ROYAUME DU MAROC
Université Mohammed V - RABAT -
جامعة محمد الخامس - الرباط -
Faculté des sciences
كلية العلوم
CENTRE D'ETUDES DOCTORALES - SCIENCES ET TECHNOLOGIES

N° d'ordre :3565

THESE

En vue de l'obtention du : ***DOCTORAT***

Structure de Recherche : *Laboratoire de Recherche en Informatique et Télécommunications*

Discipline : *Sciences de l'Ingénieur*

Spécialité : *Informatique*

Présentée et soutenue le 18/12/2021 par :

Mehdi SRIFI

A Deep Learning Content-Based Recommender System: Application in the Context of Arabic Language

JURY

Moulay Driss RAHMANI	*PES, Faculté des Sciences, Université Mohammed V de Rabat*	*Président*
Youness TABII	*PH, École Nationale Supérieure d'Informatique et d'Analyse des Systèmes, Université Mohammed V de Rabat*	*Rapporteur/Examinateur*
Noureddine FALIH	*PH, Faculté Polydisciplinaire, Université Sultan Moulay Slimane de Béni Mellal*	*Rapporteur/Examinateur*
Lahoucine BALLIHI	*PH, Faculté des Sciences, Université Mohammed V de Rabat*	*Rapporteur/Examinateur*
Younès El BOUZEKRI EL IDRISSI	*PH, École Nationale des Sciences Appliquées, Université Ibn Tofail de Kénitra*	*Examinateur*
Ayoub AIT LAHCEN	*PH, École Nationale des Sciences Appliquées, Université Ibn Tofail de Kénitra*	*Co-encadrant*
Salma MOULINE	*PES, Faculté des Sciences, Université Mohammed V de Rabat*	*Directeur*

Année Universitaire : 2021/2022

✉ Faculté des Sciences, avenue Ibn Battouta, BP. 1014 RP, Rabat –Maroc
☎ 00212 (05) 37 77 18 76 🖷 00212(05) 37 77 42 61; http://www. fsr.um5.ac.ma

To My Beloved Parents
To My Dear Sisters

Acknowledgements

Allow me, first of all, to thank **Prof. Moulay Driss RAHMANI**, who did me the honor of chairing and judging this work. I also wish to express my respect and gratitude to all jury members: **Prof. Youness TABII**, **Prof. Noureddine FALIH**, **Prof. Lahoucine BALLIHI**, **Prof. Younès EL BOUZEKRI EL IDRISSI**, for their availability to read, review, and participate in the evaluation process of this Thesis. Their participation is a great pleasure and an honor for me.

In the past years as a Ph.D. student, I must admit that I was honored to be surrounded by eminent researchers who could provide me with advice and encouragement. Not only did they initiate me into research, but they inspired me with the will and the passion for integrating myself in the path of knowledge. I want to express my sincere gratitude to my Ph.D. Supervisor **Prof. Salma MOULINE** (FSR, LRIT laboratory) who kindly supervised this work with patience, kindness, and understanding. This work would never have been accomplished without his valuable assistance and support.

My warmest thanks also go to my dear **Prof. Ayoub AIT LAHCEN** (ENSAK, LGS laboratory), co-supervisor of my Thesis, who agreed to support this work with great rigor and kindness. I would also like to underline his commitment to realizing my Master's Thesis. He offered me the opportunity to undertake this research on a topical subject while giving me the confidence to assume and ensure this task. He guided me in my work by giving me a lot of support, advice, and research ideas through this long journey. Thanks for his motivation, professionalism, and empathy.

I would like to mention the late **Prof. Driss ABOUTAJEDDINE** (FSR, LRIT laboratory), the eminent professor of the Faculty of Sciences of Rabat who suddenly left us and was the Thesis's origin advisor. Allow me to dedicate this work to him as a testimony of his scientific and human qualities. May God have his soul.

Special thanks must go to **Prof. Younès EL BOUZEKRI EL IDRISSI** (ENSAK, LGS laboratory) endowed with the quality of sharing and exchange, who always provided me advice and encouraged me to persevere each time the opportunity arose.

Thank you very much, dear **Prof. Ahmed OUSSOUS** (FSTM, LIM laboratory), friend, and member of the team, for your moral and technical support and for having been by my side when I needed to exchange and discuss my thoughts.

I'm very appreciative to be a member of the LRIT group, where I have spent excellent times with my colleagues and friends. Thanks to all **LRIT members** with whom I shared memorable memories.

I am immensely grateful to **My Parents** for their endless sacrifices, understanding, and love that allowed me to become the person I am. I know very well that there are no proper words to convey my deep gratitude and love for you. In all this, thank you so much for tolerating my mood swings throughout this long period. Without your whole-hearted support, I would not be able to complete this work. I hope that this achievement will complete the dream you had for me all those many years ago.

I must express my profound gratitude to my two sisters, **Hajar** and **Dania**, who supported me and gave me the encouragement I needed throughout this process. I always knew that you believed in me and wanted the best for me. I'm grateful to both of you. You are amazing!

My gratitude and thankfulness for my two dear cousins **Nabil** and **Badr**, who have always been there for me in difficult moments, and who breathed me the strength and tenacity to lead and complete this modest work.

Thank you so much, everyone. This is just the beginning of a great adventure!

Résumé

À l'ère du big data, les Systèmes de Recommandation (Recommender Systems, RSs) sont devenus des outils de plus en plus utilisés. Ils constituent un type important d'algorithmes d'apprentissage automatique (Machine Learning, ML) qui contribuent principalement à préserver la fidélité des internautes, en mettant à leur disposition un contenu personnalisé sur les différentes plateformes électroniques telles que Amazon, Netflix, YouTube et Facebook. Les RSs sont bénéfiques à la fois pour les utilisateurs et les entreprises. Ils aident les utilisateurs à prendre des décisions, et les entreprises à faire plus de bénéfices. En fait, une grande partie des revenus de nombreuses entreprises est générée uniquement par les recommandations.

Plusieurs RSs ont été proposés dans la littérature, dont la plupart se sont principalement concentrés sur le contenu anglais. La recherche et les ressources concernant les RSs en d'autres langues notamment l'arabe sont restreintes. Ces derniers temps, le contenu arabe sur le Web a considérablement augmenté, et ce en raison du nombre croissant d'internautes arabes; l'arabe classée quatrième parmi les dix premières langues utilisées sur internet. Ce qui sollicite la nécessité de conduire des études portant sur le contenu arabe, plus particulièrement dans le domaine des RSs.

Cette thèse aborde les récents travaux réalisés dans le domaine des RSs, tout en mettant l'accent sur le manque de recherches concernant les RSs arabes. Par ailleurs, elle présente d'autres nouvelles contributions visant à améliorer ce domaine peu exploré jusqu'à présent.

La première contribution porte sur l'exploration et l'investigation des RSs récents initialement consacrés au contenu anglais, lorsqu'ils sont appliqués au contenu arabe. Elle envisage d'expérimenter des RSs de pointe à partir de trois aspects à savoir, l'applicabilité, l'impact du prétraitement et la performance lors du changement de la langue du contenu. En réalisant cet ensemble d'expérimentations, l'apport de cette première contribution réside aussi dans la construction et la mise à disposition de quatre datasets arabes de taille permettant l'exploration des RSs dans un contexte arabe. Elle permet ainsi de combler le manque énorme de ressources arabes dédiées aux RSs.

La deuxième contribution est consacrée à l'enrichissement du domaine des RSs arabes en proposant un nouveau système de recommandation moderne, adapté au contenu arabe. La mise en œuvre du système proposé a été réalisé en combinant des modèles indépendants d'apprentissage en profondeur (Deep Learning, DL) en un seul système, à savoir des Réseaux de Neurones Convolutifs (Convolutional Neural Networks, CNNs) avec un Perceptron Multi-couche (Multi-Layer Perceptron, MLP). Ce système a pour principaux avantages, la capacité d'améliorer efficacement la précision des prédictions, et à traiter un grand volume de données.

Des expérimentations approfondies ont été menées en exploitant les datasets arabes construits dans cette thèse. Les résultats obtenus de notre exploration et de notre proposition garantissent des aboutissements prometteurs, pouvant inspirer la communauté des chercheurs à mener des études supplémentaires dans ce sens.

Mots-clés: Big data, Arabe, Apprentissage automatique, Apprentissage profond, Traitement automatique des langues, Réseau neuronal convolutif, Systèmes de recommandation, Recommandations personnalisées, Prédiction des notes, Filtrage collaboratif, Factorisation de matrices, Avis de l'utilisateur, Préférences de l'utilisateur, Modélisation de l'utilisateur, Analyse de texte.

Abstract

In the era of big data, Recommender Systems (RSs) have become growing essential tools. They represent important Machine Learning (ML) algorithms that keep users engaged with personalized content in different e-platforms like Amazon, Netflix, YouTube and Facebook. RSs are beneficial to both users and businesses. They help users make decisions, besides assisting companies to make more profits. A large chunk of many businesses' revenue is generated from recommendations alone.

Several RSs have been proposed in the literature, and most of them have primarily focused on English content. However, research and existing resources remain very limited for content in other languages like Arabic. In recent times, the Arabic content on the Web has significantly increased because of the growing number of Arabic web users. Arabic came fourth in the top ten Internet languages. This highlights the need for exhaustive in-depth work on Arabic content, especially in the RS field.

This thesis describes the different works about RSs in general while emphasizing the lack of research on Arabic RSs. Furthermore, it presents several new contributions aiming to improve this field, little explored until now.

The first contribution explores and investigates recent RSs initially devoted to English content when applied to Arabic content. It plans to experiment with state-of-the-art RSs from three aspects: the applicability, the impact of the preprocessing, and the performance when changing the language of the content. By conducting this set of experiments, this contribution also makes four Arabic datasets available to explore RSs in an Arabic context. It thus allows addressing the considerable lack of Arabic resources devoted to RSs.

The second contribution is devoted to enriching the field of Arabic RSs by proposing a new modern recommendation system adapted to Arabic content. The implementation of the proposed system has been achieved by combining independent Deep Learning (DL) models into one system, namely Convolutional Neural Networks (CNNs) with a Multi-Layer Perceptron (MLP). The main advantages of this system are its capacity to effectively improve the accuracy of predictions and deal with a large volume of data.

Extensive experiments have been carried out exploiting the Arabic datasets constructed in this thesis. The results obtained from our exploration and our proposal ensure promising findings, inspiring the research community to carry out additional studies in this direction.

Keywords: Big data, Arabic, Machine Learning, Deep Learning, Natural Language Processing, Convolutional neural network, Recommender systems, Personalized recommendations, Rating prediction, Collaborative filtering, Matrix factorization, User reviews, User preferences, User modeling, Text analysis.

Related Publications

This thesis is partially based on the following peer-reviewed publications:

International journals

- **Mehdi Srifi**, Ahmed Oussous, Ayoub Ait Lahcen, Salma Mouline. "*A Novel Deep learning based Recommender System for Arabic Content*". Journal of Intelligent & Fuzzy Systems. Under review (modifications have been requested).

- **Mehdi Srifi**, Ahmed Oussous, Ayoub Ait Lahcen, Salma Mouline. "*Evaluation of recent advances in recommender systems on Arabic content*". J Big Data 8, 35 (2021). https://doi.org/10.1186/s40537-021-00420-2.

- **Mehdi Srifi**, Ahmed Oussous, Ayoub Ait Lahcen, Salma Mouline. 2020. "*Recommender Systems Based on Collaborative Filtering Using Review Texts—A Survey*" Information 11, no. 6: 317. https://doi.org/10.3390/info11060317.

International conferences

- **Mehdi Srifi**, Ahmed Oussous, Ayoub Ait Lahcen, Salma Mouline. "*Evaluating Recommender Systems when Applied to Arabic Content*". International Symposium on Advanced Electrical and Communication Technologies (ISAECT), 2020, pp. 1-6, doi: 10.1109/ISAECT50560.2020.9523708.

- **Mehdi Srifi**, Badr Ait Hammou, Ayoub Ait Lahcen, Salma Mouline. "*Collaborative Recommender Systems Based on User-Generated Reviews: A Concise Survey*". International Symposium on Advanced Electrical and Communication Technologies (ISAECT), 2018, pp. 1-6, doi: 10.1109/ISAECT.2018.8618822.

- **Mehdi Srifi**, Badr Ait Hammou, Ayoub Ait Lahcen, and Salma Mouline. "*A Concise Survey on Content Recommendations*". In : International Conference on Big Data, Cloud and Applications. Springer, Cham, 2018. p. 393-405.

Contents

List of Figures

List of Tables

List of Acronyms

A3NCF	Adaptive Aspect Attention-based Neural Collaborative Filtering
ALFM	Aspect-Aware Latent Factor Model
BMF	Biased Matrix Factorization
CARL	Context-Aware user-item Representation Learning
CARP	CApsule network based model for Rating Prediction
CBF	Content-Based Filtering
CF	Collaborative Filtering
CNN	Convolutional Neural Networks
DL	Deep Learning
FM	Factorization Machines
GMF	Generalized Matrix Factorization
GloVe	Global Vectors for Word Representation
LDA	Latent Dirichlet Allocation
LFM	Latent Factor Model
LSA	Latent Semantic Analysis
MAE	Mean Absolute Error
MF	Matrix Factorization
MLP	Multi-Layer Perceptron
ML	Machine Learning
MT	Machine Translation
MSE	Mean Squared Error
NLP	Natural Language Processing
NMF	Non-negative Matrix Factorization

NeuMF	Neural Matrix Factorization
OM	Opinion Mining
PARL	Pair-dependent features from Auxiliary Reviews written by Like-minded users
PCC	Pearson Correlation Coefficient
PLSA	Probabilistic Latent Semantic Analysis
PMF	Probabilistic Matrix Factorization
RMSE	Root Mean Squared Error
RSs	Recommender Systems
RS	Recommender System
ReLU	Rectified Linear Unit
S3VM	Semi-Supervised Support Vector Machine
SA	Sentiment Analysis
SGD	Stochastic Gradient Descent
SVD	Singular Value Decomposition
SVM	Support Vector Machine
TF-IDF	Term Frequency-Inverse Document Frequency

Part I

Opening

Chapter 1

Introduction- in French

Contents

1.1 Contexte de la recherche

À l'ère du big data, le volume d'informations sur le Web a augmenté à un rythme sans précédent [OBLB18]. Bien que la disponibilité de données à grande échelle puisse être bénéfique, elle peut également rendre le processus de prise de décision assez difficile. En fait, les internautes sont submergés d'innombrables informations, ce qui provoque un problème crucial connu sous le nom de surcharge d'informations [AT05]. De ce fait, il devient difficile pour les utilisateurs d'accéder à ce qui les intéresse sur le Web au moment opportun. Des Systèmes de Recommandation (RSs, de l'anglais Recommender Systems) ont émergé pour affronter ce problème en collectant de manière autonome des informations et en les adaptant de manière proactive aux préférences personnelles [CCW15], par exemple, quel article consommer (Amazon), dans quel hôtel séjourner (TripAdvisor), quelle chanson écouter à (Last.fm), etc. Les RSs sont avantageux à la fois pour les utilisateurs et les fournisseurs de services [PCH11]. Ils améliorent le processus de prise de décision des utilisateurs et permettent aux fournisseurs de services de faire plus de profits dans diverses applications Web telles que le commerce électronique, le tourisme électronique, les réseaux sociaux, les sites de films, etc [LWM$^+$15].

Le but principal des RSs est de fournir des suggestions qui répondront le plus probablement aux préférences de l'utilisateur en prédisant son intérêt pour un produit/service cible en fonction de ses préférences passées, c'est-à-dire ses intérêts, ses goûts ou ses besoins [BOHG13]. De telles préférences sont explicitement énoncées par l'utilisateur ou sont déduites des interactions passées entre l'utilisateur et les articles précédemment consommés [HRCB19], respectivement, des évaluations numériques (alias ratings) [HKBR99] et des relevés de consommation [HKV08]. Il existe cependant plusieurs modèles qui peuvent être adoptés pour modéliser les préférences de l'utilisateur à partir des données. Les RSs génèrent principalement des prédictions de pertinence des articles en se basant sur les modèles construits.

Généralement, il existe deux principaux types d'approches de recommandation, à savoir, le Filtrage Basé sur le Contenu (CBF, de l'anglais Content-Based Filtering) et le Filtrage Collaboratif (CF, de l'anglais Collaborative Filtering) [AT05, BOHG13, IFO15]. L'approche basée sur le contenu (CBF) vise à produire des recommandations en impliquant les informations individuelles des utilisateurs et le contenu descriptif des éléments. La seconde est connue sous le nom de CF, elle représente l'une des approches les plus réussies pour les RSs. L'idée de base de la méthode CF est de produire des recommandations aux utilisateurs en modélisant leurs préférences à partir d'une matrice d'évaluation des éléments par les utilisateurs. L'approche CF fonctionne bien lorsque la matrice de notes est suffisamment renseignée. Cependant, leur performance chute de manière significative en cas de rareté de notes, en raison de la faible couverture de l'espace de recommandation, ou de la difficulté rencontrée par les utilisateurs à évaluer un élément donné, en utilisant seulement des scores numériques [CCW15].

De nos jours, il existe de nombreuses applications très connues, telles que les sites de médias sociaux et de commerce électronique où les utilisateurs évaluent les articles par une notation numérique et/ou par des commentaires personnels [HRCB19]. De telles commentaires se présentent généralement sous forme textuelles qui expriment la raison pour laquelle les utilisateurs apprécient ou n'apprécient pas les éléments évalués. Comparativement avec la notation numérique, les commentaires textuels contiennent plus d'informations sémantiques, fournissant ainsi aux RSs des informations plus fines, nuancées et fiables sur les préférences de l'utilisateur [MJZ03]. En conséquence, les RSs peuvent construire une représentation bien détaillée sur les préférences des utilisateurs, que la notation numérique ne permet d'obtenir [CCW15]. Ces préférences, extraites des critiques écrites des utilisateurs, peuvent également être exploitées par les vendeurs/entreprises afin d'améliorer leurs produits et services [VKVTL05].

Au fil du temps, l'arabe est devenu la quatrième langue la plus parlée dans le monde et l'une des langues les plus utilisées sur Internet (Top Ten Languages Used in the Web - March 31, 2020). De nombreuses personnes parlent la langue arabe dans de nombreux pays, avec plus de 420 millions de locuteurs (Complete List of Arabic Speaking Countries – 2020 Update). La population arabe représente environ 5,6% de la population mondiale et environ 4,8% des internautes (Arabic Speaking Internet Users And Population Statistics - 2017). D'autre part, les internautes arabes sont récemment devenus d'importants consommateurs de services Internet. Par conséquent, ils partagent de nombreux contenus, tels que des critiques textuelles, contenant leurs opinions et leurs préférences [OBLB20]. Ainsi, il est devenu possible d'exploiter un tel contenu par les RSs pour gérer le problème bien connu de surcharge d'informations dans un contexte arabe.

1.2 Motivations et Problématiques

Avec l'augmentation de l'utilisation d'Internet, une grande quantité de contenu généré par les utilisateurs est produite, à savoir les notes, les critiques et les commentaires. Cependant, dans de nombreux sites de médias sociaux et de commerce électronique, les internautes partagent leurs expériences, opinions, sentiments sur un article consommé sous la forme d'un commentaire, et/ou une note numérique indiquant leurs préférences. Ces commentaires peuvent fournir une vue d'ensemble sur les articles ou des opinions spécifiques sur certaines de leurs caractéristiques. Ils constituent une source d'informations précieuses sur les préférences des utilisateurs, et peuvent être utilisés pour construire des profils fins de ces derniers améliorant ainsi la personnalisation des recommandations [CCW15]. Les résultats empiriques des études sur le comportement des utilisateurs ont également révélé l'efficacité des avis textuels sur les processus de prise de décisions des nouveaux utilisateurs [KS07, CM06].

Récemment, des efforts croissants ont été déployés pour intégrer les informations précieuses contenues dans les avis textuels dans le processus de modélisation des utilisateurs et de génération de recommandations [SOALM20, CCW15]. Les résultats expérimentaux de ces études ont montré l'influence positive du contenu généré par les utilisateurs, tels que les avis textuels, sur les performances de précision des recommandations. Cependant, la plupart de ces travaux ont été réalisés pour démontrer l'avantage de l'incorporation de ces avis dans le contexte anglais. Plus précisément, ils ont démontré des performances élevées sur divers datasets de commentaires en anglais pour les RSs, tels que Yelp [1], Amazon [2], IMDB [3], Beer [4] ... En conséquence, une variété de systèmes de haute qualité et d'outils avancés sont désormais disponibles pour le contenu en anglais. Néanmoins, pour gérer le contenu généré par l'utilisateur écrit dans d'autres langues telles que l'arabe, très peu de RSs ont été proposés [ZAS+17, HASA20]. L'accent a été principalement mis sur l'anglais. Les outils, les ressources et la recherche sur le contenu

[1] https://www.yelp.com/dataset

[2] https://jmcauley.ucsd.edu/data/amazon/

[3] https://www.imdb.com/interfaces/

[4] https://data.world/socialmediadata/beeradvocate

dans d'autres langues, en particulier la langue arabe, sont très rares. Notons que dans la littérature, il y a absence de datasets contenant des avis textuels arabes disponibles et principalement conçu pour enquêter et valider les RSs. Un examen des études précédentes menées dans ce domaine révèle que les chercheurs ont exploré leurs propositions de RSs uniquement sur des datasets de commentaires arabes initialement conçus pour l'Analyse de Sentiments (SA, de l'anglais Sentiment Analysis) et non à des fins de recommandation. Ce fait est considéré comme l'une des principales raisons qui nous ont motivés à créer des datasets d'avis en langue arabe pour les RSs dans cette recherche. D'un autre côté, il convient également de mentionner que les travaux examinés souffrent d'une variété d'autres limitations communes. Par exemple, les datasets arabes utilisés dans ces travaux sont de taille modeste par rapport aux datasets à grande échelle utilisés pour enquêter sur les RSs anglais. En outre, aucun de ces travaux n'a utilisé plusieurs datasets arabes pour évaluer convenablement son RS dans le contexte arabe. Conséquemment, une étude empirique plus approfondie est requise. De même, ces travaux n'utilisent que des paradigmes de recommandation traditionnels, contrairement aux modèles modernes et performants de recommandation adoptés dans les études consacrées aux contenus en anglais. En outre, les auteurs ont effectué des analyses de traitement de texte très simples pour impliquer les avis textuels dans le processus de recommandation. De telles analyses sont considérées comme dépassées dans la littérature des RSs anglais, car elles ne permettent pas d'extraire des informations importantes telles que la sémantique.

