranking assigned by the recommendation algorithm (DCG - Equation 5.1), allowing an evaluation regarding the position of the item in the top@k recommendation list.

$$DCG = \sum_{i=1}^{n} \frac{relevance_i}{\log_2(i+1))} \tag{5.1}$$

$$nDCG = \frac{DCG}{iDCG} \tag{5.2}$$

$$HR = 1 - missRatio \tag{5.3}$$

The framework used for the evaluation was the original Tensorflow implementation of Sun et al. [185], available at `https://github.com/FeiSun/BERT4Rec`. The max sequence used in chERMSeq was 50 and in aRMSeq was 100, for the original sequence. For the enriched sequences, the max sequence was $50 + (50 \times n)$ for chERMSeq and $100 + (100 \times n)$ for aRMSeq, where n is the number of similar items added to each original item in the sequence. The models were trained on a NVIDIA Tesla P4 GPU with a batch size of 256.

### 5.3.4 SeEn Item-Item similarity methods

As we already mentioned, different fields depend on different features for calculating the similarity between the items. In the Chemistry case study, we are dealing with chemical compounds. There are several methods for measuring the similarity between chemical compounds, such as structural similarity and semantic similarity. Some studies suggest that the

semantic metrics are better for finding the similarity between the compounds [68, 109, 212]. In [31], the authors used the semantic similarity between the items for creating a hybrid semantic recommender system for chemical compounds. They tested the metrics Resnik [163], Lin [120], and Jiang and Conrath (JC) [96]. The authors also provided an open-access database with more than 128k compound-compound similarity for all the three metrics, which was created using the framework DiShIn[1] [51]. The Lin metric results were one of those that had better results, which is why we are using it in this work.

In the Astronomy case study, for calculating the similarity between the open clusters of stars, we used the features of the Gaia ESA's dataset [154]. Gaia is an astronomical mission with the goal of collecting information about the stars in the Milky Way. The dataset is in the third release, and it has more than 1.9 million stars. We used the stars in Gaia mapped to each open cluster for this work. Then we calculated the mean of the features for each open cluster, and the mean of the features was used for calculating the similarity, using the Cosine similarity (Equation 5.4, where x and y are two non-zero vectors). For the tests presented in this work, we used the features related to the location: longitude, latitude and parallax. The output was a dataset of cluster-cluster similarity with approximately 1.5 million entries.

The code for creating the SeEn datasets is available at:

$$cosine\ similarity(\boldsymbol{x}, \boldsymbol{y}) = \frac{\boldsymbol{x} \cdot \boldsymbol{y}}{||\boldsymbol{x}|| \cdot ||\boldsymbol{y}||} \tag{5.4}$$

## 5.4 Results

In this section, we present first the results for the new sequential datasets created through the LIBRETTI methodology, and second the results for the new sequential enrichment (SeEn) approach. The original LIBRETTI allows to create standard ¡user,item,rating¿ recommendation datasets from scientific domains, where the users are authors from research articles, the items are scientific entities mentioned in the articles, and the ratings are the number of articles where a user mentioned an entity. No timeline is regarded. The new sequence-aware recommendation datasets follow the same user and item approach, however, for each user, the items are ordered by year of publication of the article mentioning the item. The Section Datasets shows how sequence-aware vs non sequence-aware algorithms behave, and also

[1] https://github.com/lasigeBioTM/DiShIn

how sequence-aware algorithms behave when provided with sequential vs non-sequential datasets as input.

The Section SeEn presents the results related to the new recommendation approach which enhances the datasets with the most similar items to the ones the user already interacted with, and uses the new enriched dataset as input to BERT4Rec, a sequence-aware recommendation algorithm. Both parts of this work where tested in the scientific fields of Astronomy and Chemistry.

### 5.4.1 Datasets

In this section, we present the results for the sequence aware recommendation datasets in the fields of Chemistry (for recommending chemical compounds) and in the field of Astronomy (for recommending open clusters of stars). Short examples of both datasets are presented in Table 5.3. The Chemicals Recommendation Matrix sequence (chERMSeq) has as columns user, item, rating and year. The user is an ID assigned by us, corresponding to an author's name. The item is the ID of the chemical compound in the ChEBI ontology. For example, the ID 18357 corresponds to (R)-noradrenaline [1]. The rating is always one, and the year corresponds to the publication year of the article mentioning the chemical compound. The astronomical Recommendation Matrix sequence (aRMSeq) dataset also has the columns user, item, rating and year, and an extra with the item name. This happens because, in this case, the column item corresponds to an ID assigned by us, thus it may be helpful also have the item name.

Table 5.4 shows the statistics of the new datasets. The chERMSeq has fewer ratings and more items than aRMSeq; thus, it is sparser. The aRMSeq dataset has longer sequences than chERMSeq. The minimum size of the sequences for both datasets is 20 to avoid users with few ratings, also known as cold start.

Figure 5.4 presents the distribution of the ratings by each one of the items in chERMSeq and aRMSeq datasets, i.e., the number of users (n users) who rated that specific item. The plots show the typical long-tail phenomenon where a small number of items has the majority of the ratings, whereas a large number of items have only a few ratings. Analysing both plots, despite the similar number of users in the datasets, the aRMSeq dataset concentrates a much larger number of ratings in a small number of items than the chERMSeq dataset. This can be

[1] https://www.ebi.ac.uk/chebi/searchId.do?chebiId=CHEBI:18357

Table 5.3: chERMSeq and aRMSeq examples.

| chERMSeq | | | | |
|---|---|---|---|---|
| User | Item | Rating | Year | |
| 378 | 18357 | 1 | 1984 | |
| 378 | 71045 | 1 | 2010 | |
| 378 | 131855 | 1 | 2015 | |
| 378 | 142842 | 1 | 2016 | |
| aRMSeq | | | | |
| User | Item | Item | Item name | year |
| 25 | 696 | 1 | NGC_2264 | 2005 |
| 25 | 625 | 1 | Melotte_22 | 2011 |
| 25 | 769 | 1 | NGC_2682 | 2013 |
| 25 | 894 | 1 | NGC_6811 | 2020 |

Table 5.4: chERMSeq and aRMSeq datasets statistics.

| Dataset | chERMSeq | aRMSeq |
|---|---|---|
| Total | 131k | 276k |
| Users | 2.5k | 2.7k |
| Items | 16k | 1k |
| Min seq | 20 | 20 |
| Max seq | 783 | 4314 |
| Mean Seq | 53.43 | 101.25 |
| Year range | 1951-2019 | 1998-2020 |
| Type of items | Chemical compounds | Open clusters of stars |
| Sparsity | 99.68 | 90.15 |

better observed in the plot of Figure 5.5, where we present the results for the distribution of the ratings by 1, 5 and 10% of the most rated items. The results show that in the chERMSeq dataset 1% of the items receives 9% of the ratings. In aRMSeq dataset, 1% of the items get 20% of the ratings.

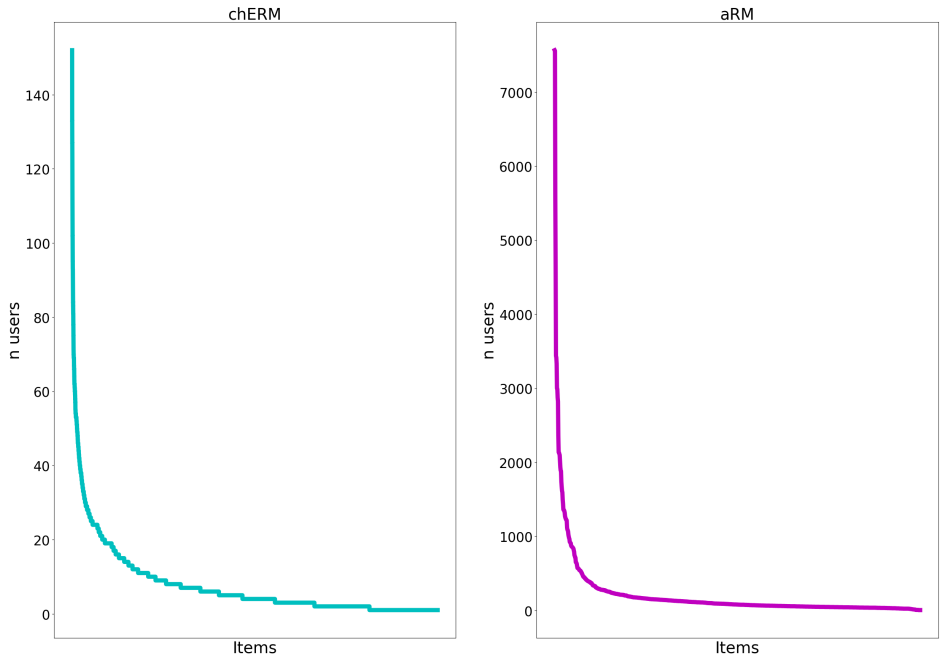


Figure 5.4: chERMSeq and aRMSeq number of users rating each item.

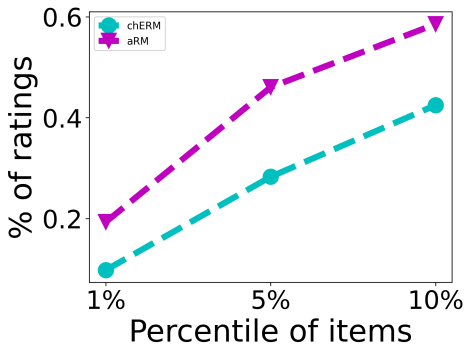


Figure 5.5: Distribution of ratings by percentile of item at 1, 5 and 10%.

Next, we present the results related to the analysis of different recommendation algorithms applied to chERMSeq and aRMSeq datasets (See table 5.1), to evaluate how the use

of sequence aware recommender algorithms, such as BERT4Rec, improve the recommendations of the next best item, and how sequential or not sequential data affect these results.

Figure 5.6 shows the plots for the recommendation algorithms most popular, Alternating Least Squares (ALS), BERT4Rec using non-sequential datasets, and BERT4Rec using sequential datasets. The most popular and the ALS algorithms are CF algorithms and do not consider the sequence of the items. The first recommends always the k items with the most ratings, and the former is a latent factor algorithm based on the similarity of the users. BERT4Rec is a state-of-the-art sequence-aware algorithm, based on neural networks. The algorithms were evaluated using the chERMSeq and the aRMSeq datasets. The evaluation metrics were the Hit Ratio (HR) and the Normalized Discounted Cumulative Gain (nDCG) @ 1, 5 and 10.

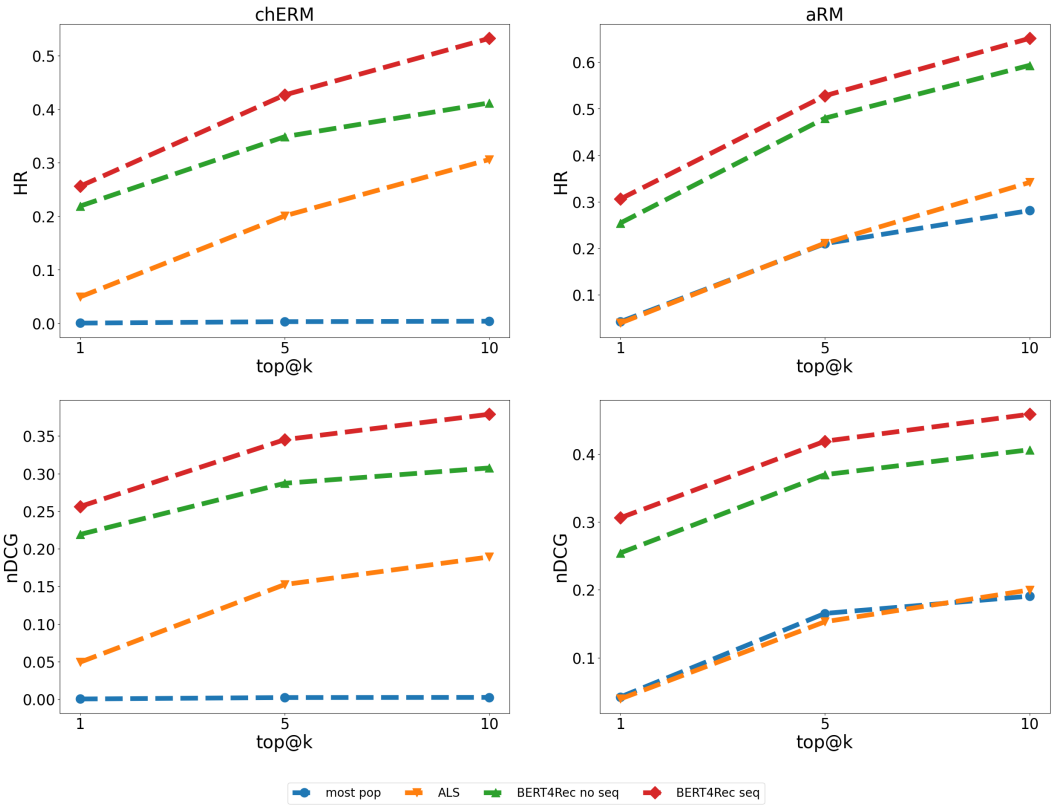


Figure 5.6: Analysis of the results for the recommendation of chemical compounds (chERMSeq) and open clusters of stars (aRMSeq), with the algorithms most pop, ALS, BERT4Rec no seq, and BERT4Rec seq, for the metrics of Hit Ratio (HR) and Normalized Discounted Cumulative Gain (nDCG) @k.

The analysis of Figure 5.6 shows that for the chERMSeq dataset, the algorithm most popular achieved the worst results, as expected, followed by an improvement of more than 20 percentage points for ALS. BERT4Rec surpasses this result when tested with the not ordered sequences. BERT4Rec achieves the best results with the sequence dataset. For the aRMSeq dataset, the most-pop and ALS algorithms achieved similar results, followed by BERT4Rec with the non-sequential dataset and BERT4Rec with the sequential dataset.

### 5.4.2 SeEn

The datasets presented in the previous section have levels of sparsity superior to 90%, which may leads to inferior recommendation results. To improve the quality of the datasets, in this work we developed the SeEn approach.

Figure 5.7 shows a real example of sequential enrichment for a sequence of chemical compounds. In the case presented, the user has three items in the train set, (R)-noradrenaline, bisdemethoxycurcumin, and terretonin, ordered by year. SeEn enriched chERMSeq has added to the sequence the most similar chemical compounds. The goal is to recommend the compound andrastin A (test item).

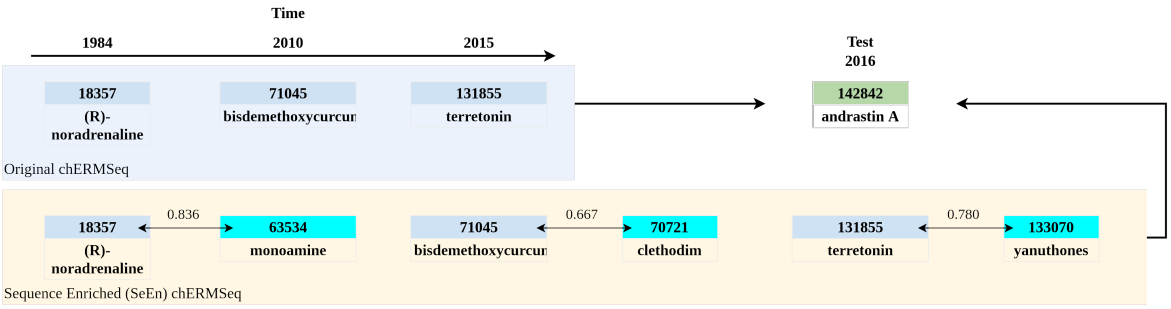


Figure 5.7: Sequential enrichment example. (R)-noradrenaline is 0.836 similar to monoamine, bisdemethoxycurcumin is 0.667 similar to clethodim, and terretonin is 0.780 similar to yanuthones.

Table 5.5 shows the results obtained using BERT4Rec, for the original dataset chERMSeq and aRMSeq, and these datasets with the sequence enriched with the SeEn approach. For the chERMSeq dataset, the Sim_lin + 1 obtained the best results for both HR and nDCG, increasing the original results by approximately seven percentage points, and the results decrease with the increase of n. For the aRMSeq dataset, the best results were achieved when

enriching the sequence with one most similar items, increasing the results by 16 percentage points. In general, the random achieved worse results than the original, proving that introducing similar items is better than introducing random items in the sequence.

Table 5.5: chERMSeq and aRMSeq SeEn results for HR and nDCG @ 1, 5, and 10.

| Dataset | Hit@1 | nDCG@1 | Hit@5 | nDCG@5 | Hit@10 | nDCG@10 |
|---|---|---|---|---|---|---|
| chERMSeq original | 0.2562 | 0.2562 | 0.4268 | 0.3451 | 0.5326 | 0.3789 |
| Sim_lin + 1 | **0.3293** | **0.3293** | **0.4741** | **0.4058** | **0.5560** | **0.4323** |
| Sim_lin + 5 | 0.2828 | 0.2828 | 0.4339 | 0.3611 | 0.5482 | 0.3885 |
| Sim_lin + 10 | 0.1980 | 0.1980 | 0.3090 | 0.2537 | 0.3929 | 0.2806 |
| rand + 1 | 0.2087 | 0.2087 | 0.4097 | 0.3273 | 0.5060 | 0.3584 |
| rand + 5 | 0.2207 | 0.2207 | 0.3532 | 0.2885 | 0.4431 | 0.3174 |
| rand + 10 | 0.1256 | 0.1256 | 0.2036 | 0.1667 | 0.2645 | 0.1863 |
| aRMSeq original | 0.3059 | 0.3059 | 0.5279 | 0.4188 | 0.6513 | 0.4585 |
| Cos + 1 | **0.4680** | **0.4680** | **0.6801** | **0.5809** | **0.7718** | **0.6107** |
| Cos + 5 | 0.2896 | 0.2896 | 0.5417 | 0.4189 | 0.6942 | 0.4680 |
| Cos + 10 | 0.2469 | 0.2469 | 0.4869 | 0.3686 | 0.6330 | 0.4159 |
| rand + 1 | 0.1866 | 0.1866 | 0.2652 | 0.2074 | 0.3348 | 0.2298 |
| rand + 5 | 0.1552 | 0.1552 | 0.2523 | 0.2091 | 0.2726 | 0.2343 |
| rand + 10 | 0.1955 | 0.1955 | 0.3097 | 0.2290 | 0.3579 | 0.2440 |

Figures 5.8 and 5.9 show the loss values for the original chERMSeq dataset, and for the chERMSeq dataset enriched with Sim_lin + 1 items, for the models trained with the BERT4Rec algorithm. The horizontal red line represents the loss equal to 1. Analysing the plots for the loss value, we see that the model trained with the chERMSeq Sim_lin + 1 dataset achieved lower loss values (bellow 1) within less steps (150000 vs 70000 for the original and SeEn, respectively). With this, we conclude that with SeEn we create better models with fewer training steps.

## 5.5 Discussion

To overcome the challenge of the lack of sequence-aware open-access recommendation dataset in scientific domains, in this work we developed a new module for an already existent

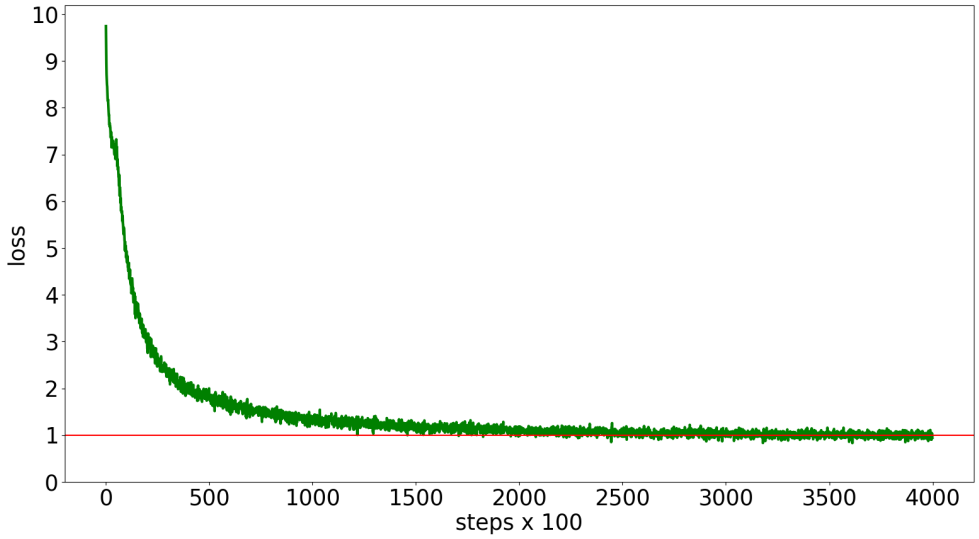


Figure 5.8: Loss for chERMSeq original dataset. Horizontal red line: loss = 1.

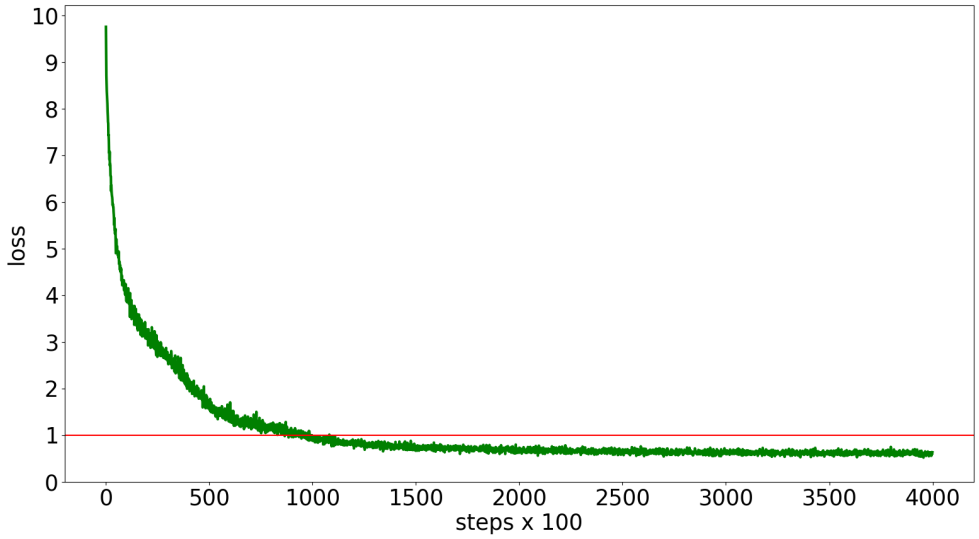


Figure 5.9: Loss for chERMSeq Sim_lin + 1 dataset. Horizontal red line: loss = 1.

method (LIBRETTI) which creates recommendation datasets in scientific fields to create sequence aware recommendation datasets for those areas. We assessed the adaptation of the method in the fields of Chemistry and Astronomy. The results were two datasets, chERMSeq and aRMSeq, for recommending chemical compounds and open clusters of stars, respectively. The larger number of items in the chERMSeq dataset (16k vs 1k) and the similar number of users (2k), results in a sparser rating matrix for chERMSeq (Table 5.4).

The distribution of the ratings by the items follows the typical long tail of the recommendation datasets, with few items having the majority of the ratings (Figure 5.4), which may influence the recommendation algorithms. The plots in Figure 5.6 show how the recommendation algorithms provide evidence that sequence aware algorithms are better for the recommendation of the next best item in chERMSeq and aRMSeq. When evaluating the most popular algorithm in chERMSeq, it got values close to zero, suggesting that the users do not share a large percentage of the most rated items. The behaviour with aRM is different, with the most-pop algorithm achieving results similar to ALS. Looking at Figure 5.5, we can see that the 10% of most popular items have 60% of the ratings in the aRMSeq dataset, against 40% in chERMSeq.

The major goal of this work was to prove that sequence aware recommendation datasets are needed for better next item recommendations. The span of results achieved by the assessed algorithms shows that algorithms not tailored for sequence aware recommendations (most-pop and ALS) perform worse than algorithms designed for sequence recommendations (BERT4Rec), by a margin of more than 20 percentage points. BERT4Rec obtains better results when provided with the datasets with the items ordered by year (chERMSeq and aRMSeq seq) than randomly shuffled. chERMSeq improves the outcomes of BERT4Rec in 12 percentage points in the HR metric and seven percentage points in the nDCG @10. aRMSeq obtained approximately five more percentage points for HR and nDCG@10 than the shuffled version.

Following the evaluation of the datasets, we tested a new approach to address the problem of lack of knowledge into a single sequence. We called sequence enrichment (SeEn) to this approach. SeEn consists of adding to the sequence of items of each user the n most similar items after each item, as exemplified in Figure 5.7. Depending on the field, the method for finding the most similar items will differ due to the specific characteristics and data available. For the case study in Astronomy, we used the cosine similarity, and for the case study in Chemistry, we used the semantic similarity with the metric Lin. Comparing the

results presented in Table 5.5, for both HR and nDCG evaluation metrics, the SeEn datasets enriched with +1 most similar item obtained better results than the original dataset. With these results, we may conclude that there is an optimal number of similar items that increase the results of the models. After that number, the entropy introduced into the sequences leads the models to less accurate predictions.

Measuring the advantages and disadvantages of the SeEn approach seems to improve the results of BERT4Rec, recommending the right next item in the first position of the list of recommendations. We believe this is because it provides the recommendation of new items, for example, if we are trying to recommend an item from this year, if it does not exist in the original dataset, it will never be recommended when the model is trained with the original datasets. A possible disadvantage of this approach is the increase in the size of the sequence, which is a problem for algorithms such as BERT4Rec, with a computational complexity of O(n2d), quadratic with the length n.

Observing the results presented in this study, we may conclude that there is a need for sequence recommendation datasets in scientific items. The enrichment of these datasets leads to better results in BERT4Rec, a state-of-the-art recommendation algorithm.

## 5.6 Conclusions

Sequence-aware Recommender Systems (RS) are still a challenge for the existing approaches. The goal of this work was to improve state-of-the-art sequence-aware recommender algorithms. To that end, we developed a new approach based on sequential enrichment (SeEn), consisting in introducing the most similar items into a sequence. The SeEn approach was evaluated on datasets from two distinct scientific fields, Chemistry and Astronomy, and the preliminary results showed that the BERT4Rec algorithm provides better results with the enriched sequences than with the original sequences, for the same users.

For future work, we intend:

- Understand if the recommended items with SeEn are more similar to the next relevant item for each user;
- Test the SeEn datasets with algorithms which consider the rating of each item and not only the sequence of items;

- Add not only the most similar items, but also filtering the items by year;
- Train the BERT4Rec models with fix masks, instead of random.

The last point may be beneficial to the recommendations because the training sequences will be original-item-1 →sim-item-1 →sim-item-n →original-item-2. The system will train the prediction to the original-item-2.

# 6

# COVID-19: A Semantic-Based Pipeline for Recommending Biomedical Entities

This Chapter answer to the Research Question 4: Will the use of multiple ontologies in the creation of the recommendation dataset in scientific fields improve the performance of state-of-the-art Collaborative-filtering (CF) algorithms, in particular when comparing with datasets with only one ontology? and it corresponds to the paper: *Barros, M.; Lamurias, A.; Sousa, D., Ruas; P., Couto, F. M; (2020, December). COVID-19: A Semantic-Based Pipeline for Recommending Biomedical Entities. In Proceedings of the 1st Workshop on NLP for COVID-19 (Part 2) at EMNLP 2020.*

With the increasing number of publications about COVID-19, it is a challenge to extract personalized knowledge suitable for each researcher. This work aims to build a new semantic-based pipeline for recommending biomedical entities to scientific researchers. To this end, we developed a pipeline that creates an implicit feedback matrix based on Named Entity Recognition (NER) on a corpus of documents, using multidisciplinary ontologies for recognizing and linking the entities. Our hypothesis is that by using ontologies from different fields in the NER phase, we can improve the results for state-of-the-art collaborative-filtering recommender systems applied to the dataset created. The tests performed using the COVID-19 Open Research Dataset (CORD-19) dataset show that when using four ontologies, the results for precision@k, for example, reach the 80%, whereas when using only one ontology, the results for precision@k drops to 20%, for the same users. Furthermore, the use of multi-fields entities may help in the discovery of new items, even if the researchers do not

have items from that field in their set of preferences.

## 6.1 Introduction

The research literature is the main form of dissemination for scientific works, growing by the minute. Platforms such as PubMed[1], account for more than 30 million articles related to biomedical literature. The emergence of new topics of investigation with particular interest for modern society, such as COVID-19 [142], leads to an even faster increase in the publication rate. The scientific literature contains vast and essential information about biomedical entities engaged in COVID-19 processes. However, it is difficult for the researchers to read all the papers and keep up with all the new topics suitable for their research.

Given the importance of COVID-19 related topics, the Allen Institute for AI, in collaboration with The White House Office of Science and Technology Policy, the National Library of Medicine, the Chan Zuckerburg Initiative, Microsoft Research, and Kaggle, collected and released the first version of COVID-19 Open Research Dataset (CORD-19)[2] [211]. The main goal of this dataset is to help in the development of new tools for the extraction of relevant information in the fight of COVID-19 disease.

One of the main techniques applied to extract information from the research literature is Named Entity Recognition (NER), followed by Relation Extraction (RE). NER consists of recognizing entities mentioned in the text by identifying the offset of their first and last character. There is an extensive work done on biomedical NER, regarding all type of entities, such as chemicals [109] and human phenotypes [122]. RE aims to identify a relation between entities mentioned in a given document or text window. Regarding biomedical RE, the research is focused not only on extracting but also on classifying the relationship between biomedical entities, ranging from phenotype-gene relations [182] to chemical-chemical interactions [86].