Par conséquent, à partir de la littérature examinée, nous pouvons conclure quelques notes critiques. Les RSs arabes ont reçu une attention infime dans la littérature de recherche. Le pourcentage d'œuvres qui proposent des RSs arabes est très limitée. Par ailleurs, les œuvres disponibles ne sont pas encore matures par rapport à celles présentant des RSs anglais. Ainsi, ils ne permettent pas de déduire des conclusions bien argumentées sur l'exploitation du contenu arabe par les avancées récentes dans le domaine des RSs. Dès lors, la communauté des chercheurs a encore besoin d'efforts de recherche pour proposer des études et des outils complèmentaires pour s'attaquer aux problèmes susmentionnés.

Nous optons pour la langue arabe dans ce travail pour plusieurs raisons. D'une part, la langue arabe est bien répandue dans divers pays et utilisée par des millions de personnes dans le monde [OLB18]. Elle, appartient à l'une des six langues officielles des Nations Unies [EE16]. D'autre part, l'utilisation massive des services électroniques par les utilisateurs arabes s'est accompagnée d'une croissance significative du contenu généré par les utilisateurs, tels que des critiques en ligne contenant leurs opinions sur différents sujets, produits ou services [AKGA$^+$14]. Ainsi, pour bénéficier d'un tel contenu, la communauté des chercheurs est appelée à proposer des RSs arabes efficaces.

L'amélioration dans le domaine des RSs arabes jouerait un rôle essentiel pour plusieurs acteurs. Les RSs pourraient être intégrés dans différentes applications arabes telles que les magasins en ligne, l'apprentissage en ligne, les réseaux sociaux, les voyages en ligne, etc. Les fournisseurs de services dans de telles applications pourraient bénéficier des RSs pour augmenter leur trafic et leurs revenus en offrant aux utilisateurs des choix/services potentiellement fascinants de manière personnalisée. En outre, les internautes peuvent profiter de ces services dans leurs activités quotidiennes, à savoir regarder des films, écouter de la musique et acheter des articles. Toutes ces améliorations prouvent l'importance des RSs arabes, mettant ainsi la lumière sur la nécessité d'avoir des études complémentaires dans ce domaine.

1.3 Objectifs et Contributions

Cette thèse vise à explorer et à améliorer la recommandation personnalisée, en particulier la tâche de prédiction de notes dans le contexte arabe tout en palliant aux limites des travaux connexes existants. Pour répondre à cet objectif majeur, la présente thèse comprend deux contributions. Dans ce qui suit, nous présentons un résumé de chacune de ces contributions.

1.3.1 Exploration des systèmes de recommandation avancés dans le contexte arabe

Le domaine des RSs arabes souffre d'un manque de ressources, d'intérêt et de recherche. Les travaux connexes sont très peu nombreux, et les études y afférant sont limitées [SOLM21]. En fait, les investigations sur les RSs lors de l'exploitation du contenu en langue arabe sont minimes. En outre, ceux disponibles n'explorent pas et n'enquêtent pas sur les avancées récentes dans le domaine des RSs. Conséquemment, nous visons à travers cette étude à combler cette lacune de recherche en tirant profit des paradigmes actuels dans les RSs. Le but essentiel est de mener une étude empirique complète pour tirer des conclusions bien argumentées sur l'application des RSs modernes dans le contexte arabe. A ce niveau, notre travail se propose d'examiner les questions de recherche suivantes :

- Est-il possible d'appliquer les RSs récents dans le contexte arabe ?

- Si oui, le contenu arabe a-t-il besoin d'une étape particulière de prétraitement pour l'incorporer dans ces RSs?

- Dans l'affirmative, l'application de ces RSs au contenu arabe donne-t-elle de bons résultats comme ceux obtenus lors de leur application au contenu anglais ?

Pour répondre à ces questions, nous testons la tâche de prédiction de notes à partir de trois perspectives :

1) Perspective d'applicabilité : Ceci est accompli en réalisant une expérience avec cinq RSs de pointe de la littérature pour effectuer une prédiction de notes en exploitant des revues textuelles arabes. Nous signalons que la performance de ces systèmes a été démontrée pour le contenu anglais; néanmoins, elle n'a jamais été prouvé pour le contenu en langue arabe. L'efficience de chaque RS est étudiée par rapport aux datasets construits. L'idée est de vérifier l'applicabilité des RSs récents dans le contexte arabe.

2) Perspective d'impact du prétraitement : à ce stade, nous visons à déterminer s'il est nécessaire d'appliquer une phase de prétraitement particulière sur le contenu arabe avant de l'intégrer dans les RSs adoptées. Ainsi, nous testons ces RSs sur les datasets construits dans deux cas : avec et sans prétraitement.

3) Perspective de performance : elle consiste à effectuer des expériences avec les cinq moteurs de recommandation adoptés pour évaluer et comparer leurs performances en variant la langue des commentaires de l'anglais à l'arabe. Cette expérimentation permet notamment de vérifier si les différences de performances des modèles de recommandation utilisés sont statistiquement significatives ou non lors du changement de la langue du contenu. Le principe est de confirmer si l'application des RSs récents au contenu arabe fournit une bonne précision de prédiction de notes comme celle atteinte lors de leur application au contenu anglais.

En réalisant cet ensemble d'expérimentations, l'apport de cette partie de notre thèse réside dans les aspects suivants :

- Mettre la lumière sur le besoin significatif d'avoir des RSs dédiés au contenu arabe.

- Construire et mettre à disposition quatre nouveaux datasets arabes de taille pour les RSs. L'objectif principal est d'évaluer les RSs sur des données arabes à grande échelle.

- Appliquer les RSs récents dans le contexte arabe.

- Développer un nouveau mécanisme de prétraitement des revues en arabe et en anglais.

- Étudier l'effet de la phase de prétraitement sur les performances des RSs récents lors de l'utilisation de textes arabes.

- Analyser et évaluer les performances de divers RSs de pointe lorsqu'ils sont appliqués au contenu arabe.

- Comparer les performances des RSs récents dans les contextes anglais et arabe (en utilisant le même contenu dans deux langues différentes)

- Démontrer que les RSs modernes fournissent des résultats prometteurs lorsqu'ils sont appliqués au contenu arabe.

1.3.2 Construction d'un nouveau système de recommandation basé sur le deep learning adapté au contenu arabe

Des travaux récents dans le domaine des RSs proposent d'analyser les avis textuels pour extraire des informations indispensables pouvant aider à la prédiction de notes, et surmonter l'inconvénient de la rareté des données [CN21, SOALM20, CCW15]. De telles réalisations ont montré des succès remarquables et ont prouvé de meilleures performances par rapport aux méthodes classiques des RSs. Toutefois, tous ces progrès actuels sont dévolus pour les travaux concernant uniquement les commentaires en langue anglaise. Les travaux existants exploitant les avis en langue arabe pour la prédiction de notes ne bénéficient pas d'outils et de ressources avancés. Ils n'utilisent que des architectures traditionnelles des RSs et des techniques d'analyse de textes classiques qui sont actuellement considérées comme dépassées par la communauté des chercheurs. A cet effet, nous envisageons à travers cette étude de combler ce manque de travaux en proposant un RS moderne et efficace consacré au contenu arabe. La nouveauté de cette étude repose sur l'exploitation des progrès actuels dans le domaine des RSs. Il convient de signaler, que les questionnements abordés dans l'étude expérimentale précédente ont animé l'inspiration de cette contribution, une fois, rassuré de la garantie de l'application de ces avancées au contenu arabe.

La contribution de cette partie de la thèse reflète les aspects suivants:

- Primo, la conception d'un nouveau système de recommandation basé sur l'apprentissage profond (DL, de l'anglais Deep Learning) adapté à réaliser et améliorer la prédiction de notes dans le contexte arabe. Notre RS propose de fusionner une technique de traitement de texte basée sur des réseaux de neurones convolutifs (CNNs, de l'anglais Convolutional Neural Networks), et un modèle neuronal de prédiction, namely perceptron multicouches (MLP, de l'anglais Mutli Layer Perceptron) dans un seul système afin de prédire les notes en employant des données textuelles arabes à grande échelle. Notons que ces techniques n'ont jamais été utilisées auparavant pour effectuer la prédiction de notes dans le contexte arabe. A notre humble avis, il s'agit d'une étude pionnière qui combine ces deux modèles DL et les explore dans un contexte arabe. Elle est d'ailleurs, la première proposition dédiée à traiter un grand volume de données arabes.

- Secundo, effectuer des expériences complètes sur quatre datasets arabes pour évaluer et prouver la performance du système proposé en terme de précision des prédictions. Les résultats expérimentaux démontrent que notre RS peut atteindre une précision nettement meilleure que d'autres techniques de pointe. En particulier, notre RS améliore l'erreur quadratique moyenne (MSE, de l'anglais Mean Squared Error) entre 0,84% et 16,96%, et l'erreur absolue moyenne (MAE, de l'anglais Mean absolute Error) entre 0,14% et 13,71%.

1.4 Structure de la thèse

La suite de cette thèse est structurée comme suit:

Partie II Principaux concepts et revue de la littérature (Background and Literature review) : Cette partie comprend les chapitres 3 et 4, consacrés à présenter quelques concepts de base et l'état de l'art des travaux connexes.

Partie III Notre proposition (Our proposal) : Cette partie comprend deux chapitres 5 et 6, présentant les deux contributions faisant l'objet de cette thèse. Le chapitre 5 expose la méthodologie adoptée pour explorer dans un contexte arabe, l'application des paradigmes modernes dans le domaine des RSs. Dans un premier temps, il présente en détail tous les moyens et étapes mis en œuvre pour la réalisation de cette étude empirique approfondie. Ensuite, il expose, analyse et résume les résultats empiriques obtenus. Le chapitre 6 présente notre nouvelle proposition portant sur le développement d'un RS arabe consacré à effectuer et améliorer la tâche de prédiction de notes dans un contexte arabe. Il détaille les composants utilisés dans notre système proposé, la méthodologie suivie et les expériences réalisées pour évaluer cette proposition. Ce même chapitre discute enfin les résultats obtenus.

Partie IV Clôture (Closing) : Cette partie comprend le chapitre 7, qui résume les principales conclusions et limites, et fournit un certain nombre de perspectives dans le domaine en question.

Chapter 2

Introduction- in English

Contents

2.1 Research Context

In the era of big data, the volume of information on the Web has grown at an unprecedented rate [OBLB18]. Although the availability of large-scale data can be beneficial, it can also make the decision-making process more difficult. In fact, Internet visitors are submerged with countless information, which causes a crucial issue known as information overload [AT05]. Thereby, it becomes difficult for users to access what interests them on the Web at the right moment. Recommender systems (RSs) have emerged to manage this issue by autonomously collecting information and proactively adapting it to personal preferences [CCW15], e.g., what item to consume (Amazon), which hotel to stay in (TripAdvisor), what song to listen to (Last.fm), etc. RSs are advantageous to both users and service providers [PCH11]. They improve users' decision-making process and allow service providers to make more profits in various web applications such as e-commerce, e-tourism, social networks, movie sites, etc [LWM$^+$15].

The mainstream of RSs is to provide the items that will most likely meet the user's preferences by predicting its interest in a target item according to its past preferences, i.e., interests, tastes, or needs [BOHG13]. Such preferences are explicitly stated by the users or are inferred from past user-item interactions [HRCB19], commonly numeric evaluations (a.k.a. ratings) [HKBR99] and consumption records [HKV08], respectively. There are, however, several models that can be adopted to perform user preferences modeling from data. RSs mainly generate item relevance predictions based on the learned models.

Generally, there are two principal types of recommender approaches; namely, Content-Based Filtering (CBF) and Collaborative Filtering (CF) [AT05, BOHG13, IFO15]. The content-based approach (CBF) aims to produce recommendations based on the individual information of users and the content representations of items. The second one is known as CF, and it represents one of the most successful approaches for RSs. The basic idea of the CF method is t to produce recommendations to users by learning users' preferences from user-item rating patterns. CF approach operates well when there is good enough rating information. However, their performance drops significantly when the rating sparsity issue happens, thanks to the poor coverage of recommending space or the difficulty in letting users express their usage experiences for a given item as numerical scores [CCW15].

Nowadays, there are many popular applications such as social and e-commerce media sites where users evaluate items through ratings and give personal reviews describing their assessment of items [HRCB19]. Such reviews are usually in the form of textual comments that express why the users like or dislike the evaluated items. Compared to rating information, textual reviews have more semantic information, providing RSs with more fine-grained, nuanced, and reliable user preference information [MJZ03]. Consequently, the RSs can construct a detailed preference representation for the user, which cannot be inferred from global rating scores [CCW15]. Also, the vendors may exploit the user preferences extracted from the written reviews to improve their products and services [VKVTL05].

Over time, Arabic has become the fourth most-spoken language worldwide and one of the most used languages on the Internet (Top Ten Languages Used in the Web - March 31, 2020). Many people speak the Arabic language in many countries, with more than 420 million speakers (Complete List of Arabic Speaking Countries – 2020 Update). The Arab population represents around 5.6% of the world's population and around 4.8% of internet users (Arabic Speaking Internet Users And Population Statistics - 2017). On the other hand, Arabic web users have recently become significant consumers of Internet services. Therefore, they share much content, such as textual reviews, containing their opinions and preferences [OBLB20]. Thus, it became possible to exploit such content by RSs for managing the well-known information overload issue in an Arabic context.

2.2 Motivations and Problem statements

With the surge in internet usage, a large amount of user-generated content is being produced, such as ratings, reviews, and comments. However, in many social media websites and e-commerce systems, people share their experiences, opinions, sentiments about a consumed item in the form of a review, along with a numerical rating indicating their preferences. These reviews may provide a comprehensive overview of the items or specific opinions on some characteristics of the items. Thus, they constitute a valuable information source on users' preferences and can be used to learn fine-grained profiles of users and improve personalized recommendations [CCW15]. Empirical findings from user behavior studies have also revealed the efficiency of item reviews on the decision-making processes of new users [KS07, CM06].

Recently, growing efforts have been made to integrate the precious information embedded in reviews into the process of user modeling and recommendation generation [SOALM20, CCW15]. Experimental results of these studies have shown a positive influence of user-generated content such as user reviews on the performance of recommendation accuracy. However, most of these works were done to demonstrate the benefit of review incorporation in the English context. Specifically, they have demonstrated high performance on various English-language reviews datasets for RSs, such as Yelp [1], Amazon [2], IMDB [3], Beer [4] ... As a result, a variety of high-quality systems and advanced tools are now available for English content. However, for managing user-generated content written in other languages such as Arabic, very few RSs have been proposed [ZAS+17, HASA20]. The focus has been mainly on English. Tools, resources, and research on the content in other languages, especially Arabic-language, are very scarce. We notice in the literature that there is no available Arabic reviews dataset mainly designed for investigating and validating RSs. A review of previous studies conducted in this field shows that researchers examined their proposed RSs only on Arabic reviews' datasets originally built for Sentiment Analysis (SA) and not for recommendation purposes. This fact is considered one of the main reasons that motivated us to create Arabic-language reviews datasets for RSs in this research. On the other hand, it also should be mentioned that the reviewed works suffer from other various common limitations. For instance, the Arabic datasets utilized in these works are of modest size compared to the large-scale datasets used to investigate the English RSs. Besides, none of these works has used several Arabic datasets to evaluate its proposed RS in the Arabic context properly. Thus, a more in-depth empirical study is required. Moreover, these works use only traditional recommending paradigms contrarily to modern and performant recommendation models adopted in studies devoted to English content. Furthermore, the authors also performed straightforward text processing analyses to incorporate review information into the recommending process. Such analyses cannot capture contextual information; thus, they cause semantic information

[1] https://www.yelp.com/dataset

[2] https://jmcauley.ucsd.edu/data/amazon/

[3] https://www.imdb.com/interfaces/

[4] https://data.world/socialmediadata/beeradvocate

loss, limiting modeling expressiveness.

Therefore, from the reviewed literature, we can conclude some critical notes. Arabic RSs have received minimal attention in the research literature. The percentage of works that propose Arabic RSs is very limited. Also, the available works are still not mature yet, compared with the ones presenting English RSs. Thus, they do not derive well-argued conclusions about the Arabic content's exploitation by recent advances in the RS field. Hence, the research community still needs research efforts to provide additional studies and tools to tackle the problems above.

We choose the Arabic language in this work for several reasons. On the one hand, the Arabic language is well spread among various countries and used by millions of people worldwide [OLB18]. It belongs to one of the six official languages of the United Nations [EE16]. On the other side, the huge use of the e-services by Arabic users has been accompanied by significant growth in user-generated content such as online reviews that contain users' opinions about different topics, products, or services [AKGA+14]. Consequently, to benefit from such content, the research community should propose effective Arabic RSs.

The improvement in the Arabic RS field would play an essential role for several actors. RSs could be integrated into different online Arabic applications such as e-stores, e-learning, social networks, e-travel, etc. The service providers in such applications could benefit from RSs to increase their traffic and revenue by providing users with conceivably fascinating choices/services in a personalized way. Also, Internet users may take advantage of such services in their daily activities such as watching movies, listening to music and purchasing items. All these improvements prove the significance of Arabic RSs, and they shed light on the need to have additional studies in this direction.

2.3 Aims and Contributions

This thesis aims to explore and improve personalized recommendation, mainly the rating prediction task in the Arabic context, while tackling the limitations of the existing related works. The current thesis consists of two main contributions to answer this major goal. In what follows, we summarize each of these two contributions.

2.3.1 Exploring advanced recommender systems in the Arabic context

Arabic RS field suffers from a lack of support, research, and resources. Related works are very few, and their topics are limited [SOLM21]. The research investigations about RSs when using content in the Arabic language are minimal. Besides, those available ones do not explore and investigate recent achievements in RS field. Therefore, we aim through this study to fill this research gap by leveraging current paradigms in RSs. The main goal is to conduct a comprehensive empirical study to explore the modern recommenders in the Arabic context. At this level, our work aims to examine the following research questions:

• Is it possible to apply recent RSs in the Arabic context?

• If so, does the Arabic content need a particular preprocess step to incorporate it in these RSs?

• If so, does the application of these RSs to Arabic content provide good results like when applying them to the English content?

To answer those questions, we explore the rating prediction task from three perspectives:

1) Applicability perspective: This is achieved by running an experiment with five state-of-the-art RSs from the literature to perform rating prediction by exploiting Arabic textual reviews. We notice that the performance of these systems

has been demonstrated for English content; however, it has never been proved for content in the Arabic language. The behavior of each RS is studied with respect to the constructed reviews datasets. The idea is to verify the applicability of recent RSs in the Arabic context.

2) Preprocessing's impact perspective: at this stage, we aim to determine if it is necessary to apply a particular preprocessing phase on the Arabic content before incorporating it into the adopted RSs. Thus, we tested these RSs on the built datasets within two cases: with and without preprocessing.

3) Performance perspective: It consists of running experiments with the five adopted recommender engines to evaluate and compare their performance when varying the language of reviews from English to Arabic. In particular, this experiment allows verifying if the used recommendation models' differences in performance are statistically significant or not when changing the content's language. The principle is to confirm if applying the recent RSs to Arabic content provides good rating prediction accuracy as when applying them to English content.

By conducting this set of experiments, the contribution of this part of our thesis lies in the following aspects:

- Shedding light on the significant need for research, resources, and tools in the Arabic RS field.

- Building and making freely available four new massive Arabic datasets for RSs. The main goal is to assess RSs on large-scale Arabic data.

- Applying the recent RSs in the Arabic context.

- Developing a new mechanism for preprocessing the Arabic and English reviews.

- Investigating the effect of the preprocessing phase on the recent RSs when using Arabic texts.

- Analyzing and evaluating the performance of various state-of-the-art review-based RSs when applied to Arabic content.

- Comparing the performance of recent RSs in both English and Arabic contexts (using the same content in two different languages).