Recommender Systems (RS) are tools which allow recommending items of interest to a user, based on the similarity between her/his preferences and the preferences of other users - Collaborative-filtering (CF), or based on the similarity of the items this user already liked - Content-based (CB). Hybrid RS may be created to solve intrinsic challenges of the previous

[1] https://pubmed.ncbi.nlm.nih.gov/

[2] https://www.kaggle.com/allen-institute-for-ai/CORD-19-research-challenge

RS approaches, such as the cold start problem for new items in CF and new users in both CF and CB. Cold start refers to a new user without any item rated or a new item without any rate by the users. The users' ratings may be explicit, for example, using a star classification system, or implicit, in which case the ratings are inferred from the user's interaction with the items, for example, by buying or seeing them. The implicit rating may be binary, for example, 1 if a user saw a movie, 0 if she/he did not see a movie, or it may be a different measure, for example, the duration that a user watches a video [165].

RS are most frequently used for recommending items such as movies, books, or e-commerce products. RS approaches have been applied most recently to recommend the most appropriate research items for each researcher [29, 30, 143]. These efforts consist mostly of developing recommendation datasets of implicit feedback for various scientific fields, such as Chemistry and Artificial Intelligence, developed by mining the scientific literature. The goal of these datasets is to recommend scientific items for the research, for example, Chemical compounds, based on their past interests and their peers' interests.

This work proposes a novel pipeline for extracting, relating and recommending scientific items from CORD-19 dataset, using entities from various ontologies: Gene Ontology (GO)[1], Disease Ontology (DO)[2], Human Phenotype Ontology (HP)[3], and Chemical Entities of Biological Interest Ontology (ChEBI)[4]. The use of ontologies for the NER phase allows the extraction of the entities from the text and the linking of the entities to a definition, avoiding the ambiguity of the terms. We selected these ontologies for their importance in the COVID-19 disease. With these, we may find drugs, genes, phenotypes and diseases related to COVID-19, which may guide the researchers in the discovery of new information for stopping the disease.

The main contributions of this work are:

- A dataset of 9k articles automatically annotated with relevant items/concepts for CORD-19;
- A sample dataset curated for CORD-19;
- A sample dataset with relations between the entities of the four ontologies;

[1] http://geneontology.org/
[2] https://disease-ontology.org/
[3] https://hpo.jax.org/app/
[4] https://www.ebi.ac.uk/chebi/

- An implicit feedback matrix based on the previous datasets.

The source code for this work is fully available at:
`https://github.com/lasigeBioTM/knowledge-extraction-from-CORD-19`.

## 6.2 Related work

Despite the novelty of CORD-19 dataset, the number of works being published using this dataset increases by the day. CORD-19 is being used for developing tools in various fields. [189] created a dataset for Question and Answering about COVID-19. [107] developed a pipeline for creating a dataset based on biomedical NER, for chemicals, diseases, genes and species, using TaggerOne and GNormPlus tools. [230] focused on a search engine for CORD-19 based on neural networks, the Neural Covidex. [213] created the tool EVIDENCEMINER, which allows a user to introduce a sentence in natural language and to retrieve an evidence for that statement. They applied the EVIDENCEMINER to CORD-19. [214] created the CORD-NER dataset, a dataset with entities from 75 fields, including genes, chemicals, diseases, and specific entities related to COVID-19, for example, coronaviruses, viral proteins, evolution, materials, substrates and immune responses. [158] is developing a personalized exploratory search system for COVID-19, based on CORD-19, the CovEx. According to the authors, the system allows the user to search for keywords, recommending other keywords and research articles. The recommended keywords are extracted only from the title and abstract, using Bi-LSTM-CRF technique.

Not related to CORD-19, other research works created recommendation datasets for scientific fields. [143] created a recommendation dataset of implicit feedback for the field of artificial intelligence with the format of <article,topic,cardinality>. Their work extracts topics related to artificial intelligence from articles and the cardinality is calculated according to the importance of the topic in the article. Then, the dataset is used for recommending topics and articles. The topics are extracted from the research articles using text mining techniques based on the articles' token frequency. They do not use NER techniques.

In [29], the authors developed a methodology called LIBRETTI to create recommendation datasets of implicit feedback for scientific fields, for recommending scientific entities, such as clusters of stars and Chemicals compounds. The methodology consists of given a

list of scientific entities, finding articles related to these entities, and extracting the authors. The dataset has the format of <author,entity,rating>. The ratings are the number of articles a unique author wrote about an entity. This work uses the CHEBI ontology to extract the list of entities for a dataset of Chemical compounds; nevertheless, it is limited to this ontology and performs neither NER nor RE.

The objective of our pipeline is to create a tool for performing NER of multiple scientific fields in the CORD-19 dataset, followed by RE, and the creation of a recommendation dataset of implicit feedback based on LIBRETTI methodology, to recommend entities of different fields to the users/researchers. We hypothesize that using multiple ontologies in the creation of the recommendation dataset leads to an improvement in the performance of state-of-the-art CF algorithms, in particular when comparing with datasets with only one ontology.

## 6.3 Methodology and Experiments

The general workflow of the proposed pipeline is represented in Figure 6.1.

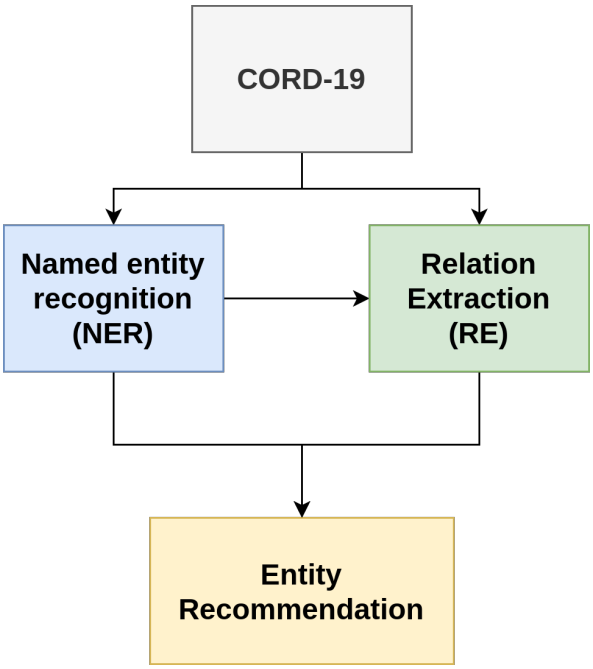


Figure 6.1: General pipeline.

The input of the pipeline is a dataset of research articles. First, we apply NER techniques

for recognizing relevant entities in the text, using ontologies for linking the entities. Second, we extract the relations between the entities (RE). And third, we create an implicit feedback matrix and apply RS algorithms for recommending the recognized entities.

The dataset used in this work is the COVID-19 Open Research Dataset (CORD-19)[211], more concretely, the 2020-03-13 version. The total number of documents is 29,500. For this work, we used the commercial use subset of 9,000 full-text documents.

### 6.3.1 Named Entity Recognition

To obtain ontology concepts from the CORD-19 dataset, we used the MER tool [52], which can identify entities in text from any ontology. We selected four biomedical ontologies to use as lexicons with MER so that we could identify those concepts in the texts: GO, HPO, DO and ChEBI. We used the latest version of each ontology available in June 2020. MER has the advantage of performing Named Entity Recognition and Entity Linking simultaneously, so we could obtain directly the reference ontology URI, which is necessary for the RS. Furthermore, it does not require annotated training data to identify new entity types.

The MER tool indexes the ontology labels and synonyms and uses regular expressions, therefore being limited in terms of what expressions it can identify. To assess the annotation quality, we manually evaluated a sample of 90 paragraphs from the CORD-19 dataset with four experts. The paragraphs were randomly selected from a pool of 100 documents, including paragraphs from the abstract, body and figure and table captions with at least five entities annotated by MER. We asked the expert annotators to verify the automatic annotations, modify and delete them, and add new annotations if necessary. Of the 90 paragraphs, 10 were annotated simultaneously by the four annotators to calculate the Inter-Annotator Agreement, using Fleiss' kappa [132]. Afterwards, we calculated Precision (Equation 6.1), Recall (Equation 6.2) and F1-score (Equation 6.3) metrics on the automatic annotations of these 90 paragraphs, given by the following formulas:

$$P = \frac{TP}{TP + FP} \tag{6.1}$$

$$R = \frac{TP}{TP + FN} \tag{6.2}$$

$$F1 = \frac{2PR}{P+R} \tag{6.3}$$

where TP corresponds to the total of True Positives (entities identified correctly), FP corresponds to False Positives (entities identified incorrectly) and FN corresponds to False negatives (entities that should have been identified).

We created a consensus corpus by merging the annotations of all 4 annotators. On the 10 overlapping paragraphs, we accepted each annotation if two or more annotators agreed on it. We calculated micro scores by adding the TP, FP and FNs of every paragraph and using Equations 6.1,6.2, and 6.3, and macro scores, which were the average of the Precision, Recall and F1-scores of all paragraphs.

### 6.3.2 Relation Extraction

We took initial steps towards COVID-19-related relation extraction training data (RE), providing a small sample dataset of ten documents, with all possible relationships between the four types of entities identified by our NER pipeline. Thus, we were able to establish ten different types of relations, encompassing the four ontologies (ChEBI, DO, HPO, and GO) in two machine-readable formats (XML and TSV), following previous works by Herrero-Zaro et al. [86] and Li et al. [113], respectively.

### 6.3.3 Recommender System

For the creation of the RS dataset, we used a methodology called LIBRETTI [29]. The methodology consists of creating datasets with the standard format of <user, item, rating>. The items are scientific entities and the users are authors from research articles, where these items are mentioned. The items may be obtained, for example, from a list or, as in previous work, from an ontology [29, 30]. As previously mentioned, for this work we used items/entities from four distinct ontologies: chemical compounds from CHEBI, functions of genes from GO, phenotypic abnormalities from HP, and diseases from DO. The output from this phase is a recommendation dataset of <user, item, rating>, where the users are authors from research articles, the items are entities from CHEBI, GO, HP or DO, and the ratings are the number of articles an author wrote about an entity. Our goal was to evaluate if using more ontologies, i.e., increasing the number of entities for each author, the results of the

recommendation algorithms are better for the same users. Thus, we assessed the results in the dataset with the items from all the ontologies, and with the items of each ontology alone, for the same group of users. We consider as baseline the results obtained with the datasets containing only items from each single ontology.

The recommendation datasets are evaluated using offline evaluation methods for the quality of the recommended ranked list of items [174]. From the vast range of metrics for evaluate ranked lists, we selected Precision@k (Equation 6.4), Recall@k (Equation 6.5) and Mean Reciprocal Rank (MRR)@k (Equation 6.6), where k is the size of the recommendation list.

$$Precision@k = \frac{relevant\_items@k}{k} \tag{6.4}$$

$$Recall@k = \frac{relevant\_items@k}{total\_relevant\_items} \tag{6.5}$$

$$MRR = \frac{1}{n\_users} \sum_{i=1}^{n\_users} \frac{1}{rank\_i} \tag{6.6}$$

Precision@k is a measure of the relevant items recommended in the top@k list, recall@k the number of relevant items recommended in the top@k list, and MRR evaluates in which position the first relevant item appears. All evaluation metrics range between 0 and 1, with the best values being closest to 1.

Since the dataset consists of ratings obtained by implicit feedback, we selected Alternating Least Squares (ALS)[1] [90], a recommendation algorithm capable of dealing with implicit feedback datasets. ALS is a latent factor algorithm that addresses the confidence of a user-item pair rating. The ALS goal is to minimize the least-squares of the rating matrix and the matrix resultant from the dot product of the user matrix and item matrix. ALS is also suitable for recommending ranked lists of items. This algorithm was already used in similar datasets with positive results, for recommending Chemical Compounds [30]. The datasets for the evaluation were split in 80% of users and items for training and 20% for testing.

[1] `https://implicit.readthedocs.io/en/latest/index.html`

## 6.4 Results and Discussion

### 6.4.1 Named Entity Recognition

The entity annotation part of the pipeline obtained a total of 2,412,671 entity mentions on the comm_use_subset of the CORD-19 dataset (9k documents). Table 6.1 shows the counts of the entities obtained according to each ontology.

Table 6.1: Statistics of the entities obtained on the CORD-19 commercial subset of 9k documents.

| Ontology | Total mentions | Unique mentions |
|---|---|---|
| CHEBI | 1,302,219 | 6,693 |
| GO | 484,266 | 3,258 |
| DO | 314,959, | 1,726 |
| HP | 311,227 | 1,774 |
| Total | 2,412,671 | 13,451 |

We obtained an average of 268.07 entity mentions per document and 67.02 unique concepts per document. The results of our manual evaluation are provided in Table 6.2. Our gold standard of 90 paragraphs obtained an IAA of 0.2978, which indicates fair agreement, according to [111]. However, if we do not take into consideration the ontology URIs, this agreement rises to 0.3760. This indicates that the definition of the URIs was a source of ambiguity and the annotators did not always agree on what was the best ontology concept for a named entity. The Precision, Recall and F1-score values obtained indicate that the entities were mostly correctly identified, with a relatively high macro and micro Recall value. It is also possible to observe that the highest F1-scores obtained were with the gold standard where at least two annotators had to agree to accept an annotation.

Table 6.2: Results of the manual evaluation of the NER module. Min Votes corresponds to the number of annotators necessary to agree on the gold standard annotations.

| | Micro | | | Macro | | |
|---|---|---|---|---|---|---|
| Min Votes | P | R | F1 | P | R | F1 |
| 1 | 0.7740 | 0.8007 | 0.7871 | 0.7671 | 0.827 | 0.7601 |
| 2 | 0.7656 | 0.8211 | 0.7924 | 0.761 | 0.8411 | 0.7641 |
| 3 | 0.7586 | 0.8255 | 0.7907 | 0.7532 | 0.8423 | 0.7616 |
| 4 | 0.7457 | 0.8374 | 0.7889 | 0.7408 | 0.8542 | 0.7579 |

The positive results obtained in this phase of the pipeline allows us to use the NER dataset for subsequent Relation Extraction, and for the creation of the recommendation dataset.

### 6.4.2 Relation Extraction

To accomplish the RE dataset, we only considered relations between entities in the same text portion, following the original dataset, identified common NER errors to exclude those entities from participating in relations, and did not consider relations between the same entities in different places of the text portion. The resulting final counts are presented in Table 6.3.

Table 6.3: Statistics for the relation extraction sample dataset possible relations.

| Pair | Count |
|---|---|
| GO-GO | 410 |
| GO-CHEBI | 765 |
| GO-HP | 396 |
| GO-DO | 342 |
| CHEBI-CHEBI | 489 |
| CHEBI-HP | 457 |
| CHEBI-DO | 349 |
| HP-HP | 242 |
| HP-DO | 440 |
| DO-DO | 149 |
| Total | 4,039 |

Following, we present Examples 6.4.1 and 6.4.2 of relations extracted from text.

**Example 6.4.1.** *Several viruses use classical receptors and transmembrane proteins that are widely represented in cells and are not restricted to the monocyte/macrophage population, such as nucleolin by the respiratory syncytial virus [75] ; sialic acid sugars by the influenza virus [76], mouse hepatitis virus [77] and Theiler's murine encephalomyelitis virus [78]; and phosphatidylserine by the vesicular stomatitis virus [79].*

*sialic acids (CHEBI_26667) - influenza (DOID_8469)*

*sialic acids (CHEBI_26667) - hepatitis (HP_0012115)*

*sialic acids (CHEBI_26667) - hepatitis (DOID_2237)*

*sialic acids (CHEBI_26667) - encephalomyelitis (DOID_640)*

*phosphatidylserine (CHEBI_18303) - stomatitis (DOID_9637)*

*phosphatidylserine (CHEBI_18303) - stomatitis (HP_0010280)*

**Example 6.4.2.** *For receptor-mediated entry, viruses can employ both nonspecific receptors, where a virus accesses a broad range of cell populations, or highly specific interactions between the virus and cell surface receptors, where a virus infects a limited set of target cells; this determines the tropism of viral infection.*

*cell surface (GO_0009986) - tropism (GO_0009606)*

*tropism (GO_0009606) - viral infection (DOID_934)*

*tropism (GO_0009606) - viral infection (GO_0016032)*

In both Examples 6.4.1 and 6.4.2, we present sentences from a research article and the respective relations between the entities extracted from these sentences. For example, in Example 6.4.1 we identified relations between the chemical compound *sialic acids* (CHEBI_26667) and the disease *influenza* (DOID_8469).

We believe this to be an initial step towards COVID-19-related relation extraction training data. The following step should be running existing machine learning models [181] to classify the possible relations as true or false. Finally, we intend to use the RE information as data for the RS detailed below.

### 6.4.3 Recommender System

From the methodology presented in Section 6.3.3, we obtained a recommendation dataset for recommending the biomedical entities in CORD-19, previously extracted with NER (Section 6.3.1), called CORD-19 Recommendation Dataset (CORD-19-RD). CORD-19-RD was assessed for the items in all four ontologies (CORD-19-RD-all), and sampled by ontology, in order to evaluate how the use of items from the four ontologies influences the results when compared to the individual ontologies. Thus, we have the sampled datasets CORD-19-RD-chebi, CORD-19-RD-go, CORD-19-RD-hp and CORD-19-RD-do. Table 6.4 shows the statistics of CORD-19-RD and its samples, presenting the number of users, items and ratings, the sparsity of each dataset, the maximum and minimum rating values ( max and min), and also the mean of items by user (mItems), and the mean of users by item (mUsers).

The number of users in the various datasets remains almost the same, with the highest variation in the CORD-19-RD-go. The number of items decreases drastically from CORD-19-RD-all to CORD-19-RD-do, with a variation of 11.644 items. The number of ratings also decreases from CORD-19-RD-all to CORD-19-RD-do. Despite the decrease in the number

Table 6.4: Statistics for the dataset CORD-19-RD-all, CORD-19-RD-chebi, CORD-19-RD-go, CORD-19-RD-hp and CORD-19-RD-do.

| Dataset | Users | Items | Ratings | Sparsity | max | min | mItems | mUsers |
|---|---|---|---|---|---|---|---|---|
| CORD-19-RD-all | 45,401 | 13,353 | 3,888,870 | 0.993 | 39 | 1 | 85.6 | 291.2 |
| CORD-19-RD-chebi | 45,401 | 6,642 | 1,939,568 | 0.993 | 39 | 1 | 42.7 | 292.0 |
| CORD-19-RD-go | 44,646 | 3,250 | 800,135 | 0.994 | 31 | 1 | 17.9 | 246.1 |
| CORD-19-RD-hp | 45,343 | 1,752 | 684,332 | 0.991 | 37 | 1 | 15.1 | 390.6 |
| CORD-19-RD-do | 45,041 | 1,709 | 464,835 | 0.993 | 39 | 1 | 10.3 | 271.9 |

of items, the sparsity is not affected. Notwithstanding that, the mean of items by user is much higher for CORD-19-RD-all.

Figure 6.2 shows the results of applying the ALS algorithm to the different datasets CORD-19-RD-all, CORD-19-RD-chebi, CORD-19-RD-go, CORD-19-RD-hp and CORD-19-RD-do, for Precision@k, Recall@k, and MRR@k, with k varying from 1 to 20, with steps of 1.

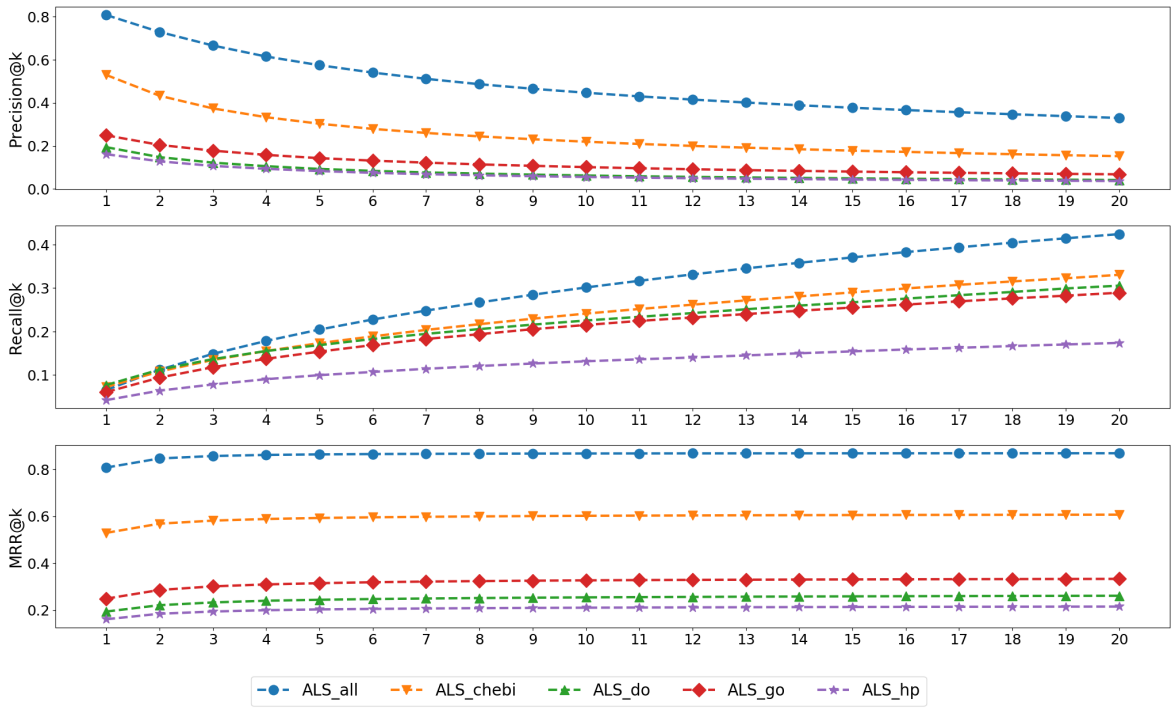


Figure 6.2: Results of the algorithm ALS for Precision@k, Recall@k and MRR@k, applied to CORD-19-RD-all, CORD-19-RD-chebi, CORD-19-RD-go, CORD-19-RD-hp and CORD-19-RD-do.

Analyzing Figure 6.2, ALS best performs in CORD-19-RD-all dataset for all the evaluation metrics. Looking at these results and the values in Table 6.4, we can see the relation between the mean of items by user and the results of ALS. The higher the mean of items by user, the higher the results for all the evaluation metrics for the same number of users. The presented results prove our hypothesis that items from several ontologies, i.e., from more than one field of Science, improve the results of state-of-the-art recommendation algorithms. Our pipeline solves the lack of ratings and allows the recommendation of items from various fields, allowing the development of multidisciplinary RS. For the COVID-19 case study, we will be able, for example, to recommend chemical compounds to some user interested in the disease, which may increase the study of new drugs, that otherwise would be much harder to find the connection.

Table 6.5 shows an example of the top@20 recommendation for a user in the CORD-19-RD-all. The recommendation algorithm recommends items from all the ontologies for this user, even though she/he does not have all the ontologies represented in the training set. This may lead to the discovery of new diseases similar to COVID-19, and chemicals used in the treatment of those diseases, which may be an object of study for its use in COVID-19.

The next step is to use the relation extracted in the RE phase to recommend the relations between the items and create an explainable RS. For example, we could use the relations extracted in Examples 6.4.1 and 6.4.2 for creating knowledge graphs and recommending items from different fields related to the articles.

We still need to understand if all the entities are suitable for being recommended. For example, the entity DOID_4 (Disease) is one of the most identified in the NER phase, consequently being one of the most recommended in the recommendation phase. However, is it relevant for a user the recommendation of "disease"? We are now studying new methods for assigning relevance to each entity. Additionally, we can extend this approach to other documents from the CORD-19, as new versions are released.

The code for this work is fully available at: `https://github.com/lasigeBioTM/knowledge-extraction-from-CORD-19`.

## 6.5 Conclusion

Given the growing number of publications, this work's goal was to develop a pipeline for extracting biomedical entities from scientific literature, finding the relations between

Table 6.5: Example of recommendation for a user in the CORD-19-RD-all. The green cells are the relevant items recommended.

| Ontology ID | Name |
|---|---|
| CHEBI_17076 | streptomycin |
| GO_0019012 | virion |
| CHEBI_149681 | methcathinone |
| HP_0001903 | Anemia |
| CHEBI_25212 | metabolite |
| CHEBI_17234 | glucose |
| GO_0019079 | viral genome replication |
| CHEBI_55308 | poly(2,5-furan) macromolecule |
| DOID_0050639 | primary cutaneous amyloidosis |
| CHEBI_15366 | acetic acid |
| CHEBI_33601 | safranin O |
| CHEBI_53233 | 3-(4,5-dimethylthiazol-2-yl)-2,5-diphenyltetrazolium bromide |
| CHEBI_15756 | hexadecanoic acid |
| DOID_3247 | rhabdomyosarcoma |
| GO_0006631 | fatty acid metabolic process |
| HP_0040280 | Obligate |
| GO_0005886 | plasma membrane |
| CHEBI_17544 | hydrogencarbonate |
| CHEBI_32952 | amine |
| HP_0030078 | light-harvesting complex, core complex |

them, and recommending entities of interest for a particular researcher. The second goal was to prove that by using ontologies from different science fields, CF RS would achieve better results when recommending ranked lists of entities to the users. We used as a case study the CORD-19 dataset, which is a dataset in a field of high relevance for this Era. Using this dataset, we performed NER using four ontologies, CHEBI, DO, HP, and GO, creating an annotated dataset of 9k documents. We also curated 100 documents from this dataset, achieving positives results for precision, recall, and F1-score. The RE phase is in the beginning. Nevertheless, this is a first step for creating a full dataset of relations between the fields in study, which can then be used for generating a knowledge base for COVID-19. We created a dataset with more than 3 million ratings, 45 thousand users, and 13 thousand items from four relevant scientific fields in the recommendation phase. We concluded that

using items from several fields for the same users, the CF algorithm reached better results. For future work, we intend to increase the number of research documents, the number of documents manually annotated, and provide a better baseline. Furthermore, the next step is to integrate the RE dataset in the RS. It is also important to perform online tests for a better understanding of the relevance of recommended items. For such, an online recommendation platform will be developed.

# 7
# Conclusions and Future Work

## 7.1 Conclusions

Recommender Systems (RS) have been broadly used in many fields, from movies to e-commerce and books, scientific papers, news and social networks. In scientific fields, RS are not used so often. We identified as a major challenge for the use of RS in scientific fields the lack of open-source recommendation datasets for testing and evaluating recommendation algorithms. This thesis's main goal was to study how the use of RS could help researchers to find new scientific entities of interest. The first step of the work was to find a solution for the lack of open-source recommendation datasets in scientific fields. As the major reliable source of Science is the research literature, the first Research Question aimed to answer was: May the use of research literature mitigate the lack of recommendation dataset for developing, testing and evaluating recommendation algorithms in scientific fields?

To answer RQ1, this thesis developed a methodology that explores scientific literature for generating utility matrices of implicit feedback. This methodology, called LIBRETTI (LIterature Based RecommEndaTion of scienTific Items), consists in identifying a list of items, finding research articles related to them, extracting the authors from each article, and finally creating a dataset where users are unique authors from the collected articles. The rating values are the number of articles a unique author wrote about an item. LIBRETTI was assessed in two distinct case studies, Astronomy and Chemistry. In the case study in Astronomy, the items were open clusters of stars, and we used two knowledge sources for extracting the articles linked to the open clusters of stars, Simbad and ADS. In the Chemistry

case study, the items were chemical compounds, and the study used the Chemical Entities of Biological Interest (ChEBI) ontology as the source of items and to find the articles linked to the chemical compounds. The contributions were two recommendation datasets, ARM and cheRM, for the case study in Astronomy and Chemistry, respectively. In the study, we compared these datasets with one of the most used recommendation datasets, the ML-100k, and with the SD4AI, a dataset also created from scientific literature, but for recommending articles and research topics. According to the results, one may conclude that research literature may be used as a source for creating reliable recommendation datasets in scientific fields. The datasets created through LIBRETTI were then used to answer research questions related to scientific fields and recommender systems.

The second research question of this thesis was: does the use of semantic similarity between the Chemical Compounds calculated through ontologies for creating a Content-based (CB) algorithm improve the results of state-of-the-art collaborative-filtering algorithms for implicit feedback recommendation datasets?

To answer RQ2, one developed an approach that consists of a hybrid recommender model suitable for implicit feedback datasets and focused on retrieving a ranked list according to the relevance of the items. The model integrates collaborative-filtering algorithms for implicit feedback (Alternating Least Squares and Bayesian Personalized Ranking) and a new content-based algorithm, called ONTO, using the semantic similarity between the chemical compounds in the ChEBI ontology. The algorithms were assessed on an implicit dataset of chemical compounds, CheRM-20, developed according to the LIBRETTI methodology, with more than 16.000 items (chemical compounds). The hybrid model improved the collaborative-filtering algorithms ALS and BPR results by more than ten percentage points in most of the assessed evaluation metrics. The obtained results allow answering affirmatively to RQ2 since the results for the Hybrids algorithms are higher when compared with the individual algorithms.

Until this point, the recommendations do not consider the timeline of the interactions between users and items. However, especially in Science, time is a key term when recommending an entity. Thus, the third research question of this thesis was: Will the semantic enrichment of sequences of items with the n most similar items improve the results of state-of-the-art sequence-aware recommendations algorithms?