- Demonstrating that modern RSs provide promising results when applied to the Arabic content.

2.3.2 Building a novel deep learning-based recommender system adapted to arabic content

Recent works in the RS field have considered analyzing review texts to extract vital information that can aid rating prediction and overcome the data sparsity drawback [CN21, SOALM20, CCW15]. Such achievements have shown remarkable successes and proved better performances over the classical RS methods. However, all this current progress is used only by the works devoted to managing reviews in the English language. Existing works exploiting Arabic-language reviews for rating prediction do not benefit from advanced tools and resources. They only use traditional RS architectures and classical text analysis techniques that are considered overwhelmed by the research community currently. Therefore, we aim through this study to fill this research gap by providing a modern RS devoted to Arabic content. The novelty of this work relies on exploiting DL techniques. We mention that our previous experimental study ignited the inspiration for this contribution. This after making sure the guarantee of applying such advances to Arabic content.

The contribution of this part of our work lies in the following aspects:

- Firstly, designing a novel DL-based RS adapted to perform and improve rating prediction in the Arabic context. Our RS proposes to fuse a CNN based text processing technique with a MLP prediction model in one system for rating prediction based on large-scale Arabic textual data. We mention that these techniques have never been used before for rating prediction in the Arabic context. To the best of our knowledge, this is the first work that combines these two DL models and explores them in an Arabic context. Besides, this is the first proposal designed to handle a large volume of Arabic data.

- Secondly, performing comprehensive experiments on four large-scale Arabic datasets to evaluate and prove the accuracy performance of the proposed RS. Experimental results show that our RS can reach significantly better prediction accuracy than other state-of-the-art techniques. Notably, it improves the MSE between 0.84% and 16.96%, and the MAE between 0.14% and 13.71%.

2.4 Thesis Structure

The remainder of this thesis is organized as follows:

Part II Background and Literature review: This part includes chapters 3 and 4 devoted to describing some background concepts and the state-of-the-art of related works.

Part III Our proposal: This part comprises two chapters, 5 and 6, presenting our two contributions. Chapter 5 sets out the methodology adopted to explore the application of modern paradigms in the RS field in the Arabic context. This chapter firstly presents in detail all the steps and resources used for realizing this comprehensive empirical study. Then, it analyses and summarizes the attained empirical findings. Chapter 6 presents our proposed Arabic RS for performing and improving rating prediction in the Arabic context. It details the used components in our proposed system, the followed methodology, and the realized experiments to evaluate this proposal. In the end, this chapter discusses the achieved results.

Part IV Closing: This part includes chapter 7, which summarizes the main conclusions, limitations, and some future research directions.

Part II

Background and Literature review

Chapter 3

Recommender Systems: Definitions, Models and Evaluation

Contents

3.1 Definitions

In the era of big data, RSs have become essential growing tools. They mainly contribute to keeping users engaged with personalized content in e-platforms. RSs are information processing systems that actively gather various kinds of data to build their recommendations. Typically, personalized RS consists of the following information [RRS15, PBD18] :

- A set of items denoted as $I = \{i_1, i_2, ..., i_{|I|}\}$, where item i represents a general abstraction that can be an instance of news, books, movies, songs, or other products.

- A set of users represented as $U = \{u_1, u_2, ..., u_{|U|}\}$, where each user u may provide its preferences on items.

- A set of ratings denoted as $R \subseteq |U| \times |I| \rightarrow D$, where a rating $r_{u,i}$ expressed as a value in a rating domain D, indicating the preference degree of a particular user u about a specific item i. Ratings can have many forms depending on the system in question. Generally, it is frequently a value from one to five (1-5 score). Each scale represents how much a user is interested in an item. For instance, a 5-star rating system where 1 means "Hated it", 2 means "Didn't like it", 3 means "It was ok", 4 means "Liked it" and 5 means "Loved it" [A$^+$16]. Implicit feedback represents another form of ratings. This form of user evaluation is not direct. It can be extracted from user actions, such as buying or browsing an item. Dealing with this data needs special treatment because it is not as direct as a personal rating [Qub15]. On the other hand, users may express their opinions in the form of text descriptions. In such cases, implicit ratings can be extracted from these opinions based on opinion mining and sentiment analysis techniques [LCT21].

The tuple (U, I, R) represents the core of RSs. In real-world applications, the users do not provide their opinions for all items, i.e., $r_{u,i}$ is unknown for most pairs (user u, item i), which is the principal cause of the data sparsity issue [CHN+17]. The set of values (U, I, R) results in a sparse matrix called a rating matrix. One example of a rating matrix is given in Table 3.1. The RSs use this matrix as a starting point and then performs one of the following two main tasks:

Rating Prediction: Given a user u and an item i for which $r_{u,i}$ is unknown. This task aims to predict the preference $\hat{r}_{u,i}$ that the user u is likely to give to the item i. After that, items are ranked according to the estimated rating scores, and then items with high ranking are suggested to the targeted user u.

Personalized Ranking: Given a user u, this task aims to find the best-ranked list of n pertinent items ($top{-}n$ items) for the user needs according to their historic preferences.

Table 3.1: An example of rating matrix [AT05].

User/Item	Item 1	Item 2	Item 3	Item 4
User 1	4	∅	2	∅
User 2	∅	∅	5	5
User 3	2	∅	4	∅
User 4	3	∅	∅	∅

3.2 Famous Recommender Systems

3.2.1 Ringo

Ringo [SM95, CC05] is one of the pioneers of music RSs, which recommends music albums and artists. Each user is requested to rate a list of artists according to how much he likes listening to them in this system. The collected ratings constitute the personal profile of the target user. Ringo evaluates the interest of a target user for a music object based on the ratings for this object by other users, called neighbors, that have similar historical rating patterns. The system then computes the weighted average of the ratings considered to recommend the music objects.

3.2.2 Amazon.com

Amazon.com (www.Amazon.com) is a pioneer e-commerce platform that uses a recommender engine to sell diverse categories of products such as books, electronics, CDs, software, and so on. In Amazon.com, the system constructs a specific profile for each user based on the explicit ratings, buying history, and browsing history. The explicit ratings in Amazon.com can take on values from 1-star to 5-star. However, the browsing and buying patterns are constructed when users are logged in with their personal Amazon accounts [A+16]. The system compares the available users' profiles in the platform to recommend the item of interest to the target user. Amazon.com popularized the feature of "people who bought this item also bought these items" [IFO15].

3.2.3 Netflix

Netflix (www.Netflix.com) is an online movie rental company that allows users to rent movies for a monthly fee. It provides users with a personalized collection of films that they wish to see in a given order of priority. Television shows and movies are delivered to users via streaming. Using a 5-point scale system, Netflix allows users to rate different films. Personalized recommendations are then made by considering various factors, including the genres of movies and TV shows, the previous ratings of the user with the streaming history, and the combined ratings of all Netflix members who have similar tastes to the user. By helping users find appropriate content to watch, Netflix has proved its efficiency in the recommendation setting. It also was behind significant advancement in the RS field due to the Netflix Prize competition. More details about the Netflix RS are presented in [TH05, A+16].

3.2.4 Google News

Google News (news.google.com) is an online information system that is intended to suggest relevant news articles to users from sources around the world. The recommendation approach used by this system is based on the click history of users [JZFF10, DDGR07]. By using identification mechanisms enabled by Gmail accounts, the system associates history of clicks to each active user [A+16]. A user clicks on news articles is considered as a positive vote. The votes in this system are implicit, as they are deduced from user actions instead of being directly provided by the users. One of the significant challenges of Google News recommender relies on the fact that this system has to wait several hours to collect enough clicks to suggest the news articles to users, resulting in undesirable time shifts between break-out news and suggestions [JZFF10].

3.3 Recommender Systems: Traditional Methods

RSs have been studied and used in a variety of domains, including the information retrieval [RNBY99], the Internet [WBC07], e-commerce [SKR01], and many others. The main goal of RSs can be viewed as a prediction of the utility for non-viewed items by the target user. To achieve it, a recommendation approach is, therefore, needed. In general, there are two principal types of approaches [RRS15]: Collaborative Filtering (CF) and Content-Based (CB). The content-based approach attempts to predict how users will rate a set of items based on their personal information and the features they liked in the past. The second type relies on the CF, which regroups the most studied approaches of recommendation that have advanced and expanded over the years [Mit16]. CF approaches ignore user and item features but involve user-item interactions to estimate the user preferences. They are the pure behavior-based recommendation techniques [Sid18]. Existing CF can be classified into memory and model-based techniques [A+16, CHC+18]. The latter techniques are arguably the most adopted ones nowadays.

The memory-based methods (also called Neighborhood-based) generate predictions based on the relationships and similarities between users or items. These relations are inferred from the user-item rating matrix managed by the system [CHC+18]. Instead, model-based techniques exploit the values of the user-item rating matrix to build a model offline, which is then utilized to infer the pertinence of novel items for the target users [LHM+14]. The majority of these methods involve data mining and machine learning approaches for developing their predictive models [YST+04].

This section gives an overview of well-known recommendation methods, namely, Neighborhood-Based CF and model-based CF.

3.3.1 Memory-based recommendation methods

The memory-based CF method is subclustered into two main classes, namely, user-based and item-based methods [DK11, BOHG13]. The user-based CF predicts the unknown ratings of the user on the target items based on ratings of similar users on given items. Formally, the rating prediction of the user u to the item j is calculated as follows:

$$\hat{r}_{u,i} = \bar{r}_u + \frac{\sum_{v \in N_u} sim(u,v) \times (r_{v,i} - \bar{r}_v)}{\sum_{v \in N_u} |sim(u,v)|}, \tag{3.1}$$

where $\bar{r}_u$ refers to the average rating of user u, $sim(u,v)$ is the similarity (for a predefined similarity metric) of the users u and v, and N_u represents a group of users similar to user u (neighbors) who rated item i.

The item-based CF relies on the similarities between items. It predicts the user's rating for an item based on the user's ratings for similar items. In these techniques, two items are identical if multiple users have evaluated these items similarly [RRS15]. The rating prediction for item-based CF is formulated as follows:

$$\hat{r}_{u,j} = \frac{\sum_{k \in N_i} sim(j,k) \times r_{u,k}}{\sum_{k \in N_i} |sim(j,k)|}, \tag{3.2}$$

where N_i is the group of similar items to item j, and $Sim(j,k)$ is the score of the similarity between the two items j and k.

The calculation of similarity among users/items constitutes a critical stage in neighborhood-based CF techniques, as it may severely decrease their accuracy and performance [DK11]. Several similarity metrics have been presented in the literature [SZL+19], among which cosine measure [LSY03], Pearson Correlation Coefficient (PCC) [HKTR04], and Jaccard coefficient [KBGM09] are ones of the popular standard criteria typically adopted for finding most similar users or items. PCC computes the similarity based on the linear correlation between two rating vectors of users/items. The cosine metric calculates the similarity by using the angle's cosine value between rating vectors. Jaccard similarity considers the number of common ratings between users/items and ignores the rating values. The choice of the similarity measure should be properly made based on the target dataset [RS20]. To calculate the similarity measure between two users u and v respectively, these metrics are based on the following expressions:

$$sim^{PCC}(u,v) = \frac{\sum_{i \in I_{u,v}} (r_{u,i} - \bar{r}_u).(r_{v,i} - \bar{r}_v)}{\sqrt{\sum_{i \in I_{u,v}} (r_{u,i} - \bar{r}_u)^2}.\sqrt{\sum_{i \in I_{u,v}} (r_{v,i} - \bar{r}_v)^2}} \tag{3.3}$$

$$sim^{Cosine}(u,v) = \frac{\sum_{i \in I_{u,v}} (r_{u,i}).(r_{v,i})}{\sqrt{\sum_{i \in I_{u,v}} (r_{u,i})^2}.\sqrt{\sum_{i \in I_{u,v}} (r_{v,i})^2}} \tag{3.4}$$

$$sim^{Jaccard}(u,v) = \frac{|I_u \cap I_v|}{|I_u \cup I_v|}. \tag{3.5}$$

In these Equations, $I_{u,v}$ denotes the items' set rated by users u and v; $\bar{r}_u$ represents the ratings' mean value of the user u, and $r_{u,i}$ represents the u's rating for the item i. I_u and I_v represent two items sets rated by users u and v respectively. On the other hand, the similarity among two items i and j is computed by involving users' ratings which have evaluated these two items:

$$sim^{PCC}(i,j) = \frac{\sum_{u \in U_{i,j}} (r_{u,i} - \bar{r}_i).(r_{u,j} - \bar{r}_j)}{\sqrt{\sum_{u \in U_{i,j}} (r_{u,i} - \bar{r}_i)^2}.\sqrt{\sum_{u \in U_{i,j}} (r_{u,j} - \bar{r}_j)^2}} \tag{3.6}$$

$$sim^{COS}(i,j) = \frac{\sum_{u \in U_{i,j}} (r_{u,i}).(r_{u,j})}{\sqrt{\sum_{u \in U_{i,j}} (r_{u,i})^2}.\sqrt{\sum_{u \in U_{i,j}} (r_{u,j})^2}} \tag{3.7}$$

$$sim^{Jaccard}(i,j) = \frac{|U_i \cap U_j|}{|U_i \cup U_j|}, \tag{3.8}$$

where $U_{i,j}$ accounts for the group of users who evaluated items i and j, and $\bar{r}_i$ reflects the average value of ratings received by the item i. U_i and U_j refer to Users sets who rated items i and j respectively.

3.3.2 Model-based recommendation methods

As the name suggests, model-based recommendation techniques rely on finding patterns in rating data [Mit16]. The basic idea of these methods is to use ML methods techniques to learn from data and then exploit the learned model to make predictions and recommendations. Some of the most popular techniques used to build model-based RS are Matrix Based techniques, Singular Value Decomposition (SVD), DL-based models [Ren10, Kor08, ZYST19].

3.3.2.1 Latent Factor Models

The latent Factor Model (LFM) is an efficient technique for model-based CF. It supposes that the user-item interaction can be computed from low-dimensional representations of users and items. LFM methods have been broadly studied, and several variants have been proposed [KBV09, ZWFM06, MS08, SKKR00, Kor08].

3.3.2.1.1 Matrix Factorization

One of the most effective paradigms of LFM is Matrix Factorization (MF) [KBV09]. Specifically, given the user-item rating matrix $R \in \mathbb{R}^{m \times n}$, the MF method factorizes this interaction matrix into two low-rank matrices $P \in \mathbb{R}^{m \times k}$ and $Q \in \mathbb{R}^{k \times n}$ with the same latent space of dimensionality k ($k < min(m,n)$), such that user-item interactions are approximated as inner product in that space [Zha19].

$$\hat{r}_{u,i} = p_u q_i^T. \tag{3.9}$$

where p_u is a row in the user-factor matrix $P \in \mathbb{R}^{m \times k}$. p_u represents a k-dimensional latent vector associated to a user u. q_i represents a column in the item-factor matrix $Q \in \mathbb{R}^{k \times n}$. q_i is a k-dimensional latent vector related to item i.

To optimize latent vectors which better predict $\hat{r}_{u,i}$, the following loss function must be minimized in such a way:

$$\min_{p_*,q_*} \sum_{(u,i) \in T} (r_{u,i} - p_u q_i^T)^2 + \beta(||p_u||^2 + ||q_i||^2), \tag{3.10}$$

where T represents the user-item (u, i) pairs for which real ratings $r_{u,i}$ are observed in training set. And β is a defined regularization parameter used to limit the overfitting of the model. In general, the minimization of the loss function (Equation 3.10) can be achieved with different techniques such as the Gradient-based or alternating least-squares [WHC$^+$17].

Another LFM widely used is the Biased Matrix Factorization (BMF) [KBV09, RRS11] which considers the user and item biases present in ratings. For instance, some users are generous and may consistently give higher ratings than others. Similarly, some items may get higher ratings than others as they are popular. To overcome these problems, the authors of [KBV09, RRS11] provide a model to integrate such biases. Formally, the model predicts the rating $r_{u,i}$ for item i for user u as:

$$\hat{r}_{u,i} = \mu + b_u + b_i + p_u q_i^T \tag{3.11}$$

where μ is the overall average rating. b_i and b_u represents, respectively, the deviations of user u and item i from μ.

3.3.2.1.2 Non-negative Matrix Factorization

Non-negative matrix factorization (NMF) [ZWFM06] is a powerful dimensionality reduction technique that belongs to the category of LFM. The main assumption of this technique is to learn an approximate decomposition of a non-negative rating matrix into two low-rank matrices. Specifically, given a non-negative matrix $R \in \mathbb{R}_+^{m \times n}$ and a factorization rank $r < min\{m, n\}$, NMF decomposes R as follows:

$$R \approx PQ \tag{3.12}$$

where $P \in \mathbb{R}_+^{m \times r}$ and $Q \in \mathbb{R}_+^{r \times n}$ refers to the two low-rank non-negative matrices.

To optimize the quality of the approximation between R and PQ, the following cost function must be minimized in such a way [HB06]:

$$||R - PQ||^2 = \sum_{(i,j)} (R_{i,j} - (PQ)_{i,j})^2 \tag{3.13}$$

There are several ways to minimize the loss function in Equation (3.13). The multiplicative update rules technique [S$^+$00] represents the most popular one due to the simplicity of its implementation. This technique proceeds as follows:

$$Q_{kj} \leftarrow Q_{kj} \frac{(P^T R)_{kj}}{(P^T PQ)_{kj}} \tag{3.14}$$

$$P_{ik} \leftarrow P_{ik} \frac{(RQ^T)_{ik}}{(RQQ^T)_{ik}} \tag{3.15}$$

3.3.2.1.3 Probabilistic Matrix Factorization

Probabilistic Matrix Factorization (PMF) was initially presented by [MS08]. This method performs the MF from a probabilistic point of view. Specifically, it adopts a probabilistic linear model with Gaussian observation noise. Given a rating matrix $R \in \mathbb{R}^{m \times n}$, there exists a small latent variable k making $R = P^T \times Q$, with $P \in \mathbb{R}^{k \times m}$,$Q \in \mathbb{R}^{k \times n}$, P and Q are unknown low dimensional matrices corresponding to the latent factors of users and items, respectively. In this case, R became a low-rank matrix with rank $d < k$. To solve the above factorization issue, PMF supposes that the conditional probability distribution of the known data in R and the prior distributions over $P \in \mathbb{R}^{k \times m}$ and $Q \in \mathbb{R}^{k \times n}$ are interpreted as follows:

$$
\begin{gathered}
p(R|P,Q,\sigma^2) = \Pi_{u=1}^{m} \Pi_{j=1}^{n} [N(R_{u,j}|P_u^T Q_j, \sigma^2)]^{I_{((u,j) \in M)}} \\
p(P|\sigma_P^2) = \Pi_{i=1}^{m} N(P_i|0, \sigma_P^2 A^l) \\
p(Q|\sigma_Q^2) = \Pi_{i=1}^{m} N(Q_j|0, \sigma_Q^2 A^l)
\end{gathered}
\tag{3.16}
$$

where $N(\chi|\mu, \sigma^2)$ refers to the Gaussian distribution probability density function with mean μ and variance σ^2 , I_{uj} represents an indicative function, which is equal to 1 if the user u rated the item i, and equal to 0 otherwise. $M = \{(u,j)|R_{ui} = 1\}$ is an index set containing the rated items in R. A^l is a l-dimensional identity matrix.

The goal of PMF is to fit each rating $R_{u,j}$ with the appropriate inner product $P_u^T Q_j$, which can be formulated as follows:

$$
L = \frac{1}{2} \sum_{u=1}^{m} \sum_{j=1}^{n} I_{u,j} (R_{u,j} - P_u^T Q_j)^2 + \frac{\lambda_P}{2} \sum_{u=1}^{m} ||P_u||_F^2 + \frac{\lambda_Q}{2} \sum_{j=1}^{n} ||Q_j||_F^2 \tag{3.17}
$$

where $\lambda_P = \sigma^2/\sigma_P^2$ and $\lambda_Q = \sigma^2/\sigma_Q^2$. $||.||_F$ is the Frobenius norm. For equation (3.17), the stochastic gradient descent technique can be utilized for iterative training to build the model.

3.3.2.1.4 Singular Value Decomposition

Singular Value Decomposition (SVD) [SKKR00] is an efficient dimensionality reduction method widely used in recommender system applications. Specifically, this method is designed to reduce the dimensionality of a rating matrix and produce low-rank matrix approximations. The main idea behind SVD is that the learned low-rank approximations can be exploited to predict the unknown user-item interactions [MR17] [A$^+$16].

Given a user-item rating matrix $R^{m \times n}$ and matrix rank parameter r, SVD factorizes R into three low-rank matrices:

$$R = P \Sigma Q^T \tag{3.18}$$

Here, P is an $m \times m$ matrix with columns representing the m orthonormal eigenvectors of RR^T. The matrix P is an $n \times n$ matrix with columns representing the n orthonormal eigenvectors of $R^T R$. The eigenvectors of RR^T and $R^T R$ are not the same but contain the same number of non-zero eigenvalues, which are identical in value. Σ is a $m \times n$ diagonal matrix in which the diagonal entries represents non-negative square roots of the r eigenvalues of $R^T R$ (or RR^T), and sorted in decreasing order of their magnitude.