To answer RQ3, this thesis developed a hybrid approach between Collaborative-filtering (CF) deep learning methods and the CB methods. The approach is called sequential enrich-

ment (SeEn), and it consists in adding to a sequence of items the n most similar items after each original item. The new sequence is then passed as input for state-of-the-art sequence-aware recommendation algorithms (BERT4Rec) with the goal of improving the results when compared with the original sequence. SeEn was tested in two datasets in the fields of Chemistry, where the items are chemical compounds, and Astronomy, where the items are open clusters of stars. For the chemical compounds, the similarity was assessed using the semantic similarity calculated using the ChEBI ontology. For the open clusters of stars, the similarity was calculated by mapping the open clusters of stars to the Gaia dataset and using the features of the stars in Gaia. Confirming the hypothesis, the models trained with the enriched datasets achieved better results in the evaluation than the models trained with the original dataset. The models trained with the Chemistry dataset obtained an increase of seven percentage points and the models trained with the Astronomy dataset improved by 16 percentage points.

Ontologies seem to be a key in the recommendation of scientific entities. Thus, the fourth research question of this thesis was: Will the use of multiple ontologies in the creation of the recommendation dataset in scientific fields improve the performance of state-of-the-art CF algorithms, particularly when comparing with datasets with only one ontology?

To answer RQ4, one used as a case study the COVID-19 disease, given its worldwide importance and the rapid growth in the research literature. Thus, to respond to RQ4, one builds a new semantic-based pipeline for recommending biomedical entities to scientific researchers. The pipeline consists of performing Named Entity Recognition (NER) on a corpus of documents related to COVID-19, using multidisciplinary ontologies to recognize and link the entities. The evaluation performed using the COVID-19 Open Research Dataset (CORD-19) dataset shows that when using four ontologies, the results for precision@k, for example, reach 80%, whereas when using only one ontology, the results for precision@k drops to 20%, for the same users. The results answer positively to RQ4, i.e., multi-field entities improve the recommendations' outcomes. It allows the recommendation of new items even if the researchers do not have items from that field in their set of preferences.

All the studies presented in this thesis seem to point in the direction that using recommender systems approaches help researchers to find new items of interest, fulfilling the main goal of the thesis.

## 7.2 Future work

The use of recommender systems in scientific fields is a poorly explored field. With this thesis, we hope that more work arises on the topic since it was already proved the value of recommender systems in discovering new knowledge.

LIBRETTI may be applied to other fields besides the ones assessed in this work. For example, it may be used for developing datasets in the fields of genes and proteins, allowing the recommendation of these entities to the researchers for further studies. The ONTO algorithm may be tested with other similarity metrics and may be used with other ontologies. The SeEn approach, given the flexibility of this method, has several unexplored hypotheses, such as how the enriched sequences will behave with other recommendation datasets, deep and non-deep learning, how adding the most similar and the less similar will interfere with the recommendation, and how will BERT4Rec respond if the random masks were modified to fixes masks.

The methods used for the CORD-19 dataset, related to COVID-19, may be applied to other diseases and with other ontologies and knowledge sources. Overall, the methods developed and described in this thesis have the potential to be applied to all the fields regarding that there is research literature and items.

Another approach that was not addressed in this thesis was the online evaluation of the recommendation methods. A possibility for future work is to integrate RS in an online platform where to implement recommendation algorithms for scientific fields to perform A/B testing, and understand the impact of these algorithms in the real world.

# Acronyms

**ADS** SAO/NASA Astrophysics Data System. 36, 47

**ALS** Alternating Least Squares. 15, 79, 130

**ANN** Artificial Neural Networks. 15

**BPR** Bayesian Personalized Ranking. 15, 79

**CB** Content-based. 2, 5, 12, 16, 45, 73, 124, 140

**CF** Collaborative-filtering. 2, 5, 6, 12, 13, 15, 17, 18, 20, 21, 41, 45, 74, 75, 77, 79–84, 96–98, 100, 123, 127, 140, 141

**CFMF** Collaborative-filtering Matrix Factorization. 15

**ChEBI** Chemical Entities of Biological Interest. 6, 18, 47, 75, 140

**DO** Disease Ontology. 7, 18, 75

**GO** Gene Ontology. 7, 18, 75

**HPO** Human Phenotype Ontology. 7

**IMDB** Internet Movie Database. 11, 21

**JC** Jiang and Conrath. 18, 82

**KB** Knowledge-based. 16

**KG** knowledge graphs. 28

**LIGO** Laser Interferometric Gravitational Observatory. 36

## Acronyms

**NEL** Named Entity Linking. 19, 89

**NER** Named Entity Recognition. 6, 19, 54, 100, 124, 141

**RS** Recommender Systems. 1–6, 11, 12, 14–21, 23, 24, 27, 29, 36, 37, 41, 44–49, 53, 70, 71, 74, 75, 77–79, 97, 121, 124, 125, 128, 129, 133, 135–137, 139

# References


[1] Chembl. URL https://www.ebi.ac.uk/chembl/. 77

[2] Chemical entities of biological interest (chebi), . URL https://www.ebi.ac.uk/chebi/. 18, 75

[3] Chebi definition for caffeine, . URL https://www.ebi.ac.uk/chebi/searchId.do?chebiId=CHEBI:27732. 18, 75

[4] Chebi entity "chemical entity", . URL https://www.ebi.ac.uk/chebi/searchId.do?chebiId=CHEBI:24431. 74

[5] Cherm: Chemical compounds recommender matrix, . URL https://github.com/lasigeBioTM/CheRM. 83

[6] Dishin: Semantic similarity measures using disjunctive shared information. URL https://github.com/lasigeBioTM/DiShIn. 82

[7] Disease ontology (do). URL http://disease-ontology.org/. 18, 75

[8] Drugbank: Pharmaceutical knowledge base. URL https://go.drugbank.com/. 98

[9] Gene ontology (go). URL http://geneontology.org/. 18, 75

[10] Fast python collaborative filtering for implicit datasets. URL https://implicit.readthedocs.io/en/latest/index.html. 81

[11] Pandas python library. URL https://pandas.pydata.org/. 85

[12] Fabian Abel, Qi Gao, Geert-Jan Houben, and Ke Tao. Analyzing user modeling on twitter for personalized news recommendations. In *international conference on user modeling, adaptation, and personalization*, pages 1–12. Springer, 2011. 38

[13] Alberto Accomazzi, Michael J. Kurtz, Edwin A. Henneken, Carolyn S. Grant, Donna M. Thompson, Roman Chyla, Steven McDonald, Taylor J. Shaulis, Sergi Blanco-Cuaresma, Golnaz Shapurian, Timothy W. Hostetler, and Matthew R. Templeton. New ADS Functionality for the Curator. pages 1–6, 2017. URL `http://arxiv.org/abs/1710.08505`. 53

[14] G Adomavicius and a Tuzhilin. Toward the Next Generation of Recommender Systems: a Survey of the State of the Art and Possible Extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6):734–749, 2005. ISSN 10414347. doi: 10.1109/TKDE.2005.99. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1423975`. 23

[15] Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE transactions on knowledge and data engineering*, 17(6):734–749, 2005. 1, 24, 45

[16] Giuseppe Agapito, Mariadelina Simeoni, Barbara Calabrese, Ilaria Caré, Theodora Lamprinoudi, Pietro H Guzzi, Arturo Pujia, Giorgio Fuiano, and Mario Cannataro. Dietos: A dietary recommender system for chronic diseases monitoring and management. *Computer methods and programs in biomedicine*, 153:93–104, 2018. 33, 52

[17] Charu C Aggarwal. Ensemble-based and hybrid recommender systems. In *Recommender Systems*, pages 199–224. Springer, 2016. 17, 75, 83

[18] Charu C Aggarwal. Evaluating recommender systems. In *Recommender systems*, pages 225–254. Springer, Boston, MA, 2016. 23

[19] Charu C Aggarwal. Knowledge-based recommender systems. In *Recommender Systems*, pages 167–197. Springer, 2016. 17

[20] Charu C Aggarwal. Model-based collaborative filtering. In *Recommender systems*, pages 71–138. Springer, 2016. xxi, 12, 14

[21] Rishabh Ahuja, Arun Solanki, and Anand Nayyar. Movie recommender system using k-means clustering and k-nearest neighbor. In *2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pages 263–268. IEEE, 2019. 16

[22] Malak Al-Hassan, Haiyan Lu, and Jie Lu. A semantic enhanced hybrid recommendation approach: A case study of e-government tourism service recommendation system. *Decision Support Systems*, 72:97–109, 2015. 25, 28, 41, 79

[23] Mohammed Fadhel Aljunid and DH Manjaiah. Movie recommender system based on collaborative filtering using apache spark. In *Data Management, Analytics and Innovation*, pages 283–295. Springer, Boston, MA, 2019. 79

[24] Faris Alqadah, Chandan K Reddy, Junling Hu, and Hatim F Alqadah. Biclustering neighborhood-based collaborative filtering method for top-n recommender systems. *Knowledge and Information Systems*, 44(2):475–491, 2015. 48

[25] Fatemeh Alyari and Nima Jafari Navimipour. Recommender systems: a systematic review of the state of the art literature and suggestions for future research. *Kybernetes*, 47(5):985–1017, 2018. 45

[26] George M Anderson. Fluvoxamine, melatonin and covid-19. *Psychopharmacology*, 238(2):611–611, 2021. 19

[27] Marleen Balvert, Georgios Patoulidis, Andrew Patti, Timo M Deist, Christine Eyler, Bas E Dutilh, Alexander Schønhuth, and David Craft. A drug recommendation system (dr. s) for cancer cell lines. *arXiv preprint arXiv:1912.11548*, 2019. 34, 36

[28] Marcia Barros and Francisco M Couto. Knowledge representation and management: a linked data perspective. *Yearbook of medical informatics*, 25(01):178–183, 2016. 16, 75, 103

[29] Márcia Barros, André Moitinho, and Francisco M Couto. Using research literature to generate datasets of implicit feedback for recommending scientific items. *IEEE Access*, 7:176668–176680, 2019. 74, 83, 103, 107, 108, 125, 126, 129

[30] Márcia Barros, André Moitinho, and Francisco M Couto. Hybrid semantic recommender system for chemical compounds. In *European Conference on Information Retrieval*, pages 94–101. Springer, 2020. 80, 107, 125, 129, 130

[31] Márcia Barros, André Moitinho, and Francisco M Couto. Hybrid semantic recommender system for chemical compounds in large-scale datasets. *Journal of cheminformatics*, 13:1–18, 2021. 107, 112

[32] Pierpaolo Basile, Cataldo Musto, Marco de Gemmis, Pasquale Lops, Fedelucio Narducci, and Giovanni Semeraro. Aggregation strategies for linked open data-enabled recommender systems. *11th ESWC*, 2014. 39

[33] Zeynep Batmaz, Ali Yurekli, Alper Bilge, and Cihan Kaleli. A review on deep learning for recommender systems: challenges and remedies. *Artificial Intelligence Review*, 52 (1):1–37, 2019. 45

[34] Joeran Beel, Bela Gipp, Stefan Langer, and Corinna Breitinger. paper recommender systems: a literature survey. *International Journal on Digital Libraries*, 17(4):305–338, 2016. 1, 23, 24, 44, 45, 48

[35] James Bennett, Stan Lanning, et al. The netflix prize. In *Proceedings of KDD cup and workshop*, volume 2007, page 35. New York, NY, USA, 2007. 103

[36] Alex Beutel, Paul Covington, Sagar Jain, Can Xu, Jia Li, Vince Gatto, and Ed H Chi. Latent cross: Making use of context in recurrent recommender systems. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pages 46–54, 2018. 26, 28

[37] Zhongqin Bi, Siming Zhou, Xiaoxian Yang, Ping Zhou, and Jiale Wu. An approach for item recommendation using deep neural network combined with the bayesian personalized ranking. In *International Conference on Collaborative Computing: Networking, Applications and Worksharing*, pages 151–165. Springer, 2019. 79

[38] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez. Recommender systems survey. *Knowledge-Based Systems*, 46:109–132, 2013. ISSN 09507051. doi: 10.1016/j.knosys.2013.03.012. URL `http://dx.doi.org/10.1016/j.knosys.2013.03.012`. 45

[39] Jesús Bobadilla, Francisco Serradilla, and Jesus Bernal. A new collaborative filtering metric that improves the behavior of recommender systems. *Knowledge-Based Systems*, 23(6):520–528, 2010. 61

[40] Jesús Bobadilla, Fernando Ortega, Antonio Hernando, and Abraham Gutiérrez. Recommender systems survey. *Knowledge-based systems*, 46:109–132, 2013. 1, 23, 24

[41] Carlos Luis Sanchez Bocanegra, Jose Luis Sevillano Ramos, Carlos Rizo, Anton Civit, and Luis Fernandez-Luque. Healthrecsys: A semantic content-based recommender system to complement health videos. *BMC medical informatics and decision making*, 17(1):1–10, 2017. 31, 51

[42] Lutz Bornmann and Rüdiger Mutz. Growth rates of modern science: A bibliometric analysis based on the number of publications and cited references. *Journal of the Association for Information Science and Technology*, 66(11):2215–2222, 2015. 44

[43] Jonas Bostrøm, Niklas Falk, and Christian Tyrchan. Exploiting personalized information for reagent selection in drug design. *Drug discovery today*, 16(5-6):181–187, 2011. 30, 50, 77

[44] Robin Burke. Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, 12(4):331–370, 2002. 1, 23, 45

[45] Jonathan H Chen, Tanya Podchiyska, and Russ B Altman. Orderrex: clinical order decision support and outcome predictions by data-mining electronic medical records. *Journal of the American Medical Informatics Association*, 23(2):339–348, 2016. 31, 50

[46] Jonathan H Chen, Muthuraman Alagappan, Mary K Goldstein, Steven M Asch, and Russ B Altman. Decaying relevance of clinical data towards future decisions in data-driven inpatient clinical order sets. *International journal of medical informatics*, 102: 71–79, 2017. 32, 51

[47] Xu Chen, Hongteng Xu, Yongfeng Zhang, Jiaxi Tang, Yixin Cao, Zheng Qin, and Hongyuan Zha. Sequential recommendation with user memory networks. In *Proceedings of the eleventh ACM international conference on web search and data mining*, pages 108–116, 2018. 26

[48] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. Wide & deep learning for recommender systems. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*, pages 7–10. ACM, 2016. 25, 28

[49] Gene Ontology Consortium. The gene ontology resource: 20 years and still going strong. *Nucleic acids research*, 47(D1):D330–D338, 2018. 18, 75, 103

[50] Gianluca Corrado, Toma Tebaldi, Fabrizio Costa, Paolo Frasconi, and Andrea Passerini. Rnacommender: genome-wide recommendation of rna–protein interactions. *Bioinformatics*, 32(23):3627–3634, 2016. 31, 51