Since some of the singular values are extremely small, they can be ignored. By ignoring such values, the dimension of the matrix is reduced. It became possible to approximately factorize the matrix R by involving only the eigenvectors corresponding to the $k \preceq min\{m, n\}$ largest singular values. As a result, the approximated matrix of the original one (R) is constructed as follows:

$$R_n = P_k \Sigma_k Q_n^T \tag{3.19}$$

Here, P_k, Σ_k , and Q_k d are $m \times k$,, $k \times k$, and $n \times k$ matrices, respectively. Note that, P_k, and Q_k, contain the k largest eigenvectors of $R^T R$ and RR^T, respectively. Σ_k contains the square-roots of the k largest eigenvalues.

3.3.2.1.5 SVD++

SVD++ [Kor08] is a recommendation model that incorporates both explicit and implicit feedback information to augment recommendation performance. It extends the biased LFM to integrate the implicit information to indicate user preferences, particularly when explicit information is unavailable. The predicted preference $\hat{r}_{u,i}$ for a user u on an item i, which includes implicit feedback, are computed by the following rule:

$$\hat{r}_{u,i} = \mu + b_u + b_i + q_i^T (p_u + |N(u)|^{\frac{1}{2}} \sum_{j \in N_{(u)}} y_j) \tag{3.20}$$

where $y_j \in \mathbb{R}^f$ is the implicit factor vector. $N(u)$ represents the set of items for which user u provided implicit preference. When the implicit feedback is not available, $N(u)$ can be replaced by $R(u)$, which regroups all the items rated by the user u.

3.3.2.2 Factorization Machines

Factorization Machines (FM) [Ren10] are a generic method that uses feature engineering to mimic large classes of LFM like MF, SVD, and SVD++. FM merges the superiority of LFM with the generality of feature engineering for predicting the interactions between categorical variables [Sid18]. Contrarily to the traditional LFM-based methods, which inputs a rating matrix, FM represents ratings as tuples of real-valued feature vectors and numeric target variables [A$^+$16].

Let us assume that the training data of the FM prediction problem is described by a training set $E = \{(x^{(k)}, y^{(k)}), k = 1, ..., |E|\}$ consisting of $|E|$ instances. Each $x^{(k)} \in \mathbb{R}^n$ denotes the real-valued feature vector of an instance k. Each $y^{(k)} \in \mathbb{R}$ represents the prediction target of an instance k.

Then, for a given instance (x, y), the rating prediction $\hat{y}_{FM}$ of second-order factorization machines is computed by using pairwise interactions between the factors as follows:

$$\hat{y}_{FM}(x) = w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n} \sum_{i'=i+1}^{n} (v_i \cdot v_{i'}) x_i x_{i'} \tag{3.21}$$

where $w_0 \in \mathbb{R}$, $w \in \mathbb{R}^n$, and $V \in \mathbb{R}^{n \times k}$ represent the model's parameters to be learned. w_0 is the global bias. w_i denotes the interaction of the i-th feature to the target. x_i denotes i-th element in the feature vector x. The $v_i \cdot v_{i'}$ term refers to the factorized interaction of i and i'. $v_i \in \mathbb{R}^k$ is the latent vector for feature i, and k is the size of the latent vector.

3.3.2.3 Deep learning based methods

DL-based recommendation methods have achieved remarkable success by overcoming obstacles of traditional techniques and achieving high recommendation quality [BYBK19] [ZYST19]. In what follows, we briefly introduce some of the well-known DL-based RSs.

3.3.2.3.1 Generalized Matrix Factorization

Generalized Matrix Factorization (GMF) [HLZ$^+$17] represents a simple non-linear generalization of MF. It can be considered as an improved variant of the original MF model. This method mainly combines the strengths of neural networks with MF for predicting user preferences. GMF first uses an embedding layer to derive latent vectors of user and item based on the one-hot encoding representations. Then, it computes the element-wise product of the user and item latent vectors and outputs the interaction vector to a fully connected neural layer for rating prediction.

Formally, given a user-item pair (u, i), GMF encodes user u and item i using the one-hot representation, i.e., $x_u \in \mathbb{R}^{m \times 1}$ and $y_i \in \mathbb{R}^{n \times 1}$. In this representation vectors x_u and y_i, only the u-th and i-th entries are respectively equal to 1. Then, by applying a lookup layer, the one-hot user and item vectors are changed into a latent vectors as follows:

$$\begin{aligned} p_u &= P^T x_u \\ q_i &= Q^T y_i \end{aligned} \tag{3.22}$$

Here $P \in \mathbb{R}^{m \times k}$ and $Q \in \mathbb{R}^{n \times k}$ denote the latent factor matrix for users and items, respectively. Then, for effectively modeling the overall structure of user-item interaction, an interaction function using a linear kernel model is applied as follows:

$$\Phi(p_u, q_i) = p_u \odot q_i \tag{3.23}$$

where $\odot$ denotes the element-wise product of vectors. Finally, to predict an unknown user-item interaction, GMF projects the resultant interaction vector to a prediction layer:

$$\hat{r}_{ui} = a_{out}(h^T (p_u \odot q_i)) \tag{3.24}$$

where a_{out} denotes the activation function, h^T denotes the weights related to the prediction layer, and $\hat{r}_{ui}$ denotes the predicted ratings. Note that, GMF become equivalent to MF, particularly, when a_{out} is an identity function and h is a uniform vector of 1. However, in the case when a_{out} is a non-linear activation function, and h is learned from training data, GMF generalizes MF to a non-linear setting which has greater learning capability than original MF.

3.3.2.3.2 Multi-Layer Perceptron

In most cases, the interactions between users and items are complex to capture. The previous method (GMF) usually assumes linear user-item interactions by decomposing the rating matrix, like the standard MF. This setting does not allow modeling high-level user-item interactions. To overcome this problem, a neural recommendation method has been proposed in [HLZ$^+$17], which adopts a standard MLP for learning the non-linear and non-trivial interactions between user and item latent features. This method uses the same approach as GMF to extract user and item latent factors. However, MLP takes a different learning strategy in the neural CF module. Rather than processing user and item latent vectors as in GMF, MLP concatenates them first, then adopts the standard MLP [GD98] to learn user-item interactions (interaction function). The CF aspect of MLP can be defined as follows:

$$\begin{aligned} z_1 = \Phi(p_u, q_i) &= \begin{bmatrix} p_u \\ q_i \end{bmatrix} \\ \Phi_2(z_1) &= \sigma_2(W_2^T z_1 + b_2) \\ &\cdots \\ \Phi_L(z_{L-1}) &= \sigma_{L-1}(W_L^T z_{L-1} + b_L) \\ \hat{r}_{ui} &= \sigma(h^T(\Phi_L(z_{L-1})) \end{aligned} \tag{3.25}$$

where W_x , b_x , and σ_x denote the weight matrix, bias vector, and activation function for the x-th layer's perceptron, respectively.

3.3.2.3.3 Neural Matrix Factorization

Neural Matrix Factorization (NeuMF) [HLZ$^+$17] is an end-to-end recommendation method that combines MF and MLP models. NeuMF can build a dual neural network for modeling the two-way interaction between users and items. The main assumption of this method is to model the high-order feature interactions via deep structure and low-order interactions with MF. Specifically, MLP helps introduce non-linearity while GMF retains the advantages of MF, which relies on the inner product operation to capture the linear and pairwise interactions between features. The formulation of NeuMF's predictive model is defined as follows:

$$\begin{aligned}
\Phi^{MF} &= p_u^G \odot q_i^G \\
\Phi^{MLP} &= a_L(W_L^T(a_{L-1}(...a_2(W_2^T \begin{bmatrix} p_u^M \\ q_i^M \end{bmatrix} + b_2)...)) + b_L) \\
\hat{r}_{ui} &= \sigma(h^T \begin{bmatrix} \Phi^{GMF} \\ \Phi^{MLP} \end{bmatrix})
\end{aligned} \tag{3.26}$$

where p_u^G and q_i^G denote the user embedding and item embedding for GMF, respectively. Similar notations of p_u^M and q_i^M for MLP. The main advantage of this method relies on the integration of the linearity of MF and the non-linearity of DL for modeling user-item latent structures.

3.4 Recommender Systems Challenges

Despite the remarkable success of RSs in tackling the information overload issue, they still suffer from different limitations [BOHG13, CCW15]. This section investigates the most common problems and challenges encountered in deploying RSs: accuracy, sparsity, and cold-start.

3.4.1 Prediction accuracy

One of the essential requirements of RSs is to provide the user with a helpful and enjoyable, personalized experience. The quality of recommendations directly impacts the fidelity level of users to the RS. Suppose the RS does not suggest interesting products to users. In this case, it can be viewed as inutile concerning user experience, thus making it evident for them to turn to alternative solutions. Consequently, a RS needs to allow an appropriate level of prediction accuracy to prove its efficiency and utility.

Rating-based systems rely on the rating history of the items given by the users in the system. The scalar rating information frequently lacks a sufficiently semantic explanation to reflect the user's detailed preferences. Thus, accuracy appears as a significant challenge, especially for these methods since they use scalar rating information as to their unique source of user preference information, making it challenging to provide a user with high-quality personalized predictions [CCW15].

Recently, due to the emergence of advanced text analysis and opinion mining methods, many works have been made to exploit user reviews to improve RS. Such works incorporate the rich information embedded in reviews into the process of user modeling and recommendation generation. In particular, information captured from reviews helps to improve the prediction accuracy of RSs by providing them additional information about user preferences, which cannot be obtained from its overall ratings. The next Chapter of our thesis discusses researches that have utilized reviews to enhance RSs.

3.4.2 Sparsity and cold-start problems

Typically, there are a large number of missing ratings in the user-item interaction data, and the sparsity is frequently superior to 99% [SLH14]. This is due to the difficulty that users encounter when they want to express their interests as numerical ratings on products [LCC06], or because of the poor recommendation space's coverage [AJAP+19]. When rating-based algorithms work on sparse datasets, they fail to effectively derive relationships among users and items. Data sparsity may lead to another critical issue known as the cold-start problem. This issue occurs when novel users/items are added to the rating matrix. In such cases, most of the rating-based methods are not able to provide these users with recommendations nor to recommend these items since the system has not yet collected enough ratings about them [KAU16]. This problem has a significant negative influence on the effectiveness of the neighbor-based approaches [PPK05]. Due to the sparsity issue, likely, the similarities among users cannot be calculated, decreasing the effectiveness of rating-based systems. When the similarities are calculable, they may be unreliable since the information obtained is insufficient. One of the remedies for rating-based RSs to deal with the sparsity issue is to convert the high-dimensional and sparse rating matrix into a lower-dimensional, and denser pattern by utilizing DL technologies [BYBK19]. Another efficient solution relies on incorporating the rich information embedded in user reviews into the user modeling process of RSs (see next Chapter). The reviews can help overcome rating sparsity by furnishing complementary information concerning user interests. For instance, by combining the review information with user-provided numerical ratings, RSs may build an improved user profile. Also, when the user does not provide rating information (cold-start problem), the reviews can be exploited to deduce the ratings required by RSs.

3.5 Recommender Systems Evaluation

The evaluation of RSs is essential to get a clear conclusion concerning the quality of different recommendation methods. This section describes evaluation approaches, some evaluation metrics, and datasets used to investigate the performance of RSs.

3.5.1 Evaluation approaches

Evaluation represents an integral part of any system-building process for proving its efficiency for the interest tasks [RSDO20]. Different evaluation approaches have been used by the research community to evaluate the performance of RSs [RRS11]. These can be widely categorized into two main methods—online and offline. The first approach implies providing recommendations to the users and then querying them regarding the recommended items. The offline approach does not involve real users' interactions. It proceeds by simulating the online process where the system generates predictions or usage predictions, and the user corrects the predictions or uses the predictions (recommendations). This is generally achieved by recording historical user data and dividing it into two parts: training the system and testing the system's capacity to simulate how a user will evaluate an item or which suggested items a user will consume. There are several ways to select the test set (set of user-item interactions to be concealed). Typically, this selection should approximate as nearly as possible the data the designer hope the RS to face when deployed in the target application. The online approach is considered the best evaluation method due to its capacity to provide precise feedback on how pertinent the system is by implying real users [GGM+19]. Nevertheless, interactions with real users are mainly time-consuming; thus, many works have adopted an offline evaluation approach [SZL+19].

3.5.2 Evaluation metrics

The main goal of the majority of RSs is to perform the prediction task. These RSs may predict user preferences on items (unknown user-item interactions) or the usage probability (consuming a recommended item). A fundamental assumption in a RS is that the user will prefer a system that provides more accurate predictions. Thus, many researchers set out to find algorithms that give better predictions. This way, accuracy has become the most studied aspect in the RS literature. Table 3.2 presents some of the standard evaluation metrics used in offline settings to investigate the accuracy of RSs, their definitions, and their formulas. These can be widely categorized into three types of metrics [RRS15]: metrics to measure the accuracy of rating predictions, metrics to measure the accuracy of usage predictions, and metrics to measure the accuracy of items rakings.

In some RS deployments, the main task of the RS relies on predicting the rating that a user would rate a given item (e.g., 1-star through 5-stars). In this case, the RS designer whish to assess the accuracy of the system's predicted ratings. Mean Absolute Error (MAE), Mean Squared Error (MSE), and RMSE (Root Mean Squared Error) are the most used metrics to evaluate predicted ratings' accuracy.

In other RS deployments, the objective of the RS is not to predict user ratings on items but to suggest to the user items that it will probability consume. For instance, when a user adds an article to the queue in a given e-platform, the system suggests a set of articles that may be fascinating, given the added article. In a similar situation, the RS designer is interested in evaluating if the RS correctly predicts that the user will add these articles to the queue (use the items) rather than assessing if the system adequately deduces the ratings of these articles. In the context of usage prediction evaluation, researchers and professionals in RSs have proposed different evaluation metrics, which Precision and Recall represent the widely used ones.

In other cases, when RS is deployed in real-world applications, it can provide the target user with a list of suggestions that respect a particular browsing order. Unlike the previous cases (predicting ratings or selecting a set of recommended items), in this situation, the RS designer is focused on evaluating the order in which items are ordered in the suggested list, according to the user's preferences. This task is typically known as "Evaluating the ranking of items". There are various metrics for evaluating the accuracy of such a ranking, including R-score, Kendall and Spearman rank correlation, and Normalized Distance-based Performance Measure (NDPM) [HKTR04].

Table 3.2: Evaluation metrics for RSs.

Metrics	Definition	Formula	References
Mean Absolute Error	It measures the average of the absolute difference among the predicted ratings and true values.	$MAE = \frac{1}{\lvert T \rvert} \sum_{(u,i) \in T} \lvert \hat{r}_{u,i} - r_{u,i} \rvert.$ where $r_{u,i}$ refers to the real rating for user u over item i and $\hat{r}_{u,i}$ is the predicted rating by a CF system, $T = \{(u,i)\}$ denotes the set of user-item pairs for which the real ratings $r_{u,i}$ are known.	[HKTR04]
Mean Squared Error	It measures the average squared difference between the predicted values and the real values.	$MSE = \frac{1}{\lvert T \rvert} \sum_{(u,i) \in T} (\hat{r}_{u,i} - r_{u,i})^2.$	[A+16]
Root Mean Squared Error	It emphasizes the contributions of the absolute errors between the predictions and the real values.	$RMSE = \sqrt{\frac{1}{\lvert T \rvert} \sum_{(u,i) \in T} (\hat{r}_{u,i} - r_{u,i})^2}.$	[A+16]
Precision	It computes the rate of the provided recommendations that are pertinent.	$Precision = \frac{\lvert U_u \cap L_{rec} \rvert}{L_{rec}},$ where U_u represents the number of all items used by the user u and L_{rec} is the list of recommended items.	[A+16]
Recall	It computes the rate of recommendations that are provided.	$Recall = \frac{\lvert U_u \cap L_{rec} \rvert}{U_u}.$	[A+16]
R-Score	It measures the quality of recommendations based on their rank position.	$rank(L_{rec}) = \sum_{j=1}^{\lvert L_{rec} \rvert} \frac{max(r_{(i_j)} - md, 0)}{2^{\frac{j-1}{\alpha-1}}},$ where $r_{(i_j)}$ is the item i's rating in the rank j, md refers to the median rating and α is the value of half-life decay.	[SZL+19]
Others	–	–	[A+16, SZL+19, HKTR04, RSDO20]

3.5.3 Real-world datasets for Offline Evaluation

An offline evaluation is realized by utilizing a pre-collected dataset of users, items, and interactions. In some cases, the dataset may contain other information types like the time at which each user has rated the item, reviews in which the user textually justifies its scalar rating, etc. Utilizing such datasets, the researchers and professionals can try to simulate the behavior of users that interact with a RS.

Pre-collected datasets are advantageous because it does not imply interactions with real users. Such that, RS researchers can investigate various algorithms at a low cost. Besides, once those datasets are collected and recorded, they can be exploited as benchmarks for analyzing RSs across different settings. Many datasets are publicly available and are commonly utilized in the RS literature. Table 3.3 presents a variety of benchmark datasets.

Table 3.3: Historical Datasets for RSs

Dataset Name and Link	**Category**	**Brief Description**
Movielens (https://grouplens.org/datasets/movielens/)	Movies	Publicly-available rating datasets, which the GroupLens Research collected from the movielens website.
Netflix (https://paperswithcode.com/dataset/netflix-prize)	Movies and TV shows	The Netflix dataset was the subject of many empirical studies, thanks to the Netflix Prize competition. It comprises rating and review data collected from the Netflix movie website.
IMDB (https://datasets.imdbws.com/)	Movies	It is a Movie reviews data that contain different subsets of data collected from the IMDb movie review website.
Yahoo Movies (https://webscope.sandbox.yahoo.com/catalog.php?datatype=r)	Movies	This dataset contains a small sample of Yahoo. It includes the Movies community's preferences for various movies. It is composed of 211,197ratings expressed by 7,642 users on 11,915 movies.
Yahoo Music (https://webscope.sandbox.yahoo.com/catalog.php?datatype=r)	Music	This dataset includes the Music community's preferences for various musical artists. It comprises over ten million ratings of musical artists given by Yahoo.
Amazon (https://jmcauley.ucsd.edu/data/amazon/)	E-commerce	It consists of various subsets of data containing product reviews and metadata from Amazon. In total, this dataset includes 142.8 million reviews spanning May 1996–July 2014.
Yelp (https://www.yelp.com/dataset)	Food	It is collected from the popular local searche directory Yelp. Yelp dataset is composed of 6,685,900 interactions expressed by 1,637,138 users for 192,609 businesses in 10 metropolitan areas.
BeerAdvocate (https://data.world/socialmediadata/beeradvocate)	Beers	This dataset consists of beer reviews obtained from ratebeer.com. It includes around 1.5 M reviews collected from 10 years (until November 2011).
Jester (https://data.world/socialmediadata/beeradvocate)	Jokes	Jester dataset consists of 4.1 million continuous ratings of jokes from 73,496 users of the jester joke website.
GoodReads (https://sites.google.com/eng.ucsd.edu/ucsdbookgraph/reviews)	Books	It contains reviews from the Goodreads website and various attributes related to the items. This dataset contains around 225,000 interactions between 808,000 users and 1,561,000.
Others (https://cseweb.ucsd.edu/ jmcauley/datasets.html)	-	-

Chapter 4

Recent Advances in Recommender Systems: User Reviews Incorporation

Contents

4.1 User Reviews: Preference-Knowledge Extraction

The growth of electronic commerce has promoted users to write and share reviews expressing their opinion regarding items. Typically, these user reviews are in free text form, which expresses various dimensions or viewpoints of the experience that a user had for a given item [SDS$^+$19]. They thus constitute a valuable information source on user preferences and may be used to learn fine-grained profiles of users and improve personalized suggestions. The authors in [CCW15] identified different types of preference knowledge elements that can be obtained from review text and exploited by RSs. Among those review elements, review topics, review opinions (sentiments), and semantic information remain the most employed ones for augmenting the standard rating-based RSs. This section presents these different types of preference knowledge and summarizes the most used learning techniques for integrating them into the recommendation process.

4.1.1 Review topics

Review topics refer to a different item's aspects which a writer reviews in its review. Such aspects describe a topic of each target item's category, and they exist in each item belonging to this category. Each aspect comprises a set of words/terms (e.g., the following words: "week", "Monday"," "month", "last", "months" are related to the "Time" aspect). The words refer to the nouns that frequently occur in the reviews, and unlike the aspects, the words may not be present in all items in question. There are two typical methods for extracting the review topics. The first one is the frequency-based technique that selects frequently occurring nouns based on a set of seed words and then groups them into topics according to a predefined dictionary or manual technique [MLF13]. The second method is based on the use of a topic modeling technique like Latent Dirichlet Allocation (LDA) [BNJ03], Latent Semantic Analysis (LSA) [LMZ11], or Probabilistic Latent Semantic Analysis (PLSA) [Hof01], to automatically discover masked topics in reviews. However, LDA remains the most common technique applied to learn and represent the review topics. LDA refers to a generative statistical technique for topic modeling in the Natural Language Processing (NLP) context. The key assumption of the topic modeling is to describe a collection of documents based on a set of identified topics [WGS$^+$20].

Given some topics K and a corpus D consisting of a collection of documents M with the total size of the vocabulary V. In the LDA process, the word generation is made based on the conditional distributions $P(w_n = w|k_n = t)$, denoted by the matrix $\phi^{K\times V}$. In a similar way, the topic generation is made based on the conditional distributions $P(k_n = t|d_n = d)$, denoted by the matrix $\theta^{M\times K}$. The likelihood of the corpus D is then:

$$P(D, K|\phi, \theta) = \Pi_w \Pi_t \Pi_d \phi_{w|t}^{N_{w|t}} \theta_{t|d}^{N_{t|d}} \tag{4.1}$$

where $N_{w|t}$ refers to the the number of times that word w is related to (is generated by) topic t. $N_{t|d}$ refers to number of times that t occurs in the document d. Direchlet priors βn and αm are placed by the model, respectively, over ϕ, $P(phi|\beta n) = \Pi_t Dir(\phi_t, \beta n)$ and $P(\theta|\alpha m) = \Pi_d Dir(\theta_d, \alpha m)$. The likelihood of seeing the word w for the topic t ($P(w|t, D, K, \beta n)$) and the likelihood of seeing the topic t in the document d ($P(t|d, D, K, \alpha m)$) are respectivelly computed as follows:

$$\begin{aligned} P(w|t, D, K, \beta n) &= \tfrac{N_{w|t} + \beta n_w}{N_{.|t} + \beta} \\ P(t|d, D, K, \alpha m) &= \tfrac{N_{t|d} + \alpha m_t}{N_{.|d} + \alpha} \end{aligned} \tag{4.2}$$

Here $N_{.|t} = \sum_w N_{w|t}$, with $N_{.|t}$ denotes the total number of times of any words generated by the topic t. $N_{.|d} = \sum_t N_{t|d}$, with $N_{.|d}$ represents the total number of topics in document d.

In many review-based RSs, the LDA technique has proved its efficiency in representing users/items with topical information [ML13, BFZ14, TZLM16, CDZK18, CDH$^+$18]. In such scenarios, LDA was done on a corpus (collection of documents) of consumers' and/or items' reviews. Specifically, each user/item was represented as a probability distribution of topics captured from its past reviews (document).

Despite the popularity of the LDA approach, it has some significant shortcomings. This approach does not capture the order of the words in reviews and ignores the contextual semantics as it operates based on the bag-of-word process. Thus, a pair of semantically similar texts can be viewed as dissimilar if a few common words co-occur (are present) in these reviews.

4.1.2 Review opinion

User opinion or sentiment orientation (i.e., positive or negative) toward a target item can be inferred from a review text. The opinion can be expressed in relation to the object as a whole or any of its particular characteristics. Note that an item may be a product or service, and the features refer to the attributes and aspects of this item. The sentiment may either be negative or positive. The negative sentiment indicates that the user is not interested in the item, while the positive opinion suggests that the user likes the item.

SA (also called Opinion Mining (OM)) represents a task of NLP that aims to extract sentiments/opinions from texts [BKBH21]. It is mainly conducted on three levels: document level, sentence level, or aspect level. The document-level analysis attempts to classify the global polarity of the document into one of the predefined sentiment polarities (positive/negative). The sentence-level analysis attempts to identify the sentiment polarity (positive/negative) expressed in each sentence in the document. While the aspect-level analysis attempts to extract item features and then classify each opinion into positive or negative. This third level is more complex than the two first levels because it performs a fine-grained analysis. It remains the most used one for user and/or item modeling in RSs. This analysis level effectively captures detailed information about the characteristics of users and items using the feature opinions extracted from reviews. However, the first two levels may fail to identify the reviewer's likes or dislikes concerning the target item [SDS+19]. For instance, a negative review of an item does not necessarily indicate that the reviewer has negative opinions about all aspects of this item. Besides, a positive review does not automatically suggest that the reviewer has appreciated all the item's features.

Based on sentiment analysis, many works are proposed for improving personalized recommendations [PFT10, ZLZ+14, DQW+14, MCW17, MdGSL17, SZYM19, DSRO20b]. Such works benefit from applying sentiment analysis by learning fine-grained user and/or item profiles using either ML approaches or Lexicon-Based approaches [HL04, SS17, BKBH21]. ML represents the most widely-used technique. This approach uses linguistic features and machine learning algorithms for performing SA. In contrast, the lexicon-based method involves sentiment lexicons formed by a list of words and phrases commonly employed to indicate positive or negative sentiments.

4.1.3 Semantic information

Semantic information captured from the textual contents of the reviews may refer to different types of information ranging from review topics, feature opinions to simple facts concerning items. Semantic information represents the meaning of entities from the review text [CN21]. It allows capturing valuable preference information by considering the contexts of words, including word order and surrounding words of each term, which has been demonstrated to be essential for improving the representation and understanding in the NLP field [YHPC18]. Besides, the semantic information contained in terms/phrases of the textual reviews may be easily used to capture other types of preference knowledge, such as review topics and review sentiment [DSRO20a, LQP+19, DSRO20b].

With the recent success of DL methods in various NLP tasks [TSK+20], most efforts have focused on utilizing deep neural networks to derive latent semantic representations from textual information (reviews) for recommendation purposes [KPO+16, KPOY17, ZNY17, SHYL17]. CNN [ZW15, GBC16] represents one powerful category of deep neural network that applies convolution operation in place of general matrix multiplication in at least one of its layers [LWL+17]. This model consists of convolutional layers, pooling layers, and fully connected layers. These layers are stacked up on top of one another to form a deep neural network.

In a typical feature extraction task using a CNN in a RS, three neural layers are involved as follows:

Convolutional layer: The convolutional layer represents the most important module of the CNN model. It extracts features from the input text by applying convolution filters with mathematical operations.

Let a D textual document (review or collection of reviews) consisting of N words after padding. CNN firstly transforms the input text D into numerical vectors. Using word embedding techniques (such as Word2vec [MCCD13] and Glove [PSM14]) the document D is transformed to a $n \times d$ matrix denoted W_{Emb} , where each word is represented by a d dimensional embedding vector $w_i \in R^{1 \times d}$.

Given the embedding matrix W_{Emb} as an input of CNN, the convolution process can be expressed as follows:

$$O_j = a(W_{Emb} * C_j + b_j) \tag{4.3}$$

Where $*$ represents the convolution operation, $a(.)$ is the activation function [DQZ18, GWK+18, AZH+21]. $b_1, b_2, ..., b_k$ the biases values, and $C_1, C_2, ..., C_k$ the k convolution weight matrices of size $f \times d$.

Pooling Layer:

The pooling layer represents the second component of the CNN model. Its main objective relies on sub-sampling the outputs of the convolutional layer.

Based on the output O_j derived using j^{th} convolution layer. The max-pooling operation is applied to capture the most informative contextual features produced by the convolution layer. The max-pooling layer can be defined as follows:

$$MO_j = max(O_j) \tag{4.4}$$

Here MO_j contains the maximum elements of O_j. Finally, all captured features from the k convolutional layers are concatenated as the output of the max-pooling layer:

$$Out = (MO_j, i = 1, ..., k) \tag{4.5}$$

Fully Connected Layer:

At the final stage of CNN, the obtained features (out) are fed into a fully connected layer f with a weight matrix $W \in R^{k \times n}$ and a bias bs for deriving the final representation of the input text as:

$$SR = f(out * W + bs) \tag{4.6}$$

The main advantage of CNN relies on using word embedding representations and a local window to extract the contextual information, facilitating a better semantic understanding of review text and thus leading to considerable improvement compared to bag-of-words-based methods for RSs. CNN model has also demonstrated high performance in other feature-learning problems, including visual recognition [LDC20], speech recognition [PMDC19], text mining [SCS20], and various other tasks [YHPC18, GWK+18, KMLD21]. However, despite the outstanding achievements of the CNN model, it suffers from one critical issue when used in RSs: the features extracted by CNN are challenging to understand, which limits the interpretability of those systems to final users.

4.2 Recommender Systems Based on User Reviews

4.2.1 Related work

Recently, many RSs have exploited user review texts to derive the associated preference knowledge to enhance the prediction quality. Such systems have used different preference knowledge such as review topics, review sentiments, and semantic information. This section presents current reviews-based RSs by categorizing them according to the type of preference knowledge they extract from the review texts.

4.2.1.1 Approaches based on review topics

These techniques extract aspects from reviews and incorporate them into the rating prediction process to enhance the RS performance. For example, the method presented in [ML13] (called Hidden Factor and Topic (HFT)) fuses ratings with review topics. Firstly, it models reviews with the LDA-based topic model and ratings with standard MF. Then, a Softmax transformation function is used for incorporating the latent topics into the learning phase of the latent features model. Based on the trained model, the final rating scores are computed.

In the same way, as in [ML13], the model proposed in [TZLM16] utilizes MF for modeling rating scores and LDA for representing the text of reviews. However, in this model, items are described as topical distribution, and the topics in elevated rating reviews are repeated to augment their importance. Alike, users are represented in a similar topical space by their numerical ratings. The item and user representations are fused into a latent factorization model to perform the rating prediction task.

Based on the fact that the LDA technique cannot model the distribution of compounded topics, authors of [BFZ14] extended HFT [ML13] by proposing the TopicMF framework. TopicMF captures topics from user review text based on non-negative MF. Then it utilizes a MF technique for factorizing the rating matrix into latent user/item features. A transform action function joins the topic features with the matching latent user/item features for rating prediction.

More recently, the work [CDZK18] presented an Aspect-Aware Latent Factor Model (ALFM) that leverages an Aspect-aware Topic Model (ATM) for modeling aspect-level user/item representations as distributions of composite topics, each of which is represented by a set of words. In ALFM, the resulting representations from ATM are fused with latent rating factors to estimate the missing ratings based on the MF model.

Unlike ALFM, the A3NCF (An Adaptive Aspect Attention Model for Rating Prediction) model proposed in [CDH$^+$18] assumes that the latent topics directly refer to the aspects of items that users discuss in the reviews. Thus, it models feature vectors of users and items in different aspects as probability distributions of words representing the same topic. The obtained vectors are then fused with their corresponding identity-based embedding features in an attention neural network to learn their final representation by considering the user's attention weights concerning the different aspects of the target item. An attentive interaction between the user's and item's final representations is fed into fully connected layers to predict missing ratings.

Table 4.1: Related works on techniques based on review topics.

Citation	User/Item Profile	Recommending Method	Tested Datasets	Main Contribution	Accuracy Performance		
					Product Reviews	Achieved Accuracy	Accuracy of related Baselines
(HFT) [ML13]	Latent ratings merged with topic factors	Hidden Factors as Topics (HFT)	Amazon (movies, books, etc.), Beeradvocate and Ratebeer (wines, beers), Yelp (restaurants), etc.	Improves rating prediction accuracy (MAE), Tackle rating sparsity issue	26 Amazon product categories	1.329	LFM [KBV09]: 1.423
(RBLT) [TZLM16]	Latent topic opinions, latent rating factors	Matrix Factorization	Amazon (26 datasets [ML13])	Prediction accuracy improvement (MSE), Alleviate data sparsity problem	Video Games	1.462	LFM [KBV09]: 1.487
(TopicMF) [BFZ14]	Latent factors associated with topic factors	Topic Matrix factorization (TopicMF)	Amazon (arts, automotive, baby, beauty, etc.) [ML13])	Enhance prediction accuracy (MSE)	22 Amazon product categories	1.3468	PMF [MS08]: 1.5585 SVD++ [Kor08]: 1.4393
(ALFM) [CDZK18]	Latent topics, latent rating factors	Matrix Factorization	Amazon (26 datasets [ML13]), Yelp (businesses)	Improve prediction accuracy RMSE, Alleviate data sparsity problem, Interpretability in recommendations	Musical Instruments	0.893	BMF [KBV09]: 1.004
(A3NCF) [CDH+18]	ID embedded features with topic features	Attention-based Neural Collaborative Filtering	Amazon(Baby, Grocery, H & k, Garden, Sports), Yelp 2017	Augment the prediction accuracy RMSE	Sports	0.94	BMF [KBV09]: 1.087 RBLT [TZLM16]: 0.963 TransNet [CC17]: 0.983

4.2.1.2 Approaches based on review sentiment

Research works in this area use the user's expressed sentiment concerning the item itself or its different aspects in reviews to boost the rating prediction task. For instance, the authors in [PFT10] transform reviews into overall sentiment scores based on a machine learning method. To achieve this, reviews vectors fused with users' real ratings are exploited for training a Naive Bayes model on negative and positive classes. This learned model is then utilized for deducting ratings from novel reviews. The review-based ratings are used to construct a rating matrix integrated into the traditional neighbor-based CF techniques to predict ratings.

In the reference, [ZLZ+14] an Explicit Factor Model (EFM) was developed to transform user reviews into aspect-sentiment pairs. Based on the phrase-level sentiment analysis, it constructs two matrices: user-aspect attention and item-aspect quality, which are simultaneously decomposed with the rating matrix for performing rating prediction in a MF-based model.

The model proposed in [DQW+14] (called JMARS) utilizes the relationship between review aspects, opinions, and ratings to conduct CF. It exploits the Dirichlet-Multinomial technique for capturing the reviews' word distribution and a MF for generating the aspects ratings, which are fused with latent factors to compute the final rating scores.

On the other hand, the authors of [MCW17] have presented a user-preference-based CF that integrates aspect-level information to reflect user interests from reviews. Specifically, two metrics for aspect interests have been proposed, namely aspect need and aspect importance, respectively reflecting the differences of opinions to aspects and the aspect relationship to explicit rating. Based on these measures, the authors compute the similarity between users, which is then incorporated into memory-based CF to further recommendations.

In the work [MdGSL17], the authors developed multi-criteria user- and item-based CF techniques that integrate opinion information of review aspects. For user/item-based cases, authors present aspect-based item/user distances, which utilize the sentiment ratings deduced from review aspects. The similarity between users or items is then computed as the inverse of the proposed distances, and ratings are calculated using the standard CF model. The authors use the SABRE engine [CBdG+17] for performing the aspect extraction task in their work.

The model proposed in [SZYM19] represents a Sentiment-Based MF (SBMF) model that incorporates reviews' sentiments. To infer the review's overall sentiments scores, this model sums the sentiment score of each keyword in the target review based on the score obtained from a constructed sentiment dictionary. These sentiment scores are converted into real values and then fused with the users' explicit ratings into an extended probabilistic MF to perform rating prediction.

In a recent work [DSRO20b], the authors proposed a unified model to integrate aspects opinion information into CF. The model uses a multichannel CNN that involves word embedding and POS tag embedding layers for extracting review aspects. It regroups aspects using an LDA technique and then exploits a lexicon approach to build the aspects rating matrices. The aspect ratings are then weighted based on a tensor factorization method and integrated with a rating matrix into an LFM for predicting final ratings.

Table 4.2: Related works on techniques based on review sentiments.

Citation	User/Item Profile	Recommending Method	Tested Datasets	Main Contribution	Accuracy Performance		
					Product Reviews	Achieved Accuracy	Accuracy of related Baselines
[PFT10]	Ratings from opinion classification	Item-based CF	Flixster (movies)	Overcoming the cold-start issue (RMSE)	Movies	0.898	user-based CF: 0.897
(EFM) [ZLZ+14]	Ratings and aspect-sentiment scores	Factorization model	Yelp (businesses), Dianping (restaurants)	Improve prediction accuracy (RMSE)	Businesses	1.212	PMF [MS08]: 1.253 NMF [ZWFM06]: 1.248
(JMARS) [DQW+14]	Latent ratings and aspect-sentiment scores	Probabilistic matrix factorization	IMDB (movies)	Prediction accuracy increasing and address the cold start problem (MSE)	Movies	4.97	PMF [MS08]: 5.99
(UPCF) [MCW17]	Ratings and aspect-opinion ratings	User-based CF	Dianping (restaurants)	Accuracy increasing (RMSE), Deal with sparsity problem	Restaurants	0.7707	User-based CF: 0.7902 item- based CF: 0.8199
(Multi-U2U) [MdGSL17]	Opinion scores (Aspects)	Multi-criteria based user/item -based CF	Yelp (restaurants), TripAdvisor (hotels), Amazon (Video Games)	Increase prediction accuracy (MAE)	Video Games	0.6276	User-based CF: 0.9789 Item-based CF: 0.9679
(SBFM) [SZYM19]	Ratings with sentiment scores of reviews	Probabilistic matrix factorization	Amazon (Patio_lawn_ and_garden, Office products, Amazon instant video, Baby, Tools and home improvement, Beauty, Cellphones and accessories, Clothing and accessories)	Prediction accuracy improvement (Normalized RMSE)	Beauty	0.2898	MF [KBV09]: 0.3411 PMF [MS08] : 0.3338 HFT [ML13]: 0.3085
(AODR) [DSRO20b]	Ratings and aspect-sentiment scores	Tensor Factorization	Amazon (Musical Instruments, Automotive, Instant Video), Yelp (businesses)	Augment Rating prediction and address data sparseness (RMSE, MAE)	Instant Video	0.7990	MF [KBV09]: 0.9583 HFT [ML13]: 0.8172 RBLT [TZLM16]: 0.8061

4.2.1.3 Approaches based on semantic information

These works consider semantic information contained in reviews for regularizing rating-based approaches to improve prediction. The first work introducing CNN into recommendation is ConvMF [KPO$^+$16] which can capture local context information in a sliding window. ConvMF utilizes reviews text as complementary information. Firstly, this model uses convolutional operations and word embedding for capturing the item latent characteristics from their review texts. After that, the inferred latent features are integrated into a MF model to compute the user ratings on target items.

The researchers in [ZNY17] proposed a Deep Cooperative Neural Networks (DeepCoNN) model, which uses one of the most famous CNN text processing architectures to integrate reviews into the rating prediction task. In particular, the used architecture consists of two parallel convolutional neural networks (CNNs) and a word embedding method for capturing latent representations for all words of reviews associated with a target user and item. The model concatenates the user and item representations and then transmits them to a regression layer involving a FM technique to perform the prediction task.

Similar to DeepConn, the model developed in the work [CZLM18] (called NARRE) uses a CNN text processing network to derive latent embeddings of users and items from review texts. Unlike DeepConn, it scores reviews through an attention network to distingue their contribution to learning latent embeddings. NARRE uses the obtained attention scores with user latent rating factors and then incorporates them into an extended MF for rating prediction.

To ameliorate DeepCoNN, TransNet [CC17] extends it by introducing an extra layer for learning the latent representation of the target user-item review during the training phase, and then regularizing the output of this layer based on the learned representation. In the TransNet model, the extended neural architecture can mimic the target user-item review representation at test time, augmenting the precision of predictions.

Another improvement of DeepCoNN consists of PARL [WQLJ18], which plugged into DeepCoNN a plug-and-play model to enrich target user's preferences exploiting reviews written by similar users, especially when reviews of the target user are incomplete or sparse. The incorporated plug-and-play model extracts user-item pair-dependent features from the user's auxiliary reviews and then integrates the extracted features into CNN layers to vectorize them. PARL combines each extra vector with the corresponding user vectors in the DeepCoNN system for final ratings.

The work in reference [WQL$^+$19] fused the ratings and review information in a unified model called CARL. This model exploits CNNs and an attention mechanism to learn the relevant latent features by considering their related reviews. CARL constructs latent rating embeddings for users and items from the interaction matrix through a rating-based component. Finally, the learned content features and latent rating embeddings are integrated into a Factorization Machine (FM) for deriving final rating scores.

Very recently, a Hybrid neural recommendation model (called HRDR) has been proposed in [LWP$^+$20] to capture user and item embeddings from reviews and ratings. Firstly, the rating representations are obtained from rating data using a Multilayer Perceptron (MLP) network. After that, CNN with an attention mechanism is used to derive review-based representations where each review is associated with an informativeness score. At the final stage, a MF is used to compute user's ratings on items based on latent ratings, review features, and ID-embeddings.

The authors in [LQP$^+$19] proposed a capsule network-based model for recommendation called (CARP), which uses a Capsule Network to extract the semantic contextual information from reviews to reason the user rating behaviors. CARP firstly captures logic units (each formed by a user viewpoint and an item aspect) from the user and item reviews using a self-attention mechanism incorporated into a convolutional layer. Then, based on a novel Routing by Bi-Agreement process, it derives the sentiment representations in user-item level for prediction.

Table 4.3: Related works on techniques based on semantic information.