[51] F Couto and Andre Lamurias. Semantic similarity definition. *Encyclopedia of bioinformatics and computational biology*, 1, 2019. 82, 109, 112

[52] Francisco M Couto and Andre Lamurias. MER: a shell script and annotation server for minimal named entity recognition and linking. *Journal of cheminformatics*, 10(1): 58, 2018. 128

[53] Wim De Smet, Karel De Loof, Paul De Vos, Peter Dawyndt, and Bernard De Baets. Filtering and ranking techniques for automated selection of high-quality 16s rrna gene sequences. *Systematic and applied microbiology*, 36(8):549–559, 2013. 30, 50

[54] Kirill Degtyarenko, Paula De Matos, Marcus Ennis, Janna Hastings, Martin Zbinden, Alan McNaught, Rafael Alcántara, Michael Darsow, Mickaël Guedj, and Michael Ashburner. Chebi: a database and ontology for chemical entities of biological interest. *Nucleic acids research*, 36(suppl_1):D344–D350, 2007. 47

[55] Marc Peter Deisenroth, A Aldo Faisal, and Cheng Soon Ong. *Mathematics for machine learning*. Cambridge University Press, 2020. 23

[56] Tommaso Di Noia and Vito Claudio Ostuni. Recommender Systems and Linked Open Data. *Reasoning Web. Web Logic Rules*, pages 88–113, 2015. doi: 10.1007/978-3-319-21768-0_4. URL `http://link.springer.com/10.1007/978-3-319-21768-0{_}4`. 17

[57] Wilton S Dias, Héktor Monteiro, Aandré Moitinho, Jácques RD Lépine, Giovanni Carraro, Ernst Paunzen, Bruno Alessi, and Lázaro Villela. Updated parameters of 1743 open clusters based on gaia dr2. *Monthly Notices of the Royal Astronomical Society*, 504(1):356–371, 2021. 108

[58] WS Dias, BS Alessi, A Moitinho, and JRD Lépine. New catalogue of optically visible open clusters and candidates. *Astronomy & Astrophysics*, 389(3):871–873, 2002. 47, 55

[59] Marcos A Domingues, Camila V Sundermann, Flávio MM Barros, Marcelo G Manzato, Maria GC Pimentel, Solange O Rezende, and Stanley Oliveira. Applying multi-view based metadata in personalized ranking for recommender systems. In *Proceedings of the 30th Annual ACM Symposium on Applied Computing*, pages 1105–1107, 2015. 39

[60] Travis Ebesu, Bin Shen, and Yi Fang. Collaborative memory network for recommendation systems. In *The 41st international ACM SIGIR conference on research & development in information retrieval*, pages 515–524, 2018. 26

[61] Tome Eftimov, Barbara Koroušić Seljak, and Peter Korošec. A rule-based named-entity recognition method for knowledge extraction of evidence-based dietary recommendations. *PloS one*, 12(6):e0179488, 2017. 39

[62] Michael D Ekstrand. The lkpy package for recommender systems experiments: Next-generation tools and lessons learned from the lenskit project. *arXiv preprint arXiv:1809.03125*, 2018. 61

[63] Mehdi Elahi, Danial Khosh Kholgh, Mohammad Sina Kiarostami, Sorush Saghari, Shiva Parsa Rad, and Marko Tkalčič. Investigating the impact of recommender systems on user-based and item-based popularity bias. *Information Processing & Management*, 58(5):102655, 2021. 13

[64] Akram Emdadi and Changiz Eslahchi. Dsplmf: a method for cancer drug sensitivity prediction using a novel regularization approach in logistic matrix factorization. *Frontiers in genetics*, 11:75, 2020. 34, 36

[65] Ali Ezzat, Peilin Zhao, Min Wu, Xiao-Li Li, and Chee-Keong Kwoh. Drug-target interaction prediction with graph regularized matrix factorization. *IEEE/ACM transactions on computational biology and bioinformatics*, 14(3):646–656, 2016. 32, 36

[66] Jun Fan, Jing Yang, and Zhenran Jiang. Prediction of central nervous system side effects through drug permeability to blood–brain barrier and recommendation algorithm. *Journal of Computational Biology*, 25(4):435–443, 2018. 32, 36

[67] Wenqi Fan, Yao Ma, Qing Li, Yuan He, Eric Zhao, Jiliang Tang, and Dawei Yin. Graph neural networks for social recommendation. In *The World Wide Web Conference*, pages 417–426, 2019. 26, 29

[68] João D Ferreira and Francisco M Couto. Semantic similarity for automatic classification of chemical compounds. *PLoS Comput Biol*, 6(9):e1000937, 2010. 18, 77, 112

[69] Mouzhi Ge, Francesco Ricci, and David Massimo. Health-aware food recommender system. In *Proceedings of the 9th ACM Conference on Recommender Systems*, pages 333–334. ACM, 2015. 46

[70] David Goldberg, David Nichols, Brian M Oki, and Douglas Terry. Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, 35(12):61–70, 1992. 12

[71] Yuyun Gong and Qi Zhang. Hashtag recommendation using attention-based convolutional neural network. In *IJCAI*, pages 2782–2788, 2016. 25

[72] Archana Goyal, Vishal Gupta, and Manish Kumar. Recent named entity recognition and classification techniques: a systematic review. *Computer Science Review*, 29: 21–43, 2018. 19

[73] Tiago Grego and Francisco M Couto. Enhancement of chemical entity identification in text using semantic similarity validation. *PloS one*, 8(5):e62984, 2013. 18, 77

[74] Felix Gr
ßer, Stefanie Beckert, Denise K
ster, Jochen Schmitt, Susanne Abraham, Hagen Malberg, and Sebastian Zaunseder. Therapy decision support based on recommender system methods. *Journal of healthcare engineering*, 2017, 2017. 32, 51

[75] Guibing Guo, Jie Zhang, and Neil Yorke-Smith. Leveraging multiviews of trust and similarity to enhance clustering-based recommender systems. *Knowledge-Based Systems*, 74:14–27, 2015. 25, 48

[76] Fang Hao and Rachael Hageman Blair. A comparative study: classification vs. user-based collaborative filtering for clinical prediction. *BMC medical research methodology*, 16(1):1–14, 2016. 31, 51

[77] Ming Hao, Stephen H Bryant, and Yanli Wang. A new chemoinformatics approach with improved strategies for effective predictions of potential drugs. *Journal of cheminformatics*, 10(1):1–9, 2018. 34, 36, 77

[78] Tianshu Hao and Ziping Zheng. The implementation and optimization of matrix decomposition based collaborative filtering task on x86 platform. In *International Symposium on Benchmarking, Measuring and Optimization*, pages 110–115. Springer, 2019. 15, 79

[79] F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4):1–19, 2015. 74, 84, 103

[80] F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4):19, 2016. 58, 61

[81] Janna Hastings, Gareth Owen, Adriano Dekker, Marcus Ennis, Namrata Kale, Venkatesh Muthukrishnan, Steve Turner, Neil Swainston, Pedro Mendes, and Christoph Steinbeck. Chebi in 2016: Improved services and an expanding collection of metabolites. *Nucleic acids research*, 44(D1):D1214–D1219, 2015. 18, 75, 103

[82] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. In *Proceedings of the 26th International Conference on World Wide Web*, pages 173–182. International World Wide Web Conferences Steering Committee, 2017. 15, 26

[83] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 639–648, 2020. 27, 29

[84] John L Hennessy and David A Patterson. *Computer architecture: a quantitative approach*. Elsevier, Waltham, MA, 2011. 86

[85] Antonio Hernando, Jesús Bobadilla, and Fernando Ortega. A non negative matrix factorization for collaborative filtering recommender systems based on a bayesian probabilistic model. *Knowledge-Based Systems*, 97:188–202, 2016. 25, 48

[86] María Herrero-Zazo, Isabel Segura-Bedmar, Paloma Martínez, and Thierry Declerck. The DDI corpus: An annotated corpus with pharmacological substances and drug–drug interactions. *Journal of biomedical informatics*, 46(5):914–920, 2013. 124, 129

[87] Balázs Hidasi and Alexandros Karatzoglou. Recurrent neural networks with top-k gains for session-based recommendations. In *Proceedings of the 27th ACM international conference on information and knowledge management*, pages 843–852, 2018. 15, 19, 104

[88] Natalie R Hinkel, Cayman Unterborn, Stephen R Kane, Garrett Somers, and Richard Galvez. A recommendation algorithm to predict giant exoplanet host stars using stellar elemental abundances. *The Astrophysical Journal*, 880(1):49, 2019. 35, 36

[89] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006. 29

[90] Yifan Hu, Yehuda Koren, and Chris Volinsky. Collaborative filtering for implicit feedback datasets. In *2008 Eighth IEEE International Conference on Data Mining*, pages 263–272. Ieee, 2008. 15, 23, 79, 81, 85, 130

[91] Galatia Iatraki, Haridimos Kondylakis, Lefteris Koumakis, Maria Chatzimina, Eleni Kazantzaki, Kostas Marias, and Manolis Tsiknakis. Personal health information recommender: implementing a tool for the empowerment of cancer patients. *ecancermedicalscience*, 12, 2018. 33

[92] Andrea Iovine, Fedelucio Narducci, and Giovanni Semeraro. Conversational recommender systems and natural language:: A study through the converse framework. *Decision Support Systems*, 131:113250, 2020. 40

[93] Tsukasa Ishihara, Yuji Koga, Yoshiyuki Iwatsuki, and Fukushi Hirayama. Identification of potent orally active factor xa inhibitors based on conjugation strategy and application of predictable fragment recommender system. *Bioorganic & medicinal chemistry*, 23(2):277–289, 2015. 31, 36, 49, 50, 74, 77

[94] Seongwon Jang, Hoyeop Lee, Hyunsouk Cho, and Sehee Chung. Cities: Contextual inference of tail-item embeddings for sequential recommendation. In *2020 IEEE International Conference on Data Mining (ICDM)*, pages 202–211. IEEE, 2020. 106

[95] Dietmar Jannach, Markus Zanker, Mouzhi Ge, and Marian Gröning. Recommender systems in computer science and information systems–a landscape of research. In *International Conference on Electronic Commerce and Web Technologies*, pages 76–87. Springer, 2012. 46

[96] Jay J Jiang and David W Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. *arXiv preprint cmp-lg/9709008*, 1997. 18, 82, 109, 112

[97] Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 197–206. IEEE, 2018. 19, 26, 104

[98] Mozhgan Karimi, Dietmar Jannach, and Michael Jugovac. News recommender systems – Survey and roads ahead. *Information Processing and Management*, (February 2017):1–25, 2018. ISSN 03064573. doi: 10.1016/j.ipm.2018.04.008. URL `https://doi.org/10.1016/j.ipm.2018.04.008`. 23

[99] Jared L Katzman, Uri Shaham, Alexander Cloninger, Jonathan Bates, Tingting Jiang, and Yuval Kluger. Deepsurv: personalized treatment recommender system using a cox proportional hazards deep neural network. *BMC medical research methodology*, 18(1):1–12, 2018. 33, 52

[100] W. E. Kerzendorf. Knowledge discovery through text-based similarity searches for astronomy literature. 2017. URL `http://arxiv.org/abs/1705.05840`. 53

[101] Wolfgang E Kerzendorf. Knowledge discovery through text-based similarity searches for astronomy literature. *Journal of Astrophysics and Astronomy*, 40(3):1–7, 2019. 35, 36

[102] Farhan Khawar and Nevin L Zhang. Conformative filtering for implicit feedback data. In *European Conference on Information Retrieval*, pages 164–178. Springer, 2019. 74

[103] Donghyun Kim, Chanyoung Park, Jinoh Oh, Sungyoung Lee, and Hwanjo Yu. Convolutional matrix factorization for document context-aware recommendation. In *Proceedings of the 10th ACM conference on recommender systems*, pages 233–240, 2016. 25, 28

[104] Hyunjin Kim, Sang-Min Choi, and Sanghyun Park. Gseh: A novel approach to select prostate cancer-associated genes using gene expression heterogeneity. *IEEE/ACM transactions on computational biology and bioinformatics*, 15(1):129–146, 2016. 34, 36

[105] Yehuda Koren, Robert Bell, and Chris Volinsky. MATRIX FACTORIZATION TECHNIQUES FOR RECOMMENDER SYSTEMS. pages 30–37, 2009. 15

[106] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009. 45, 75

[107] Hermann Kroll, Jan Pirklbauer, Johannes Ruthmann, and Wolf-Tilo Balke. A semantically enriched dataset based on biomedical NER for the COVID19 open research dataset challenge, 2020. 126

[108] Michael J Kurtz, Guenther Eichhorn, Alberto Accomazzi, Carolyn Grant, Markus Demleitner, and Stephen S Murray. Worldwide use and impact of the nasa astrophysics data system digital library. *Journal of the American Society for Information Science and Technology*, 56(1):36–45, 2005. 47

[109] Andre Lamurias, Tiago Grego, and Francisco M Couto. Chemical compound and drug name recognition using crfs and semantic similarity based on chebi. In *BioCreative Challenge Evaluation Workshop*, volume 2, page 75. Citeseer, 2013. 18, 77, 112, 124

[110] Chao Lan, Sai Nivedita Chandrasekaran, and Jun Huan. On the unreported-profile-is-negative assumption for predictive cheminformatics. *IEEE/ACM transactions on computational biology and bioinformatics*, 17(4):1352–1363, 2019. 34, 36

[111] J Richard Landis and Gary G Koch. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174, 1977. 131

[112] James Lee-Thorp, Joshua Ainslie, Ilya Eckstein, and Santiago Ontanon. Fnet: Mixing tokens with fourier transforms. *arXiv preprint arXiv:2105.03824*, 2021. 110

[113] Jiao Li, Yueping Sun, Robin J Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Thomas C Wiegers, and Zhiyong Lu. Biocreative V CDR task corpus: a resource for chemical disease relation extraction. *Database*, 2016, 2016. 129

[114] Jing Li, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Tao Lian, and Jun Ma. Neural attentive session-based recommendation. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 1419–1428, 2017. 26

[115] Yi Liang, Shaokang Zeng, Yande Liang, and Kaizhong Chen. Accelerating parallel als for collaborative filtering on hadoop. In *International Symposium on Benchmarking, Measuring and Optimization*, pages 123–137. Springer, 2019. 15, 79

[116] I-En Liao, Wen-Chiao Hsu, Ming-Shen Cheng, and Li-Ping Chen. A library recommender system based on a personal ontology model and collaborative filtering technique for english collections. *The electronic library*, 28(3):386–400, 2010. 41, 79

[117] U Liji, Yahui Chai, and Jianrui Chen. Improved personalized recommendation based on user attributes clustering and score matrix filling. *Computer Standards & Interfaces*, 57:59–67, 2018. 103

[118] Hansaim Lim and Lei Xie. A new weighted imputed neighborhood-regularized tri-factorization one-class collaborative filtering algorithm: Application to target gene prediction of transcription factors. *IEEE/ACM transactions on computational biology and bioinformatics*, 18(1):126–137, 2020. 34, 36

[119] Carla Limongelli, Matteo Lombardi, Alessandro Marani, and Davide Taibi. Enrichment of the dataset of joint educational entities with the web of data. In *2017 IEEE 17th International Conference on Advanced Learning Technologies (ICALT)*, pages 528–529. IEEE, 2017. 39

[120] Dekang Lin et al. An information-theoretic definition of similarity. In *Icml*, volume 98, pages 296–304. Citeseer, 1998. 18, 82, 109, 112

[121] Hui Liu, Yan Zhao, Lin Zhang, and Xing Chen. Anti-cancer drug response prediction using neighbor-based collaborative filtering with global effect removal. *Molecular Therapy-Nucleic Acids*, 13:303–311, 2018. 33, 36

[122] Manuel Lobo, Andre Lamurias, and Francisco M Couto. Identifying human phenotype terms by combining machine learning and validation rules. *BioMed Research International*, 2017. 124

[123] Pasquale Lops, Marco De Gemmis, and Giovanni Semeraro. Content-based recommender systems: State of the art and trends. In *Recommender systems handbook*, pages 73–105. Springer, 2011. 16

[124] Jie Lu, Dianshuang Wu, Mingsong Mao, Wei Wang, and Guangquan Zhang. Recommender system application developments: a survey. *Decision Support Systems*, 74: 12–32, 2015. 1, 44, 45

[125] Linyuan Lü, Matus Medo, Chi Ho Yeung, Yi-Cheng Zhang, Zi-Ke Zhang, and Tao Zhou. Recommender Systems. *Physics Reports*, 519(1):1–49, 2012. ISSN 03701573. doi: 10.1016/j.physrep.2012.02.006. URL `http://arxiv.org/abs/1202.1112{%}0Ahttp://dx.doi.org/10.1016/j.physrep.2012.02.006`. 2

[126] Dmitriy Lyubimov and Andrew Palumbo. *Apache Mahout: Beyond MapReduce*. CreateSpace Independent Publishing Platform, 2016. 61

[127] Jianxin Ma, Chang Zhou, Hongxia Yang, Peng Cui, Xin Wang, and Wenwu Zhu. Disentangled self-supervision in sequential recommenders. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 483–491, 2020. 106

[128] Alessandra Alaniz Macedo, Juliana Tarossi Pollettini, José Augusto Baranauskas, and Julia Carmona Almeida Chaves. A health surveillance software framework to deliver information on preventive healthcare strategies. *Journal of biomedical informatics*, 62:159–170, 2016. 31, 51

[129] KL Malanchev, MV Pruzhinskaya, VS Korolev, PD Aleo, MV Kornilov, EEO Ishida, VV Krushinsky, F Mondon, S Sreejith, AA Volnova, et al. Anomaly detection in the zwicky transient facility dr3. *Monthly Notices of the Royal Astronomical Society*, 502 (4):5147–5175, 2021. 35, 37

[130] Marcelo G Manzato, Marcos A Domingues, Arthur C Fortes, Camila V Sundermann, Rafael M D'Addio, Merley S Conrado, Solange O Rezende, and Maria GC Pimentel. Mining unstructured content for recommender systems: an ensemble approach. *Information Retrieval Journal*, 19(4):378–415, 2016. 39

[131] Carmen Martinez-Cruz, Carlos Porcel, Juan Bernabé-Moreno, and Enrique Herrera-Viedma. A model to represent users trust in recommender systems using ontologies and fuzzy linguistic modeling. *Information Sciences*, 311:102–118, 2015. 25, 28, 29

[132] Mary L McHugh. Interrater reliability: the kappa statistic. *Biochemia medica*, 22(3): 276–282, 2012. 128

[133] Nikhil Mukund, Saurabh Thakur, Sheelu Abraham, AK Aniyan, Sanjit Mitra, Ninan Sajeeth Philip, Kaustubh Vaghmare, and DP Acharjya. An information retrieval and recommendation system for astronomical observatories. *The Astrophysical Journal Supplement Series*, 235(1):22, 2018. 53

[134] Nikhil Mukund, Saurabh Thakur, Sheelu Abraham, AK Aniyan, Sanjit Mitra, Ninan Sajeeth Philip, Kaustubh Vaghmare, and DP Acharjya. An information retrieval and recommendation system for astronomical observatories. *The Astrophysical Journal Supplement Series*, 235(1):22, 2018. 35, 36

[135] Anam Mustaqeem, Syed Muhammad Anwar, Abdul Rashid Khan, and Muhammad Majid. A statistical analysis based recommender model for heart disease patients. *International journal of medical informatics*, 108:134–145, 2017. 31, 51

[136] Anam Mustaqeem, Syed Muhammad Anwar, and Muhammad Majid. A modular cluster based collaborative recommender system for cardiac patients. *Artificial intelligence in medicine*, 102:101761, 2020. 34

[137] Cataldo Musto, Giovanni Semeraro, Pasquale Lops, and Marco de Gemmis. Combining distributional semantics and entity linking for context-aware content-based recommendation. In *International Conference on User Modeling, Adaptation, and Personalization*, pages 381–392. Springer, 2014. 38

[138] David M Ng, Marcos H Woehrmann, and Joshua M Stuart. Recommending pathway genes using a compendium of clustering solutions. In *Biocomputing 2007*, pages 379–390. World Scientific, 2007. 30, 36, 50

[139] Mehrbakhsh Nilashi, Othman Ibrahim, and Karamollah Bagherifard. A recommender system based on collaborative filtering using ontology and dimensionality reduction techniques. *Expert Systems with Applications*, 92:507–520, 2018. 41, 79

[140] Shumpei Okura, Yukihiro Tagami, Shingo Ono, and Akira Tajima. Embedding-based news recommendation for millions of users. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1933–1942, 2017. 26, 28

[141] Alberto Olivares Alarcos. Semantic distances between medical entities. Master's thesis, Universitat Politècnica de Catalunya, 2018. 98

[142] World Health Organization et al. Coronavirus disease 2019 (COVID-19) situation report-1, 2020. 124

[143] Fernando Ortega, Jesús Bobadilla, Abraham Gutiérrez, Remigio Hurtado, and Xin Li. Artificial intelligence scientific documentation dataset for recommender systems. *IEEE Access*, 6:48543–48555, 2018. 46, 47, 53, 61, 62, 74, 125, 126

[144] Fernando Ortega, Bo Zhu, Jesús Bobadilla, and Antonio Hernando. Cf4j: Collaborative filtering for java. *Knowledge-Based Systems*, 152:94–99, 2018. 48, 61

[145] Vito Claudio Ostuni, Tommaso Di Noia, Eugenio Di Sciascio, and Roberto Mirizzi. Top-n recommendations from implicit feedback leveraging linked open data. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 85–92. ACM, 2013. 41, 79

[146] Art B Owen, Josh Stuart, Kathy Mach, Anne M Villeneuve, and Stuart Kim. A gene recommender algorithm to identify coexpressed genes in c. elegans. *Genome research*, 13(8):1828–1837, 2003. 30, 36, 50

[147] Makbule Guclin Ozsoy, Tansel Øzyer, Faruk Polat, and Reda Alhajj. Realizing drug repositioning by adapting a recommendation system to handle the process. *BMC bioinformatics*, 19(1):1–14, 2018. 33, 36

[148] Umberto Panniello, Alexander Tuzhilin, and Michele Gorgoglione. Comparing context-aware recommender systems in terms of accuracy and diversity. *User Modeling and User-Adapted Interaction*, 24(1-2):35–65, 2014. ISSN 09241868. doi: 10.1007/s11257-012-9135-y. 62

[149] Michael J Pazzani and Daniel Billsus. Content-based recommendation systems. In *The adaptive web*, pages 325–341. Springer, 2007. 16, 109

[150] Andre Luiz Vizine Pereira and Eduardo Raul Hruschka. Simultaneous co-clustering and learning to address the cold start problem in recommender systems. *Knowledge-Based Systems*, 82:11–19, 2015. 25, 27, 48

[151] Saverio Perugini, Marcos André Gonçalves, and Edward A Fox. Recommender systems research: A connection-centric survey. *Journal of Intelligent Information Systems*, 23(2):107–143, 2004. 1, 45

[152] Ladislav Peska, Krisztian Buza, and Júlia Koller. Drug-target interaction prediction: a bayesian ranking approach. *Computer methods and programs in biomedicine*, 152: 15–21, 2017. 32, 36

[153] Ivens Portugal, Paulo Alencar, and Donald Cowan. The use of machine learning algorithms in recommender systems: A systematic review. *Expert Systems with Applications*, 97:205–227, 2018. 45

[154] T Prusti, JHJ de Bruijne, A Vallenari, C Babusiaux, CAL Bailer-Jones, U Bastian, M Biermann, DW Evans, L Eyer, F Jansen, et al. The gaia mission. *Astronomy & Astrophysics*, 595:A1, 2016. 112

[155] Evgenii Pustozerov, Polina Popova, Aleksandra Tkachuk, Yana Bolotko, Zafar Yuldashev, and Elena Grineva. Development and evaluation of a mobile personalized blood glucose prediction system for patients with gestational diabetes mellitus. *JMIR mHealth and uHealth*, 6(1):e6, 2018. 33, 52

[156] Quan Qi and Jing Dong. Named entity recognition in titles of chinese videos from the web. In *2011 IEEE International Conference on Computer Science and Automation Engineering*, volume 4, pages 220–224. IEEE, 2011. 38

[157] Massimo Quadrana, Paolo Cremonesi, and Dietmar Jannach. Sequence-aware recommender systems. *ACM Computing Surveys (CSUR)*, 51(4):1–36, 2018. 19, 102

[158] Behnam Rahdari, Peter Brusilovsky, Khushboo Thaker, and Hung Kim Chau. Covex: An exploratory search system for COVID-19 scientific literature. 126

[159] Zhiyun Ren, Bo Peng, Titus K Schleyer, and Xia Ning. Hybrid collaborative filtering methods for recommending search terms to clinicians. *Journal of Biomedical Informatics*, 113:103635, 2021. 35

[160] Steffen Rendle, Leandro Balby Marinho, Alexandros Nanopoulos, and Lars Schmidt-Thieme. Learning optimal ranking with tensor factorization for tag recommendation. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 727–736. ACM, 2009. 74

[161] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence*, pages 452–461. AUAI Press, 2009. 15, 23, 74, 79, 81, 85

[162] Allen H Renear, Simone Sacchi, and Karen M Wickett. Definitions of dataset in the scientific and technical literature. *Proceedings of the Association for Information Science and Technology*, 47(1):1–4, 2010. 3

[163] Philip Resnik. Using information content to evaluate semantic similarity in a taxonomy. *arXiv preprint cmp-lg/9511007*, 1995. 18, 82, 109, 112

[164] Francesco Ricci, Lior Rokach, and Bracha Shapira. Recommender systems: introduction and challenges. In *Recommender systems handbook*, pages 1–34. Springer, 2015. 44

[165] Francesco Ricci, Lior Rokach, and Bracha Shapira. Recommender systems: introduction and challenges. In *Recommender systems handbook*, pages 1–34. Springer, 2015. 1, 20, 74, 102, 125

[166] Pedro Ruas, Andre Lamurias, and Francisco M Couto. Linking chemical and disease entities to ontologies by integrating pagerank with extracted relations from literature. *Journal of Cheminformatics*, 12(1):1–11, 2020. 89

[167] John Savage, Akihiro Kishimoto, Beat Buesser, Ernesto Diaz-Aviles, and Carlos Alzate. Chemical reactant recommendation using a network of organic chemistry. In *Proceedings of the Eleventh ACM Conference on Recommender Systems*, pages 210–214, 2017. 31, 36

[168] Hanna Schäfer, Santiago Hors-Fraile, Raghav Pavan Karumur, André Calero Valdez, Alan Said, Helma Torkamaan, Tom Ulmer, and Christoph Trattner. Towards health (aware) recommender systems. In *Proceedings of the 2017 international conference on digital health*, pages 157–161. ACM, 2017. 46

[169] Lynn M Schriml, Elvira Mitraka, James Munro, Becky Tauber, Mike Schor, Lance Nickle, Victor Felix, Linda Jeng, Cynthia Bearer, Richard Lichenstein, et al. Human disease ontology 2018 update: classification, content and workflow expansion. *Nucleic acids research*, 47(D1):D955–D962, 2018. 18, 75, 103

[170] Gunnar Schröder, Maik Thiele, and Wolfgang Lehner. Setting goals and choosing metrics for recommender system evaluations. In *UCERSTI2 Workshop at the 5th ACM conference on recommender systems, Chicago, USA*, volume 23, page 53, 2011. 21, 23

[171] Atsuto Seko, Hiroyuki Hayashi, and Isao Tanaka. Compositional descriptor-based recommender system for the materials discovery. *The Journal of chemical physics*, 148(24):241719, 2018. 33, 36, 49, 52, 74, 77

[172] Qusai Shambour and Jie Lu. A trust-semantic fusion-based recommendation approach for e-business applications. *Decision Support Systems*, 54(1):768–780, 2012. 23, 41, 79, 85

[173] Guy Shani and Asela Gunawardana. Evaluating Recommendation Systems. *Recommender Systems Handbook*, pages 257–297, 2011. ISSN 08909369. doi: 10.1007/978-0-387-85820-3_8. URL `http://link.springer.com/10.1007/978-0-387-85820-3{_}8`. 45

[174] Guy Shani and Asela Gunawardana. Evaluating recommendation systems. In *Recommender systems handbook*, pages 257–297. Springer, Boston, MA, 2011. 21, 84, 130

[175] Guy Shani, David Heckerman, Ronen I Brafman, and Craig Boutilier. An mdp-based recommender system. *Journal of Machine Learning Research*, 6(9), 2005. 19, 104

[176] Babak Maleki Shoja and Nasseh Tabrizi. Customer reviews analysis with deep neural networks for e-commerce recommender systems. *IEEE Access*, 7:119121–119130, 2019. 77

[177] Ahu Sieg, Bamshad Mobasher, and Robin Burke. Improving the effectiveness of collaborative recommendation with ontology-based user profiles. In *proceedings of the 1st International Workshop on Information Heterogeneity and Fusion in Recommender Systems*, pages 39–46. ACM, 2010. 41, 79

[178] Sanjeevan Sivapalan, Alireza Sadeghian, Hossein Rahnama, and Asad M Madni. Recommender systems in e-commerce. In *World Automation Congress (WAC), 2014*, pages 179–184. IEEE, 2014. 1, 44

[179] Yang Song, Ali Mamdouh Elkahky, and Xiaodong He. Multi-rate deep learning for temporal recommendation. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 909–912, 2016. 25