Citation	User/Item Profile	Recommending Method	Tested Datasets	Main Contribution	Accuracy Performance		
					Product Reviews	Achieved Accuracy	Accuracy of related Baselines
(ConvMF) [KPO$^+$16]	Latent ratings and item review based on semantic information	CNN with PMF	Amazon (Instant Video), MovieLens (movies)	Enhance the rating prediction accuracy (RMSE)	Instant Video	1.1337	PMF [MS08]: 1.4118 CTR [WB11]: 1.5496
(DeepCoNN) [ZNY17]	Semantic information from reviews	CNN with FM	Yelp (restaurants), Amazon (Musical instruments), Beer (beers)	Improve prediction accuracy (MSE), Alleviate the sparsity problem	Musical Instruments, restaurants and beers	0.994	MF [KBV09]: 1.292 PMF [MS08]: 1.256 CTR [WB11]: 1.112
(TransNet) [CC17]	Semantic features from reviews	Target network integrated into DeepConn model	Amazon (Electronincs, Clothing-Shoes-Jewelry, Movies and TV), Yelp17	Augment prediction accuracy (MSE)	Clothing-Shoes-Jewelry,	1.449	MF [KBV09]: 1.521 DeepCoNN [ZNY17]: 1.549
(PARL) [WQLJ18]	Semantic contextual features from reviews	CNN based plug-and-play model incorporated into DeepConn architecture	Amazon (Office Products, Digital Music, Tools Improvement and Video Games), Beer	Augment prediction accuracy (MSE), Tackle the textual sparsity issue in reviews	Beer	0.561	PMF [MS08]: 1.636 SVD++ [Kor08]: 0.726 RBLT [TZLM16]: 0.576 ConvMF [KPO$^+$16]: 0.853 DeepCoNN [ZNY17]: 0.617 TransNet [CC17]: 0.586
(NARRE) [CZLM18]	Latent factors from ratings, and Semantic features from reviews	CNN with MF	Amazon (Toys_and_Games, Kindle_Store, and Movies_and_TV), Yelp (businesses)	Increase prediction accuracy (RMSE), Interpretability in recommendations	Kindle Store	0.7783	PMF [MS08]: 0.9914 NMF [ZWFM06]: 0.9023 SVD++ [Kor08]: 0.7928 HFT [ML13]: 0.7917 DeepCoNN [ZNY17]: 0.7875
(CARL) [WQL$^+$19]	Latent feature ratings, Semantic factors from review words	CNN and FM	Amazon (Musical Instruments, Office Products, Digital Music, Video Games, and Tools Improvement), RateBeer (Beer), Yelp (Restaurants)	Augment rating prediction performance (MSE)	Musical Instruments	0.776	PMF [MS08]: 1.401 ConvMF [KPO$^+$16]: 0.991 DeepCoNN [ZNY17]: 0.814 RBLT [TZLM16]: 0.815
(CARP) [LQP$^+$19]	Semantic features (viewpoints and aspects	Neural recommender based on a capsule network	Amazon (Musical Instruments, Office Products, Digital Music, Video Games, and Tools Improvement), RateBeer (Beer), Yelp16-17	Improve prediction accuracy (MSE) and Interpretability in recommendations	Beer	0.556	PMF [MS08]: 1.641 D-Att [SHYL17]: 0.614 DeepCoNN [ZNY17]: 0.618 RBLT [TZLM16]: 0.576 TransNet [CC17]: 0.587
(HRDR) [LWP$^+$20]	Explicit features from ratings, semantic features from reviews, ID embeddings	CNN with MF	Yelp 2013 and Yelp 2014 (yelp.com), Amazon (Video games and Gourmet food)	Augment recommendation accuracy (RMSE)	Video games	1.011	PMF [MS08]: 1.139 HFT [ML13]: 1.073 CTR [WB11]: 1.071 JMARS [DQW$^+$14]: 1.064 ConvMF+[KPO$^+$16]: 1.073 DeepCoNN [ZNY17]: 1.063 NARRE [CZLM18]: 1.055

4.2.2 Practical implications of review incorporation

From Tables 4.1, 4.2, 4.3, and 4.4, we can see that all works on review-based recommendation algorithms have proven their advantages compared to the traditional rating-based approaches. This section discusses the practical benefits of these review-incorporated techniques on two main issues: rating sparsity and rating prediction improvement.

4.2.2.1 Rating Sparsity

As indicated in the previous chapter, the lack of pertinent data like sparsity considerably reduces the efficiency of the rating-based techniques. To tackle this problem, researchers have explored user reviews in different ways (see Tables 4.1, 4.2, 4.3, and 4.4):

The works proposed in References [ML13, DQW$^+$14, BFZ14, TZLM16] have demonstrated the capacity of their approaches to mitigating the rating sparsity issue. These works exploit review topics (aspects) for enriching the latent factor model. They extract aspects from review texts using topic models and learn latent features from ratings using MF methods. Then, the latent topics and latent factors are combined to boost prediction performance. For example, HFT [ML13] uses a defined transform function to learn the latent factors and latent topics together. JMARS model [DQW$^+$14] leverages a one-to-one matching among the latent factors and the learned latent aspects for determining the final ratings. The work [BFZ14] fuses aspects in reviews with latent factors in a user-item rating matrix by exploiting a transform function. The RBLT model [TZLM16] uses a linear combination between them for building the final users/items representations, which are then used in the rating prediction task. RBLT extracts reviews' topics and associates them with aspects, and then uses an extended latent factor model to enrich latent ratings with aspects.

The authors in [PFT10] show that user reviews can be converted into text-based ratings and then used to replace the explicit user ratings in the CF process. This approach first infers opinion ratings from reviews based on a machine learning model and then executes a neighbor-based CF method. Therefore, this work has proved its ability to mitigate the rating sparsity issue by inferring ratings from review texts.

On the other hand, the research [MCW17] leverages review text for capturing the weights preference which the target user assigns to different aspects. All the user's reviews are used to derive the aspect preferences, making easy the similarities' computation among each users' pairs, no importance how a number of items they frequently rate, that can mitigate the data sparseness issue.

Differently, the studies [KPO$^+$16, ZNY17, DSRO20b] have demonstrated the capacity of their approaches to alleviating the sparsity problem by using rich semantic features extracted from review words through CNNs. Specifically, these studies confirmed that CNN helps adjust latent ratings by effectively representing contextual features of user/item review texts when the rating data is sparse.

Table 4.4: Comparisons based on sparsity situations.

Approach	**Level of Sparsity**	**Improvement in Accuracy**
UPCF [MCW17]	Dianping (Data-5) with #Reviews of each user: 5–9	MSE of HFT [ML13] < MSE of User-based CF
HFT [ML13]	Amazon (movies) with #Reviews for each user/product: 1–10	MSE of LFM [Kor08] − MSE of HFT > 0
RBLT [TZLM16]	Amazon (26 datasets) with #Reviews for each user/item: 1–10	MSE of (LFM [Kor08]/HFT [ML13]) − MSE of RBLT > 0
JMARS [DQW$^+$14]	IMDB #training reviews for each user/movie: 1–100	MSE of HFT [ML13] − MSE of JMARS > 0
ALFM [CDZK18]	Amazon (24 item categories) #reviews for each user/item: 1–10	RMSE of (BMF [KBV09]/HFT [ML13]/RBLT [TZLM16]) − RMSE of ALFM > 0
ConvMF [KPO$^+$16]	MovieLens-1m: 7 sub-datasets of different densities (0.93%; 1.39%; 1.86%; 2.32%; 2.78%; 3.25%; 3.71%)	RMSE of ConvMF < RMSE of (PMF [MS08]/CTR [WB11])
DeepCoNN [ZNY17]	Three datasets: Yelp, Beer, and Amazon (Music Instruments) with #training reviews for each user/item: 1–5	MSE of MF [KBV09] − MSE of DeepConn > 0
AODR [DSRO20b]	Amazon (Musical Instruments, Automotive, Instant Video) and Yelp datasets with #reviews for each user/item: 1–10	RMSE of (BMF [KBV09]/HFT [ML13]/RBLT [TZLM16]) − RMSE of AODR >0 MAE of (BMF [KBV09]/HFT [ML13]/RBLT [TZLM16]) − MAE of AODR > 0

4.2.2.2 Rating Prediction Improvement

Many works have proposed incorporating user review texts to improve traditional rating-based techniques. These researches can be classified into two main categories. The first focuses on integrating preference knowledge into rating prediction methods as extra data (factor/variable). Specifically, it modifies the standard rating-based techniques to incorporate implicit scores inferred from review texts to adjust explicit ratings and get more reliable and fine-grained ratings. For instance, the authors of References [ML13, TZLM16, DQW+14, BFZ14, CDZK18, ZLZ+14] have presented different modified versions of the standard latent factor model, namely, HFT, RBLT, JMARS, TopicMF, ALFM, and EFM models for improving numerical ratings by aligning them with latent topics in reviews. The works in References [KPO+16, CZLM18, WQL+19, LWP+20, DSRO20b] have improved the real ratings in traditional latent factor models by fusing them with latent feature vectors inferred from review words through an integrated CNN architecture. On the other hand, in Reference [MCW17], the traditional user similarity in neighborhood-based CF recommenders has been improved by considering the user aspect preference vectors inferred from reviews. Moreover, in Reference [SZYM19], the standard PMF has been improved by adjusting its real ratings by the sentiment scores inferred from reviews. The classic neural CF in [CDH+18] has been improved by merging the rating-based representations with topic-based ones extracted from reviews.

The second category focuses on integrating preference knowledge into rating prediction methods as virtual data, replacing real ratings to generate predictions. Concretely, it replaces the explicit user ratings in standard rating-based approaches with implicit ones generated from textual reviews. For example, the text-based ratings inferred from reviews can replace explicit ratings in neighbor-based CF approaches [PFT10, MdGSL17]. The user and item latent embeddings obtained by CNNs from reviews can be used as features in LFM to conduct rating prediction [ZNY17, WQLJ18, CC17, LQP+19].

These existing review-incorporated works have proven their efficiency in exploiting user review texts (see summarizes in Tables 4.1, 4.2, 4.3, and 4.4). For instance, In References [MCW17, MdGSL17], the neighborhood-based CF technique based on inferred sentiment scores has been shown to provide results superior to the traditional memory-based CF approaches. On the other hand, the modified latent factor models that fuse real ratings with review-based ratings have proven to be more precise than the traditional models, which only leverage real ratings [ML13, SZYM19, KPO+16, CZLM18, WQL+19, LWP+20, TZLM16, DQW+14, BFZ14, CDZK18, ZLZ+14, DSRO20b]. This is due to the rich information of user interests and item characteristics in reviews, which could be practical complementary to numerical ratings.

Furthermore, particular works have compared different review-based CF methods. The neural network techniques [SZYM19, KPO+16, ZNY17, CZLM18, WQL+19, LWP+20, DSRO20b, WQLJ18, CC17, LQP+19] usually outperform methods that rely on CF with topic modeling [ML13, TZLM16, DQW+14, BFZ14, ZLZ+14] because of the robust representation capacity of neural network architectures that can capture rich semantic features from review texts for representing users and items. However, techniques relying on topic modeling lose the deep textual characteristics through this coarse-grained text mining method.

On the other hand, we realize that the techniques [CZLM18, WQL+19, LWP+20, DSRO20b] leveraging attention networks usually outperform strategies without attention [ZNY17, KPO+16]. This is due to the attention mechanism usage, which allows capturing the more significant features in reviews and consequently provides a way for deriving user and item representations more precisely.

4.3 Recommender Systems Based on Arabic-language Reviews

4.3.1 Existing works

Various studies have proved the effectiveness of incorporating reviews into CF RSs. All these works have validated their systems using English review data. However, very few works were done to demonstrate the benefit of review incorporation in the Arabic context (using Arabic-language reviews). The published researches that we found are [ZAS+17, HASA20]. In these researches, the authors developed RSs exploiting Arabic reviews to help users decide on various items. Rather than using only the user-item rating matrix in their rating prediction process, they add the polarity of reviews. In [ZAS+17], the authors investigated their RS on three datasets; the first one contains 100 reviews in Algerian dialect, the second one accommodates 1000 reviews in Arabic, and the third one regroups 2000 reviews in French and English. They adopt a Semi-Supervised Support Vector Machine (S3VM) algorithm to derive the sentiment scores from reviews. Then, the polarity scores are used as votes in the user-based CF RS for rating prediction. On the other hand, the research [HASA20] focuses mainly on Arabic reviews. It exploits an Opinion Corpus for Arabic (OCA) dataset [RSMVULPO11], containing 500 Arabic reviews about different movies. The authors extract opinion polarity scores from the reviews using the Support Vector Machine (SVM) classifier. The derived sentiment scores are then combined with scalar ratings into an SVD-based MF technique to predict unknown ratings. Experimental studies in both works have shown that their models can efficiently predict items' ratings by considering the sentiment of their corresponding reviews (see Table 4.5).

Table 4.5: Related works on techniques based on Arabic reviews.

Citation	User/Item Profile	Recommending Method	Tested Datasets	Main Contribution	Accuracy Performance		
					Product Reviews	Achieved Accuracy	Accuracy of related Baselines
S3VM-based RS [ZAS+17]	Reviews polarities	User-based CF	English (Restaurant_TijuanaRestaurant), French (High-Tech), Arabic and dialects (JumiaMarket)	Accurate precision in the recommendation task	Jumia reviews	MAE: 0.60 Precision: 0.90 Recall: 1.0	–
SVD-based RS [HASA20]	user star ratings and reviews polarities	SVD-based MF technique	Opinion Corpus for Arabic (OCA) [RSMVULPO11]	Improve prediction accuracy (MAE)	Movie reviews	0.423	item-based CF with Mean Average Coefficient (MAC): 0450

4.3.2 Discussion and limitations

Table 4.5 shows that the available works on review-based CF algorithms have proven their advantages compared to the traditional rating-based recommending approaches. Table 4.6 provides details about the datasets used in these works. Nevertheless, from this reviewed literature, we can conclude some critical conclusions:

• Arabic RSs have received limited attention in the research literature.

• Their reported experiments have been conducted on inaccessible datasets and/or datasets originally devoted to SA purposes.

• Their reported experiments have been conducted on small product datasets of less than a thousand reviews, and none of them has focused on large Arabic datasets.

• The authors used elementary text analysis techniques to extract (useful features) information from reviews. Such methods cannot capture the context of words; thus, they cause semantic information loss.

• These works use only traditional RSs contrarily to modern and performant systems adopted in studies devoted to English content.

• None of these works has used several Arabic reviews datasets to evaluate its proposed RS in the Arabic context properly.

However, these works are thus not mature yet compared to English RSs studies. They do not provide comprehensive and complete ideas about the rating prediction application in the Arabic context. Aiming to shed light on this situation, in this work, we report an extensive investigation and propose novel tools and solutions to deal with the limitations of the above studies.

Table 4.6: Details about the used datasets

Dataset Name and Link	Category	Brief Description
Arabic and dialects (JumiaMarket) jumia reviews (https://www.jumia.dz/)	Clothes	It is a non-accessible reviews dataset. It is collected from dz.jumia.com "JumiaMarket", which is an Algerian-based online shopping websites. This dataset consists of 10 users, 5 oriental clothing for women and 50 evaluations.
Opinion Corpus for Arabic (OCA) (ttp://sinai.ujaen.es/wiki/index.php/OCA_Corpus_(English_version))	Movies	It is a publically-available arabic corpus for the Sentiment Analysis (SA) task. The corpus contains 500 movie reviews collected from different web pages and blogs in Arabic, 250 of them considered as positive and the other 250 as negative opinions.

Part III

Our proposal

Chapter 5

Exploring Advanced Recommender Systems in the Arabic Context

Contents

5.1 Overview

After reviewing the RSs literature, we found that few RSs have exploited the Arabic content. Specifically, the reader may refer to [ZAS^{+}17, HASA20] for early and only researches on RSs devoted to using Arabic-language reviews. Although these works were the precursors in the Arabic RS field, they suffer from different limitations discussed in the next chapter. The main ones are:

- In these studies, methods and recent advances in the RS field are not empirically used or compared.

- The datasets used in the experimentations of these works are inaccessible and/or small-in-size and/or designed primarily for research on Arabic SA.

- None of these studies has reported if there is or not a necessity to apply a special scheme for preparing the Arabic reviews before their usage in the RSs.

Because of all those shortcomings as well as the lack of works that apply the recent recommendation approaches in the Arabic context, we decided to conduct this work to give well-argued answers to the following three stated research questions, namely:

• Is it possible to apply recent RSs in the Arabic context?

• If so, does the Arabic content need a particular preprocess step to incorporate it in these RSs?

• If so, does the application of these RSs to Arabic content provide good results like when applying them to the English language content?

In what follows, we present the followed evaluation methodology for exploring RSs while exploiting Arabic content. It consists of running experiments with several state-of-the-art review-based recommender engines originally devoted to English content to evaluate and compare their performance when varying content language from English to Arabic. In this sense, we choose to deal with RSs in both contexts, Arabic and English, by exploiting textual reviews written in these two different languages. Our methodology can be summarized as follows: first, we prepare different English reviews datasets and generate their equivalent Arabic versions. Second, we preprocessed the constructed datasets. Third, we applied various recommending paradigms for rating prediction in both contexts (English and Arabic). Finally, we measured and compared their accuracy. The details of each phase are explained in the following sub-sections. The overall process is illustrated in Figure 5.1.

As a result of this study, we first build four publicly available large-scale Arabic datasets for recommendation purposes. Second, we develop a new relevant scheme for preprocessing the adopted Arabic and English reviews datasets. Third, we apply the rating prediction task in two different contexts (English and Arabic) by first implementing five widely used RSs in the literature and then applying them to the same content written in different languages, namely English and Arabic. Finally, we provide well-argued conclusions about the usage of modern RSs in the Arabic context.

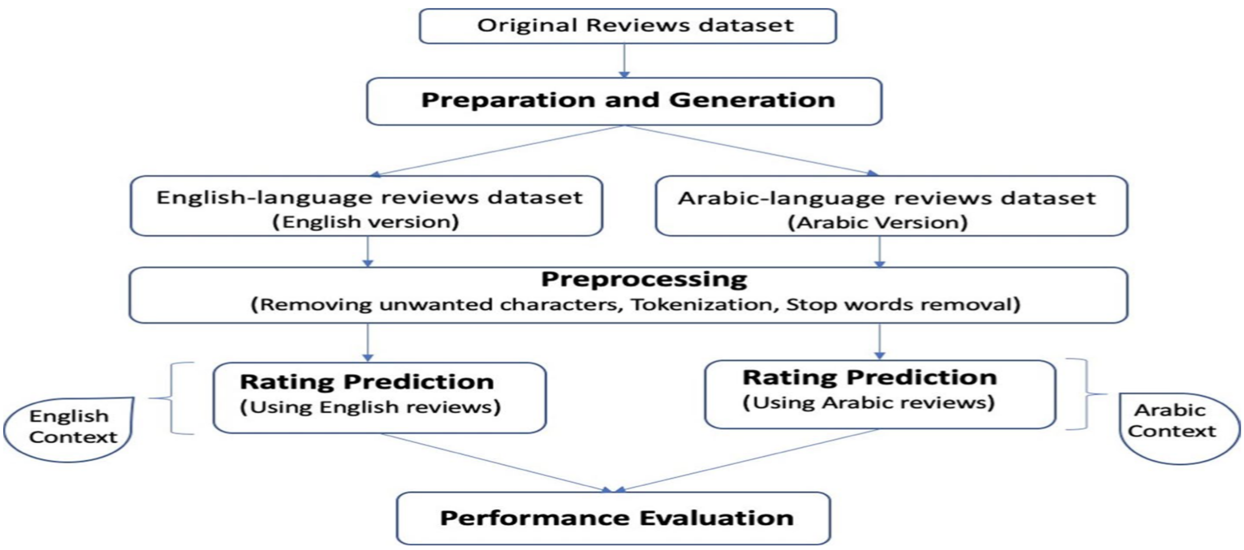


Figure 5.1: Steps followed in our methodology.

5.2 Data Preparation and Generation

Being aware of the lack of resources related to the Arabic language in the RSs field, we decided to build four publicly available Arabic datasets for RSs. The construction of each of these datasets was performed by preparing and translating English-language reviews obtained from a specific publicly available Amazon database [1] (Figure 5.2). This is the most extensive database used for RSs evaluation. It is collected by McAuley [ML13] and comprised of reviews and metadata from different varieties of Amazon products. It includes 142.8 million reviews spanning May 1996 - July 2014. In this thesis, we choose the four following English datasets, namely Musical Instruments (MI), Patio Lawn and Garden (PLG), Automotive (Auto), and Instant Video (IV).

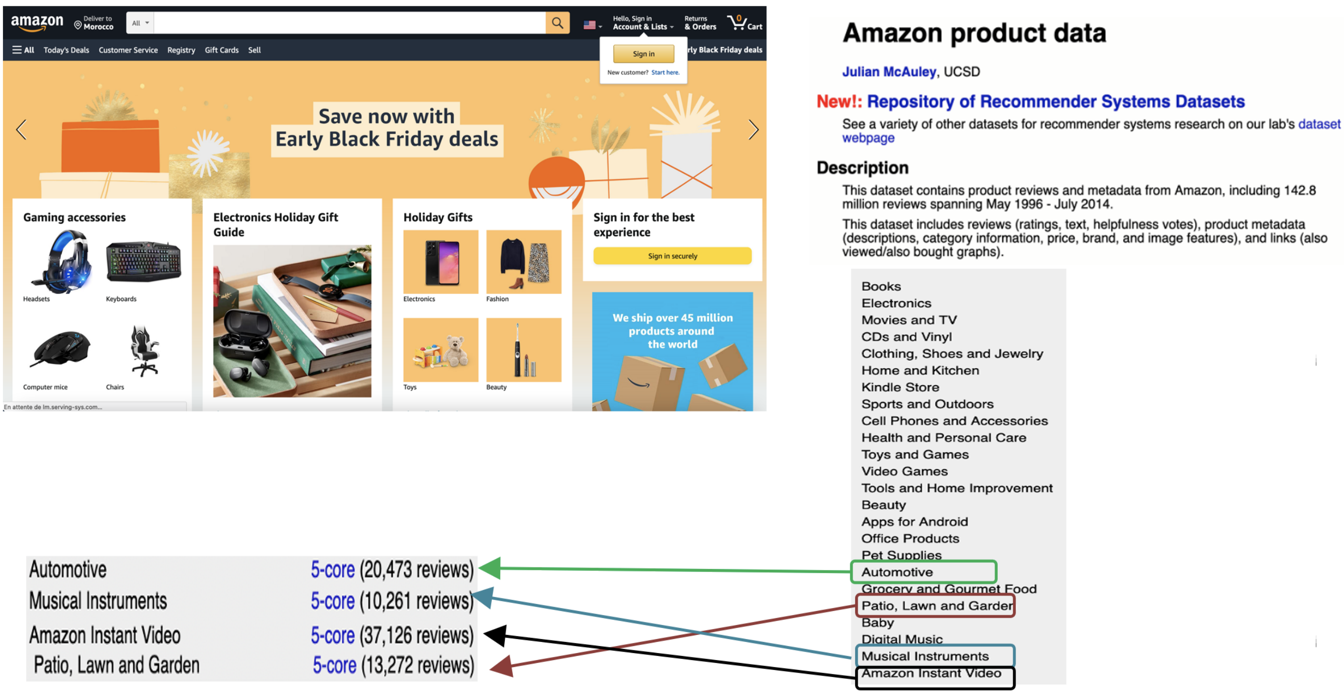


Figure 5.2: Amazon review data.

The information about these datasets are as follows:

- MI: The dataset contains 10,254 reviews that 1,429 users submitted for 900

[1] //jmcauley.ucsd.edu/data/amazon/

different musical instruments.

• PLG: The dataset contains 13,258 reviews that 1,686 users submitted for 962 different patios, lawn, and garden tools.

• Auto: The dataset contains 20,467 reviews that 2,928 users submitted for 1,835 different automotive accessories.

• IV: The dataset contains 37,125 reviews that 5,130 users submitted for 1,685 different movies and TV shows.

In fact, each review in these datasets consists of the following nine fields (See Figure 5.3), namely:

- reviewerID: ID (Identifier) of the reviewer in the Amazon platform.

- asin: ID of the item in Amazon websites.

- reviewerName: represents the name of the reviewer.

- helpful: helpfulness score of the review.

- reviewText: contains the review's text.

- overall: the rating score of the item.

- summary: contains the review's summary.

- unixReviewTime: the review's time (Unix [2] time).

- reviewTime: the review's time and date (raw).

[2]https://en.wikipedia.org/wiki/Unix

```
{
  "reviewerID": "A2SUAM1J3GNN3B",
  "asin": "0000013714",
  "reviewerName": "J. McDonald",
  "helpful": [2, 3],
  "reviewText": "I bought this for my husband who plays the piano.
He is having a wonderful time playing these old hymns.  The music  is
at times hard to read because we think the book was published for
singing from more than playing from.  Great purchase though!",
  "overall": 5.0,
  "summary": "Heavenly Highway Hymns",
  "unixReviewTime": 1252800000,
  "reviewTime": "09 13, 2009"
}
```


Figure 5.3: Sample Amazon review.

To prepare the datasets used in this study, we implemented a special parser in Python, allowing us to extract reviews from the original English datasets (JavaScript Object Notation (JSON) files) by retaining only the specific fields that we require. In fact, for each review, only the reviewerID, asin, reviewText, and overall were kept. The remaining fields were ignored since they are needless for this task. Thus, each record in our datasets contains one review, including the information associated with the targeted four fields.

However, online reviews are usually short informal texts generated by non-experts. They are characterized by using everyday language, grammar and misspelling errors, non-standard vocabulary such as replicated characters, non-formal abbreviations, etc. Thus, the reviews cleaning stage is essential to ensure the quality of their content. Its main goal is to clean reviews text from spelling errors and slang words to help the downstream process easily understand and resolve their meanings. To attain it, we proceed as follows: first, all the text data are converted into lower case letters to ensure that the texts are in a uniform format. The application of this task to our texts makes sure that "The" and "the" or "Do" and "do" are treated as the same. Then, we adopted regular expressions to eliminate extra characters in any sequence that repeated more than twice. After that, we addressed the issue of the misspelled terms in reviews based on a Spelling corrector in Python called Autocorrect [3]. This tool corrects all the detected misspelled words and replaces them with the correct terms. At the final stage, we replace the contractions marked by clitic apostrophes with their extended forms. This is

[3] https://libraries.io/pypi/autocorrect

achieved by transforming the Wikipedia English contraction-to-expansion list [4] into a python dictionary and then exploiting a regular expression for expanding all the existent contractions.

We used a free online Machine Translation (MT) service to generate the Arabic versions of the prepared English datasets. We decided to use the Google Translate MT tool due to its high efficiency in the translation task. Thus, we implemented a python script using the Python googletrans library [5] that allows interaction with Google Translate API [6]. For each review in the used datasets, the content of its corresponding reviewText field was sent to the API to get its translation into the Arabic language. Figure 5.4 describes the overall process adopted to obtain our datasets.

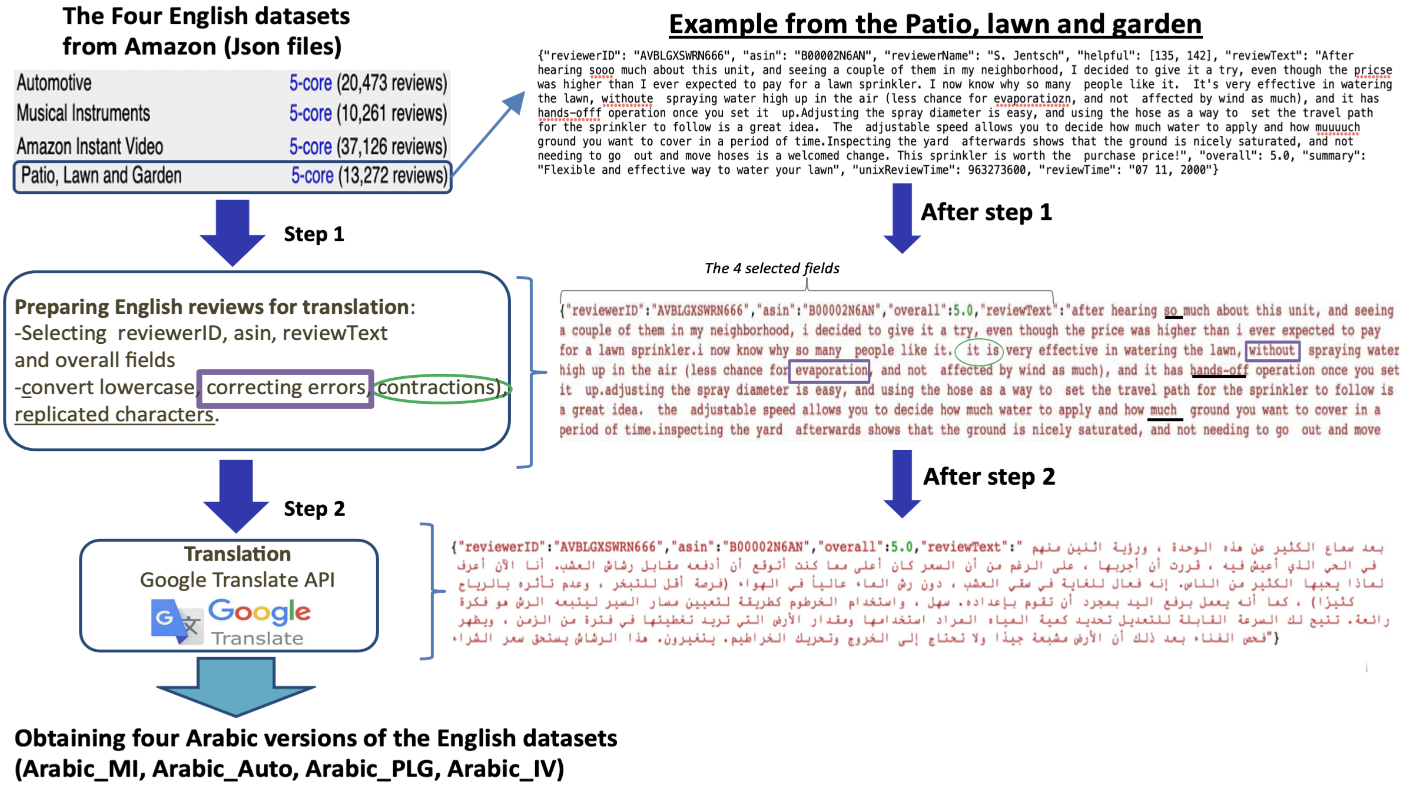


Figure 5.4: The overall process adopted to obtain our datasets.

[4]https://en.wikipedia.org/wiki/Wikipedia:List of English contractions
[5]https://pypi.org/project/googletrans/
[6]https://translate.google.com

The four constructed Arabic-language reviews datasets are Arabic_MI, Arabic_ PLG, Arabic_Auto, and Arabic_IV. Each dataset contains the users' reviews towards a specific category of items. Each review consists of a reviewerID, ID's item, a score of the evaluated item (from 1 to 5), a review text in the Arabic language. Figure 5.5 shows an example of the constructed reviews from the prepared English original equivalent ones. Table 5.1 contains the essential properties of the created datasets.

{"reviewerID":"A195EZSQDW3E21","asin":"1384719342","overall":5.0,"reviewText":"the primary job of this device is to block the breath that would otherwise produce a popping sound, while allowing your voice to pass through with no noticeable reduction of volume or high frequencies. the double cloth filter blocks the pops and lets the voice through with no coloration. the metal clamp mount attaches to the mike stand secure enough to keep it attached. the goose neck needs a little coaxing to stay where you put it."}

{"reviewerID":"A195EZSQDW3E21","asin":"1384719342","overall":5.0,"reviewText":" تتمثل المهمة الأساسية لهذا الجهاز في منع التنفس
الذي قد ينتج عنه صوت فرقعة ، مع السماح لصوتك بالمرور دون انخفاض ملحوظ في الحجم أو الترددات العالية. يحجب مرشح القماش المزدوج
الملوثات العضوية الثابتة ويسمح للصوت بالمرور بدون تلوين. يتم توصيل المشبك المعدني بحامل الميكروفون بشكل آمن بما يكفي لإبقائه
متصلًا. تحتاج رقبة الإوزة إلى القليل من الإقناع للبقاء في المكان الذي تضعه فيه"}

(a) An Arabic translated review (from the first English-reviews dataset)

{"reviewerID":"AVBLGXSWRN666","asin":"B00002N6AN","overall":5.0,"reviewText":"after hearing so much about this unit, and seeing a couple of them in my neighborhood, i decided to give it a try, even though the price was higher than i ever expected to pay for a lawn sprinkler.i now know why so many people like it. it is very effective in watering the lawn, without spraying water high up in the air (less chance for evaporation, and not affected by wind as much), and it has hands-off operation once you set it up.adjusting the spray diameter is easy, and using the hose as a way to set the travel path for the sprinkler to follow is a great idea. the adjustable speed allows you to decide how much water to apply and how much ground you want to cover in a period of time.inspecting the yard afterwards shows that the ground is nicely saturated, and not needing to go out and move hoses is a welcomed change. this sprinkler is worth the purchase price!"}

{"reviewerID":"AVBLGXSWRN666","asin":"B00002N6AN","overall":5.0,"reviewText":" بعد سماع الكثير عن هذه الوحدة ، ورؤية اثنين منهم
في الحي الذي أعيش فيه ، قررت أن أجربها ، على الرغم من أن السعر كان أعلى مما كنت أتوقع أن أدفعه مقابل رشاش العشب. أنا الآن أعرف
لماذا يحبها الكثير من الناس. إنه فعال للغاية في سقي العشب ، دون رش الماء عالياً في الهواء (فرصة أقل للتبخر ، وعدم تأثره بالرياح
كثيرًا) ، كما أنه يعمل برفع اليد بمجرد أن تقوم بإعداده. سهل ، واستخدام الخرطوم كطريقة لتعيين مسار السير ليتبعه الرش هو فكرة
رائعة. تتيح لك السرعة القابلة للتعديل تحديد كمية المياه المراد استخدامها ومقدار الأرض التي تريد تغطيتها في فترة من الزمن ، ويظهر
فحص الفناء بعد ذلك أن الأرض مشبعة جيدًا ولا تحتاج إلى الخروج وتحريك الخراطيم. يتغيرون. هذا الرشاش يستحق سعر الشراء"}

(b) An Arabic translated review (from the second English-reviews dataset)

{"reviewerID":"A20S66SKYXULG2","asin":"B00002243X","overall":4.0,"reviewText":"these long cables work fine for my truck, but the quality seems a little on the shabby side. for the money i was not expecting 200 dollar snap-on jumper cables but these seem more like what you would see at a chinese knock off shop like harbor freight for 30 bucks."}

{"reviewerID":"A20S66SKYXULG2","asin":"B00002243X","overall":4.0,"reviewText":" تعمل هذه الكابلات الطويلة بشكل جيد لشاحنتي ، لكن
الجودة تبدو ضعيفة بعض الشيء. بالنسبة للمال ، لم أكن أتوقع كبلات توصيل إضافية بقيمة 200 دولار ، لكن هذه تبدو أشبه بما تراه في متجر
صيني مثل شحن المرفأ مقابل 30 دولارًا"}

(c) An Arabic translated review (from the third English-reviews dataset)

{"reviewerID":"A2MOIORZE53NL8","asin":"B000H4YNM0","overall":5.0,"reviewText":"each episode gives me more entertainment than anything else on the tube. though i may not want to have these characters as real friends - the women are hot though - the plots and characters make for great viewing.do not see myself abandoning this series.i give 5 stars to every episode and season of always sunny in philadelphia!thanks amazon!"}

{"reviewerID":"A2MOIORZE53NL8","asin":"B000H4YNM0","overall":5.0,"reviewText":" تمنحني كل حلقة مزيدًا من الترفيه أكثر من أي شيء
آخر على القناة. على الرغم من أنني قد لا أرغب في الحصول على هذه الشخصيات كأصدقاء حقيقيين - فالنساء مثيرات على الرغم من - المؤامرات
والشخصيات تجعل المشاهدة رائعة. لا أرى نفسي أتخلى عن هذه السلسلة. أعطي 5 نجوم لكل حلقة وموسم مشمس دائمًا في فيلادلفيا! شكرا أمازون"}

(d) An Arabic translated review (from the fourth English-reviews dataset)

Figure 5.5: An example of the Arabic reviews obtained from the prepared English original ones.

Table 5.1: The essential properties of the created Arabic datasets.

Datasets	**# users**	**# items**	**# reviews**
Arabic_MI	1,429	900	10,254
Arabic_PLG	1,686	962	13,258
Arabic_Auto	2,928	1,835	20,467
Arabic_IV	5,130	1,685	37,125

5.2.1 Preprocessing data

Due to the unstructured text of reviews, it won't be easy to analyze them directly. Thus, it is recommended to preprocess all the reviews before integrating the reviews into the recommendation process. To do this, we created our own text preprocessing scheme, which implies different stages, including unwanted characters removal, Tokenization, removing standard stop words, and deleting domain-specific stop words. The details related to each step are discussed below. Figure 5.6 illustrates the adopted overall preprocessing flow. Figure 5.7 shows an example of reviews after applying the preprocessing steps.

Removing unwanted characters: The first stage of the preprocessing module consists of removing unwanted characters. It removes all marks, extra whitespaces, and any other non-alphabetic characters detected in the reviews. This phase is very significant as such instances do not add any value to the content of reviews. Hence, removing these characters will help in manipulating and processing text information. Our preprocessing scheme uses Python's re module [7] for removing all the aforementioned unwanted characters from the text of the reviews.

Tokenization: As another step of preprocessing, there is Tokenization. This step aims to split the text into a set of tokens (words) based on the punctuation or whitespaces characters. In our text preprocess scheme, this task was applied for each review text to divide it into multiple words utilizing whitespaces characters. There are many libraries to perform tokenization like NLTK [8], SpaCy [9],

[7]https://docs.python.org/3/library/re.html
[8]https://pypi.org/project/nltk/
[9]https://pypi.org/project/spacy/

and TextBlob [10]. The NLTK was utilized in this work. The obtained tokens are exploited for further preprocessing stages.

Standard Stop words removal: Stop words like prepositions, pronouns, articles, and conjunctions are employed frequently in reviews. Such words carry less meaning than other keywords, and thus they are not significant for feature extraction tasks for RS. Therefore, a stop words removal step was performed in our study before beginning the recommendation process. There is no unique universal stopwords list in the literature. For this, we manually created two lists of stop words in our work. The first one is devoted to the English language, and the second one concerns the Arabic language. We also add the words that have the document frequency [Aiz03] higher than 0.5 by considering them as stop words in these lists. This step compares each target token with its corresponding language stop word list. If it appears in the list, then it is eliminated. For instance, the words such as (have, he, in, is, it, its) are suppressed from the English and Arabic reviews.

Domain-specific Stop words removal: While we removed the frequent stop words, it is crucial in text mining to identify unusual words with little discriminant value within a specific field or context. These words are called Domain-specific stop words because they differ from one domain to another. The metric most widely utilized to perform the Domain-specific Stop words removal is" Term Frequency Inverse Document Frequency" (TF-IDF) [Aiz03]. It computes the relevance of words in a document based on their occurrence frequency on several documents. Using this metric in our scheme, we may pick out the most pertinent words allowing us to represent better the reviews related to the target dataset. Thus, we decided to apply the TF-IDF technique on all datasets by retaining the top 20,000 distinct words (with high TF-IDF scores) as a vocabulary for each small dataset (Number of reviews$<$15,000) and the top 50,000 distinct words for large datasets (Number of reviews$>$15,000). All words out of the vocabulary are removed and considered as domain-specific stop words. We further filter out the reviews containing empty text after applying this step.

[10]https://pypi.org/project/textblob/

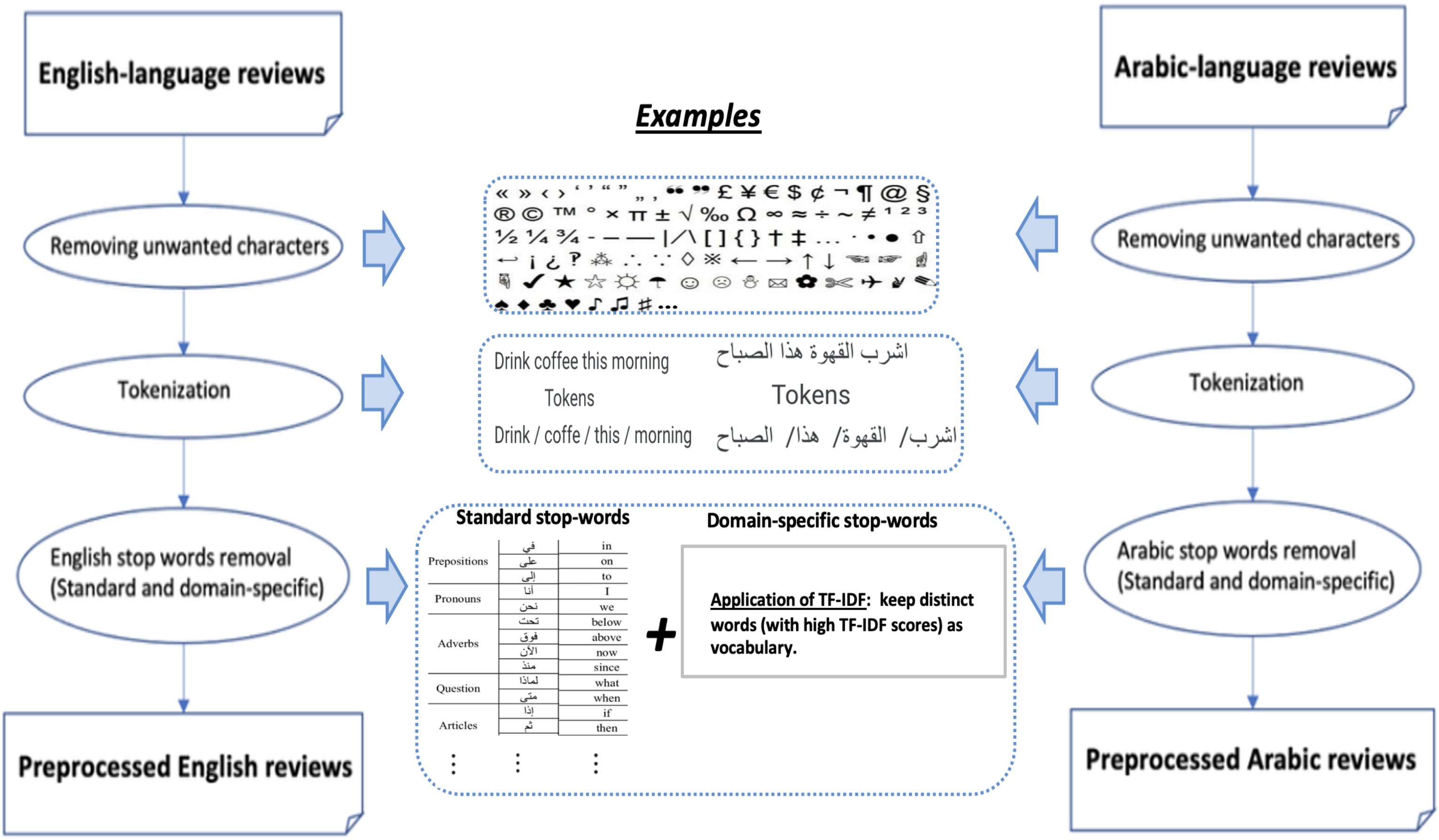


Figure 5.6: Preprocessing flows for the used English and Arabic reviews.