[180] Ekaterina A Sosnina, Sergey Sosnin, Anastasia A Nikitina, Ivan Nazarov, Dmitry I Osolodkin, and Maxim V Fedorov. Recommender systems in antiviral drug discovery. *ACS omega*, 2020. 77

[181] Diana Sousa and Francisco M Couto. Biont: deep learning using multiple biomedical ontologies for relation extraction. In *European Conference on Information Retrieval*, pages 367–374. Springer, 2020. 133

[182] Diana Sousa, André Lamúrias, and Francisco M Couto. A silver standard corpus of human phenotype-gene relations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1487–1492, 2019. 124

[183] Raghuram Srinivas, Pavel V Klimovich, and Eric C Larson. Implicit-descriptor ligand-based virtual screening by means of collaborative filtering. *Journal of cheminformatics*, 10(1):1–20, 2018. 34, 36

[184] Xiaoyuan Su and Taghi M Khoshgoftaar. A survey of collaborative filtering techniques. *Advances in artificial intelligence*, 2009:4, 2009. 1, 13, 23, 24, 45, 74

[185] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*, pages 1441–1450, 2019. 15, 19, 26, 104, 105, 110, 111

[186] Chayaporn Suphavilai, Denis Bertrand, and Niranjan Nagarajan. Predicting cancer drug response using a recommender system. *Bioinformatics*, 34(22):3907–3914, 2018. 33, 36, 46, 52

[187] Ewin Tang. A quantum-inspired classical algorithm for recommendation systems. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 217–228, 2019. 27

[188] Jiaxi Tang and Ke Wang. Personalized top-n sequential recommendation via convolutional sequence embedding. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pages 565–573, 2018. 19, 104

[189] Raphael Tang, Rodrigo Nogueira, Edwin Zhang, Nikhil Gupta, Phuong Cam, Kyunghyun Cho, and Jimmy Lin. Rapidly bootstrapping a question answering dataset for COVID-19. *arXiv preprint arXiv:2004.11339*, 2020. 126

[190] John K Tarus, Zhendong Niu, and Ghulam Mustafa. Knowledge-based recommendation: a review of ontology-based recommender systems for e-learning. *Artificial Intelligence Review*, 50(1):21–48, 2018. 16, 75, 103

[191] Hossen Teimoorinia, Sara Shishehchi, Ahnaf Tazwar, Ping Lin, Finn Archinuk, Stephen DJ Gwyn, and JJ Kavelaars. An astronomical image content-based recommendation system using combined deep learning models in a fully unsupervised mode. *The Astronomical Journal*, 161(5):227, 2021. 35, 37

[192] Nguyen Tho Thong et al. Intuitionistic fuzzy recommender systems: an effective tool for medical diagnosis. *Knowledge-Based Systems*, 74:133–150, 2015. 25, 29

[193] Yonghong Tian, Bing Zheng, Yanfang Wang, Yue Zhang, and Qi Wu. College library personalized recommendation system based on hybrid recommendation algorithm. *Procedia CIRP*, 83:490–494, 2019. 77

[194] Nava Tintarev and Judith Masthoff. A survey of explanations in recommender systems. In *Data Engineering Workshop, 2007 IEEE 23rd International Conference on*, pages 801–810. IEEE, 2007. 1, 45

[195] Atefeh Torkaman, Nasrollah Moghaddam Charkari, and Mahnaz Aghaeipour. An approach for leukemia classification based on cooperative game theory. *Analytical Cellular Pathology*, 34(5):235–246, 2011. 30, 50

[196] Denis Torre, Patrycja Krawczuk, Kathleen M Jagodnik, Alexander Lachmann, Zichen Wang, Lily Wang, Maxim V Kuleshov, and Avi Ma'ayan. Datasets2tools, repository and search engine for bioinformatics datasets, tools and canned analyses. *Scientific data*, 5(1):1–10, 2018. 102

[197] Ferran Torrent-Fontbona and Beatriz López. Personalized adaptive cbr bolus recommender system for type 1 diabetes. *IEEE journal of biomedical and health informatics*, 23(1):387–394, 2018. 34, 52

[198] Mike Uschold and Michael Gruninger. Ontologies: Principles, methods and applications. *The knowledge engineering review*, 11(2):93–136, 1996. 16, 75

[199] André Calero Valdez, Martina Ziefle, Katrien Verbert, Alexander Felfernig, and Andreas Holzinger. Recommender systems for health informatics: state-of-the-art and future perspectives. In *Machine Learning for Health Informatics*, pages 391–414. Springer, 2016. 46

[200] Andreu Vall, Hamid Eghbal-Zadeh, Matthias Dorfer, Markus Schedl, and Gerhard Widmer. Music playlist continuation by learning from hand-curated examples and song features: Alleviating the cold-start problem for rare and out-of-set songs. In *Proceedings of the 2nd Workshop on Deep Learning for Recommender Systems*, pages 46–54, 2017. 81

[201] Andreu Vall, Matthias Dorfer, Hamid Eghbal-Zadeh, Markus Schedl, Keki Burjorjee, and Gerhard Widmer. Feature-combination hybrid recommender systems for automated music playlist continuation. *User Modeling and User-Adapted Interaction*, 29 (2):527–572, 2019. 81

[202] Bogdan Walek and Vladimir Fojtik. A hybrid recommender system for recommending relevant movies using an expert system. *Expert Systems with Applications*, page 113452, 2020. 77

[203] Annie Wang, Hansaim Lim, Shu-Yuan Cheng, and Lei Xie. Antenna, a multi-rank, multi-layered recommender system for inferring reliable drug-gene-disease associations: repurposing diazoxide as a targeted anti-cancer therapy. *IEEE/ACM transactions on computational biology and bioinformatics*, 15(6):1960–1967, 2018. 33, 36, 46, 51

[204] Hang Wang, Jianing Xi, Minghui Wang, and Ao Li. Dual-layer strengthened collaborative topic regression modeling for predicting drug sensitivity. *IEEE/ACM transactions on computational biology and bioinformatics*, 17(2):587–598, 2018. 34, 36

[205] Hao Wang, Naiyan Wang, and Dit-Yan Yeung. Collaborative deep learning for recommender systems. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1235–1244. ACM, 2015. 62

[206] Hao Wang, Naiyan Wang, and Dit-Yan Yeung. Collaborative deep learning for recommender systems. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1235–1244. ACM, 2015. 25, 27

[207] Hongwei Wang, Fuzheng Zhang, Jialin Wang, Miao Zhao, Wenjie Li, Xing Xie, and Minyi Guo. Ripplenet: Propagating user preferences on the knowledge graph for recommender systems. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 417–426, 2018. 26, 28, 29

[208] Hongwei Wang, Fuzheng Zhang, Xing Xie, and Minyi Guo. Dkn: Deep knowledge-aware network for news recommendation. In *Proceedings of the 2018 world wide web conference*, pages 1835–1844, 2018. 26, 29

[209] Hongwei Wang, Fuzheng Zhang, Mengdi Zhang, Jure Leskovec, Miao Zhao, Wenjie Li, and Zhongyuan Wang. Knowledge-aware graph neural networks with label smoothness regularization for recommender systems. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 968–977, 2019. 26, 29

[210] Jianling Wang, Kaize Ding, Liangjie Hong, Huan Liu, and James Caverlee. Next-item recommendation with sequential hypergraphs. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1101–1110, 2020. 106

[211] Lucy Lu Wang, Kyle Lo, Yoganand Chandrasekhar, Russell Reas, Jiangjiang Yang, Darrin Eide, Kathryn Funk, Rodney Kinney, Ziyang Liu, William Merrill, et al. Cord-19: The COVID-19 open research dataset. *ArXiv*, 2020. 124, 128

[212] Xiangeng Wang, Xiaolei Zhu, Mingzhi Ye, Yanjing Wang, Cheng-Dong Li, Yi Xiong, and Dongqing Wei. Sts-nlsp: a network-based label space partition method for predicting the specificity of membrane transporter substrates using a hybrid feature of structural and semantic similarity. *Frontiers in Bioengineering and Biotechnology*, 7: 306, 2019. 18, 77, 112

[213] Xuan Wang, Weili Liu, Aabhas Chauhan, Yingjun Guan, and Jiawei Han. Automatic textual evidence mining in COVID-19 literature. *arXiv preprint arXiv:2004.12563*, 2020. 126

[214] Xuan Wang, Xiangchen Song, Yingjun Guan, Bangzheng Li, and Jiawei Han. Comprehensive named entity recognition on cord-19 with distant or weak supervision. *arXiv preprint arXiv:2003.12218*, 2020. 126

[215] Jian Wei, Jianhua He, Kai Chen, Yi Zhou, and Zuoyin Tang. Collaborative filtering and deep learning based recommendation system for cold start items. *Expert Systems with Applications*, 69:29–39, 2017. 25, 28

[216] Marc Wenger, François Ochsenbein, Daniel Egret, Pascal Dubois, François Bonnarel, Suzanne Borde, Françoise Genova, Gérard Jasniewicz, Suzanne Laloë, Soizick Lesteven, et al. The simbad astronomical database-the cds reference database for astronomical objects. *Astronomy and Astrophysics Supplement Series*, 143(1):9–22, 2000. 47

[217] Martin Wiesner and Daniel Pfeifer. Health recommender systems: concepts, requirements, technical basics and challenges. *International journal of environmental research and public health*, 11(3):2580–2607, 2014. 30, 50

[218] Hans Christian Wittich, Marco Seeland, Jana W
ldchen, Michael Rzanny, and Patrick M
der. Recommending plant taxa for supporting on-site species identification. *BMC bioinformatics*, 19(1):1–17, 2018. 33, 46

[219] Felix Wortmann and Kristina Flüchter. Internet of things. *Business & Information Systems Engineering*, 57(3):221–224, 2015. 24

[220] Chao-Yuan Wu, Amr Ahmed, Alex Beutel, Alexander J Smola, and How Jing. Recurrent recommender networks. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, pages 495–503. ACM, 2017. 25

[221] Le Wu, Peijie Sun, Yanjie Fu, Richang Hong, Xiting Wang, and Meng Wang. A neural influence diffusion model for social recommendation. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*, pages 235–244, 2019. 26

[222] Shu Wu, Yuyuan Tang, Yanqiao Zhu, Liang Wang, Xing Xie, and Tieniu Tan. Session-based recommendation with graph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 346–353, 2019. 27, 29

[223] Yao Wu, Christopher DuBois, Alice X Zheng, and Martin Ester. Collaborative denoising auto-encoders for top-n recommender systems. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*, pages 153–162. ACM, 2016. 15, 25

[224] Hong-Jian Xue, Xinyu Dai, Jianbing Zhang, Shujian Huang, and Jiajun Chen. Deep matrix factorization models for recommender systems. In *IJCAI*, volume 17, pages 3203–3209. Melbourne, Australia, 2017. 15, 26

[225] Longqi Yang, Cheng-Kang Hsieh, Hongjian Yang, John P Pollak, Nicola Dell, Serge Belongie, Curtis Cole, and Deborah Estrin. Yum-me: a personalized nutrient-based meal recommender system. *ACM Transactions on Information Systems (TOIS)*, 36(1): 1–31, 2017. 33, 51

[226] Nobuaki Yasuo, Yusuke Nakashima, and Masakazu Sekijima. Code-dti: Collaborative deep learning-based drug-target interaction prediction. In *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 792–797. IEEE, 2018. 34, 36

[227] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L Hamilton, and Jure Leskovec. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 974–983, 2018. 26, 28, 29

[228] Sobia Zahra, Mustansar Ali Ghazanfar, Asra Khalid, Muhammad Awais Azam, Usman Naeem, and Adam Prugel-Bennett. Novel centroid selection approaches for kmeans-clustering based recommender systems. *Information sciences*, 320:156–189, 2015. 25, 48

[229] Xiangxiang Zeng, Yinglai Lin, Yuying He, Linyuan L
, Xiaoping Min, and Alfonso Rodríguez-Patón. Deep collaborative filtering for prediction of disease genes. *IEEE/ACM transactions on computational biology and bioinformatics*, 17(5):1639–1647, 2019. 34, 36

[230] Edwin Zhang, Nikhil Gupta, Rodrigo Nogueira, Kyunghyun Cho, and Jimmy Lin. Rapidly deploying a neural search engine for the COVID-19 open research dataset: Preliminary thoughts and lessons learned. *arXiv preprint arXiv:2004.05125*, 2020. 126

[231] Fuzheng Zhang, Nicholas Jing Yuan, Defu Lian, Xing Xie, and Wei-Ying Ma. Collaborative knowledge base embedding for recommender systems. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 353–362. ACM, 2016. 25, 28, 48

[232] Heng-Ru Zhang, Fan Min, and Bing Shi. Regression-based three-way recommendation. *Information Sciences*, 378:444–461, 2017. 25

[233] Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay. Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys (CSUR)*, 52(1): 1–38, 2019. 15

[234] Wen Zhang, Hua Zou, Longqiang Luo, Qianchao Liu, Weijian Wu, and Wenyi Xiao. Predicting potential side effects of drugs by recommender methods and ensemble learning. *Neurocomputing*, 173:979–987, 2016. 46

[235] Xian-Da Zhang. Machine learning. In *A Matrix Algebra Approach to Artificial Intelligence*, pages 223–440. Springer, 2020. 14

[236] Xiaoyan Zhang, Haihua Luo, Bowei Chen, and Guibing Guo. Multi-view visual bayesian personalized ranking for restaurant recommendation. *APPLIED INTELLIGENCE*, 2020. 80

[237] Yin Zhang, Daqiang Zhang, Mohammad Mehedi Hassan, Atif Alamri, and Limei Peng. Cadre: Cloud-assisted drug recommendation service for online pharmacies. *Mobile Networks and Applications*, 20(3):348–355, 2015. 31

[238] Yin Zhang, Min Chen, Dijiang Huang, Di Wu, and Yong Li. idoctor: Personalized and professionalized medical recommendations based on hybrid matrix factorization. *Future Generation Computer Systems*, 66:30–35, 2017. 26

[239] Feng Zhao, Yu Shen, Xiangyu Gui, and Hai Jin. Sdbpr: Social distance-aware bayesian personalized ranking for recommendation. *Future Generation Computer Systems*, 95:372–381, 2019. 80

[240] Lei Zheng, Vahid Noroozi, and Philip S Yu. Joint deep modeling of users and items using reviews for recommendation. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, pages 425–434. ACM, 2017. 26

[241] Kun Zhou, Hui Wang, Wayne Xin Zhao, Yutao Zhu, Sirui Wang, Fuzheng Zhang, Zhongyuan Wang, and Ji-Rong Wen. S3-rec: Self-supervised learning for sequential recommendation with mutual information maximization. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 1893–1902, 2020. 106

[242] Martina Ziefle. Machine Learning for Health Informatics. 9605(November):0–24, 2016. doi: 10.1007/978-3-319-50478-0. URL `http://link.springer.com/10.1007/978-3-319-50478-0`. 3, 24

**