Example of a preprocessed english review

{"reviewerID":"AVBLGXSWRN666","asin":"B00002N6AN","overall":5.0,"reviewText":"hearing unit couple neighborhood price higher expected pay lawn sprinkler people like watering lawn spraying water air less chance evaporation affected wind hands-off once spray diameter hose travel path sprinkler adjustable speed water ground cover period inspecting yard ground saturated hoses welcomed sprinkler worth purchase price"}

Example of a preprocessed arabic review

{"reviewerID":"AVBLGXSWRN666","asin":"B00002N6AN","overall":5.0,"reviewText":" سماع الوحدة رؤية الحي أعيش قررت أجربها الرغم السعر أدفعه رشاش العشب فعال سقي العشب رش الماءالهواء فرصة للتبخر تأثره الرياح برفع اليد بمجرد تقوم بإعداده استخدام الخرطوم كطريقة لتعيين مسار السير الرش فكرة السرعة القابلة كمية المياه المراد ومقدار الأرض تغطيتها فترة الزمن فحص الفناء الأرض مشبعة الخروج تحريك الخراطيم الرشاش يستحق سعر الشراء"}

Figure 5.7: Examples of reviews after applying the preprocessing tasks.

5.2.2 Characteristics of the datasets after preprocessing

Initially, it should be noted that all datasets (English and Arabic) contained a certain noise level and were not totally cleaned up. After preprocessing the textual reviews for all datasets (English and Arabic), we ended up with new preprocessed versions of those same datasets. Details about the English datasets are summarized in Table 5.2. Details about the Arabic datasets are summarized in Table 5.3

Table 5.2: Statistical details of the English datasets after preprocessing.

Datasets	# users	# items	# reviews
MI	1,429	900	10,254
PLG	1,686	962	13,258
Auto	2,928	1,835	20,467
IV	5,130	1,685	37,125

Table 5.3: Statistical details of the Arabic datasets after preprocessing.

Datasets	# users	# items	# reviews
Arabic_MI	1,429	900	10,254
Arabic_PLG	1,686	962	13,258
Arabic_Auto	2,928	1,835	20,409
Arabic_IV	5,130	1,685	37,123

5.3 Adopted Recommender Systems

In our experimentation, the rating prediction task was investigated using five RSs: ALFM, A3NCF, PARL, CARL, and CARP. These RSs are presented below. We refer the readers to read the original papers [CDZK18, CDH$^+$18, WQLJ18, WQL$^+$19, LQP$^+$19] in order to have more details about these RSs. Figures 5.8, 5.9, 5.10, and 5.11 provide the architectural structures of the adopted models, respectively.

ALFM: is the state-of-the-art recommender model that adopts the Latent Dirichlet Allocation (LDA) paradigm with the Probabilistic MF for rating prediction. This model firstly runs an LDA-based algorithm on reviews' texts to model user's preferences and item's properties in different aspects, thus capturing the importance of aspects for the user and item. More specifically, it exploits an Aspect aware Topic Model (ATM) for modeling aspect importance for target user/item as a probability distribution of composite topics, each of which is represented by a set of words from reviews. Then, the output from ATM is combined with ALFM, which associates latent factors with different aspects exploiting the MF approach, such that the model can predict aspect ratings. Finally, the overall rating is obtained by a linear combination of the aspect ratings, which are weighted by the importance of corresponding aspects.

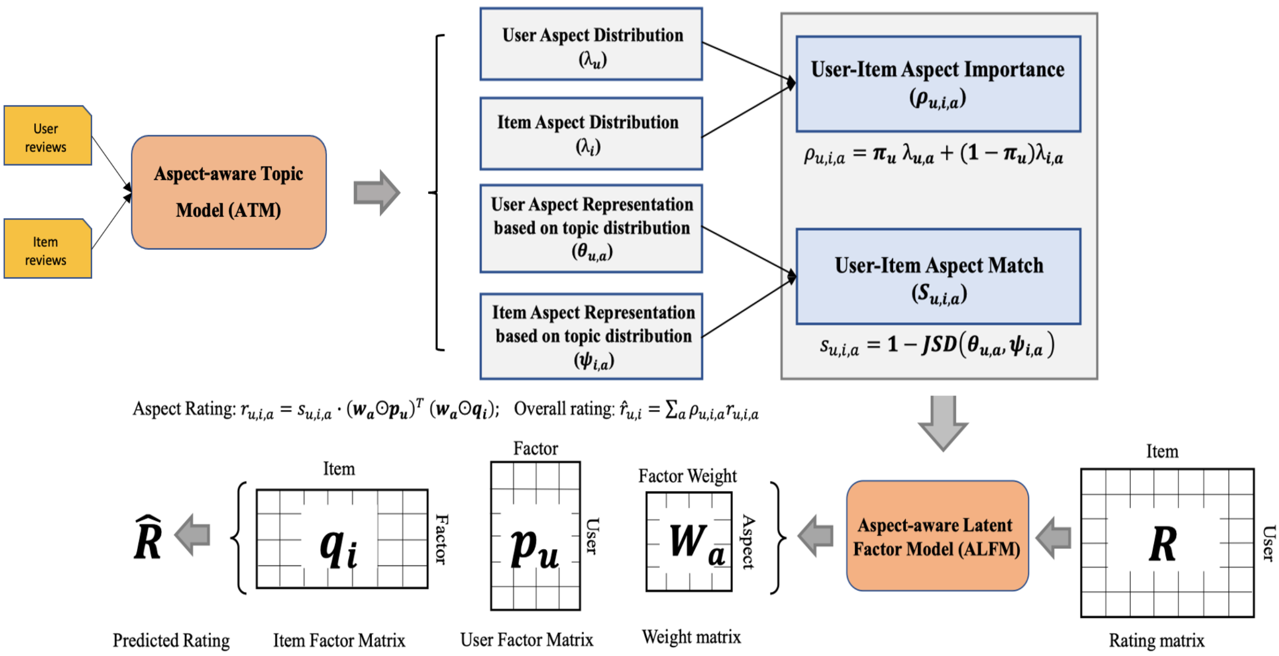


Figure 5.8: The structure of ALFM model [CCZ+19].

A3NCF: This is the state-of-the-art RS that fuses topic modeling and DL. This system firstly adopts LDA to obtain the topics vectors of users and items from textual reviews. Thus, it models users' and items' feature vectors in different topics (the aspects of items that users discuss in reviews) as probability distributions of words that refer to the same topic. For each user and item, their related topics vectors and embedding vectors (from ratings) are fused into an attention neural network to learn their final representation by considering the user's attention weights with respect to the different aspects of the target item. To get the embedding features for users and items, the model represents user and item as one-hot encoded vectors and then incorporates them into an embedding layer. Finally, an attentive interaction between the user's and item's final representations is fed into fully connected layers (MLP) with regression to predict the final ratings.

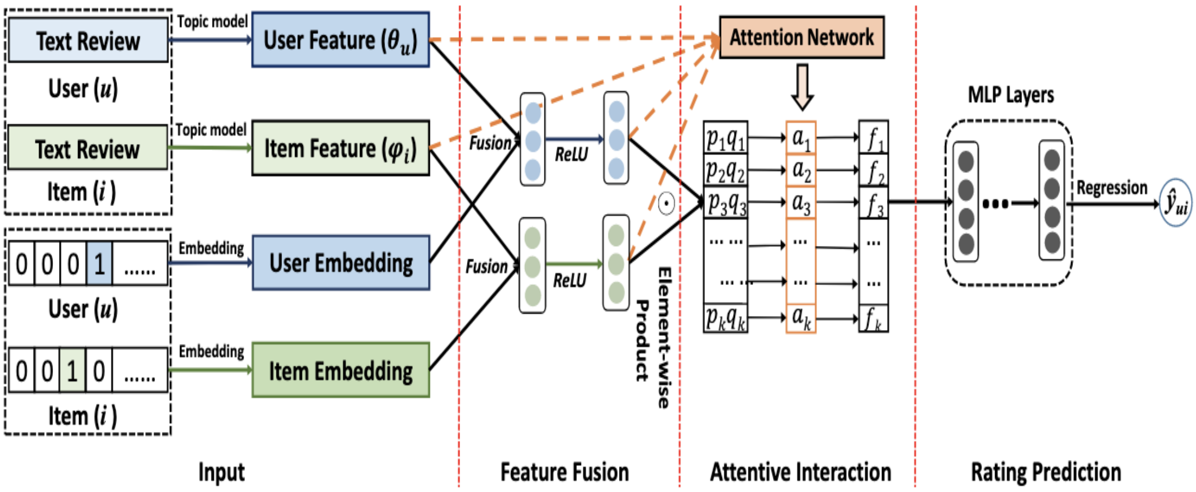


Figure 5.9: The structure of A3NCF model [CDH+18].

PARL: is a plug-and-play deep-learning architecture that has been plugged into DeepCoNN, one of the state-of-the-art review-based RS to improve its prediction accuracy upon different user-item pairs. DeepCoNN RS uses two parallel CNNs and a word embedding method for capturing latent representations from the reviews associated with the target user and item. The model concatenates the user and item vectors and then transmits it to a regression layer involving the FM method to predict ratings. Although DeepCoNN has shown good effectiveness for the rating prediction task, it remains limited. The review sparsity issue is one of the significant limitations when the texts of reviews are short and scarce. In this case, a few valuable features can be extracted by CNN from the incomplete text data. PARL extracts useful user-item pair-dependent features from auxiliary user reviews (written by other users with identical rating scores as the target user) to alleviate this limitation. Like DeepCoNN, PARL incorporates the extracted auxiliary reviews into CNN layers to transform them into feature vectors. To preserve the valuable features for each target user-item pair, PARL contains the obtained feature vectors into an abstracting layer involving a highway network and a gated mechanism. For final ratings, the auxiliary vectors are combined with the corresponding users' vectors in DeepCoNN.

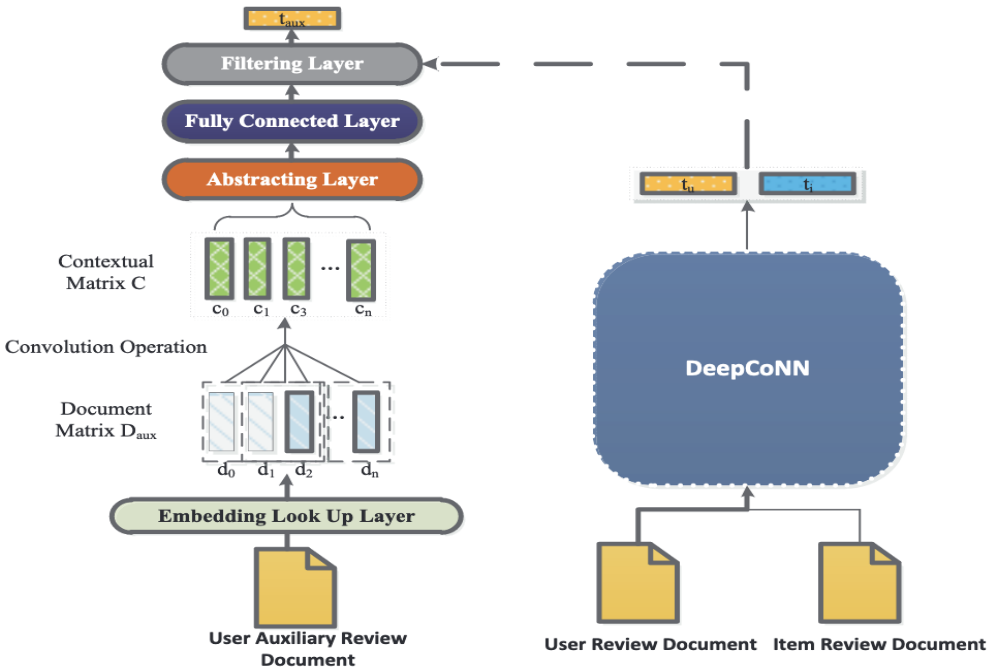


Figure 5.10: The structure of PARL model [WQLJ18].

CARL: This is the state-of-the-art RS that learns context-aware representations for each user-item pair based on their characteristics and their interactions by exploiting both the textual reviews and the user-item interaction data. CARL consists of two learning components, namely review-based feature learning and interaction-based feature learning. The first one adopts a CNN (using two parallels subnetworks) and an attention mechanism to jointly learn useful latent features for a user-item pair based on the user and item reviews. CARL also adopts an abstraction layer in this component to obtain pertinent latent features using an average pooling strategy. On the other hand, in the interaction-based part, complementary features are learned for each user and item based on their interaction data. CARL feeds each component's latent representation into the FM to produce ratings through each module. The final rating score is then computed based on a dynamic fusion strategy that fuses both components' ratings.

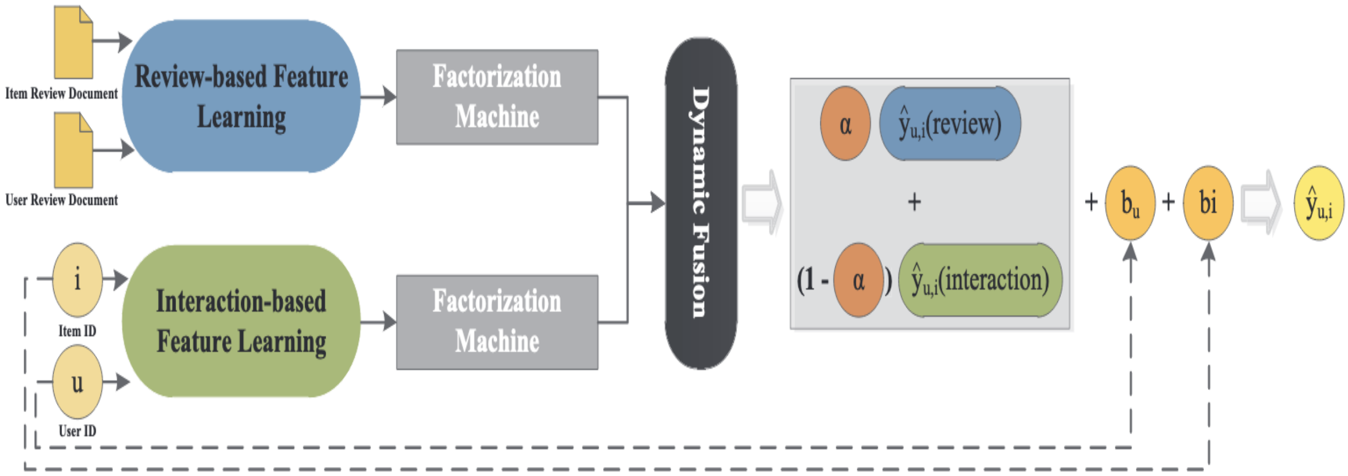


Figure 5.11: The structure of CARL model [WQL+19].

CARP: CARP represents the state-of-the-art RS that uses Capsule Network to extract the semantic contextual information from reviews for rating prediction. CARP is based on two modules: viewpoint and aspect extraction and sentiment capsules. The first component adopts a variant of self-attention stacked over a convolutional layer to capture the logic units formed by a given user viewpoint and an item aspect extracted from user and item reviews. In the second component, positive and negative capsules are exploited by a Routing by Bi-Agreement architecture to jointly choose some logic units as the informative ones and produce output vectors that encode their sentiments. Each created vector encodes the user's attitudes on a given item in the target sentiment. Besides, the lengths of the vectors suggest the probability of each of these two sentiments. Finally, to predict a user-item pair's missing rating, the magnitudes and odds in their two corresponding sentiment poles are incorporated into a one-layer highway network and then passed to a rescaled sigmoid function.

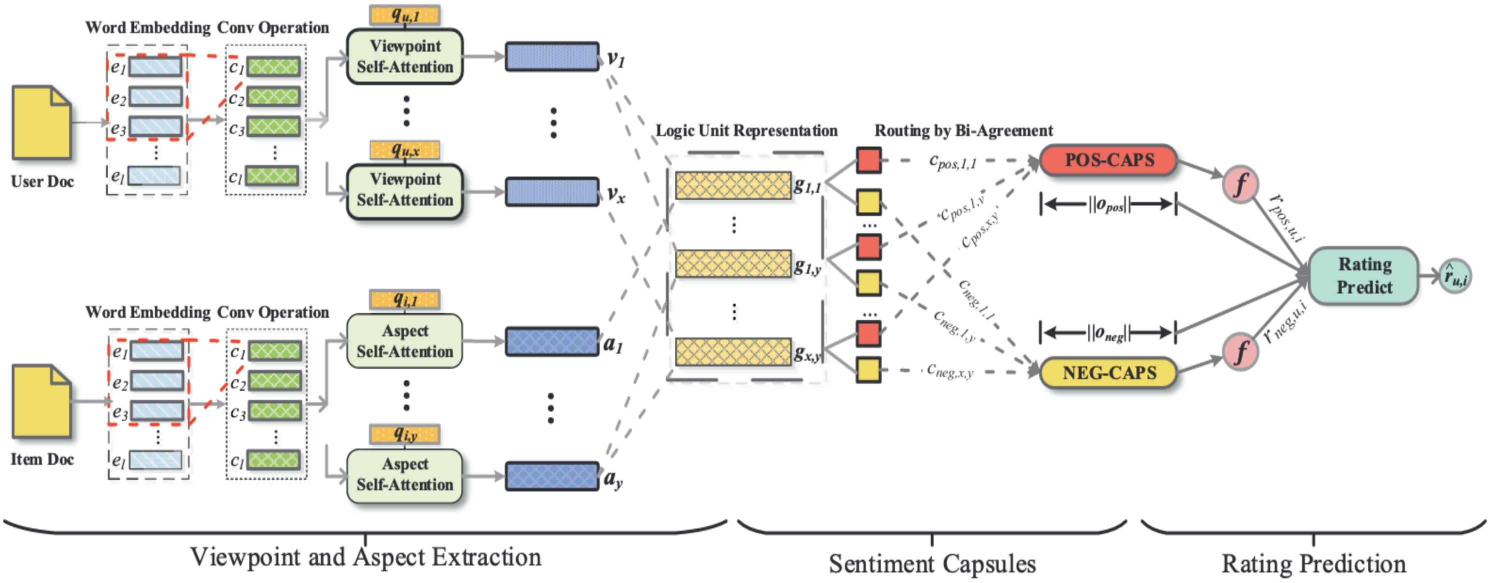


Figure 5.12: The structure of CARP model [LQP+19].

5.4 Experimental Setup

To answer the research questions stated in this work, we run our experiments on four English-language reviews datasets and their equivalent versions in the Arabic language. All these datasets were built based on the Data preparation and generation process described above.

We randomly selected 80% of each dataset as the training set for our evaluations and the remaining 20% as the test set. We trained the adopted models on the training set and evaluated the performance on the test set. To ensure that the testing set reviews are unavailable during the recommending process, such as in real-world applications, we utilized the review information only in the training set.

The parameters of all evaluated models (ALFM, A3NCF, PARL, CARL, and CARP) are set as reported in their related papers with the best performance.

We used RMSE as a performance metric for the evaluation metric, explained in Section 3.5 (chapter 3). This metric is broadly utilized in several related works for performance evaluation [LLX+20, AU19, HFT20].

5.5 Experimental Results

This Section presents the realized experiments to address our research questions (see Overview section).

The analysis of the attained empirical findings is organized as follows. We started by discussing the applicability of the modern RSs in the Arabic context. Next, we evaluated the effect of applying a preprocessing scheme on the Arabic texts before incorporating them into RSs. We ended up by assessing the performance of the recent RSs in the Arabic context.

5.5.1 Verifying the applicability of recent RSs in the Arabic context

This part aims to verify the possibility of applying the five adopted RSs in the Arabic context. Table 5.4 shows the performances of the five RSs ALFM, A3NCF, PARL, CARL, and CARP when applied to Arabic data. From the results, we can note that for the four Arabic datasets: Arabic_MI, Arabic_PLG, Arabic_Auto, and Arabic_IV, all tested RSs worked well, i.e., overall, they reached good RMSE scores. According to the achieved RMSE scores, we note that each used RS successfully predicted most of the hidden ratings when using Arabic reviews. Therefore, we can conclude from our results that it's possible to apply modern recommender engines to Arabic content.

Table 5.4: Performance on the constructed Arabic datasets in terms of RMSE.

Datasets	ALFM	A3NCF	PARL	CARL	CARP
Arabic_MI	0.883	0.977	0.891	0.882	0.881
Arabic_PLG	0.975	1.052	0.980	0.985	0.967
Arabic_Auto	0.895	0.993	0.914	0.901	0.894
Arabic_IV	0.978	1.031	1.020	0.990	0.986

5.5.2 Analyzing the impact of the preprocessing phase on recent RSs when applied to Arabic content

As done by other researchers in [ZAS$^+$17, HASA20], and due to the Arabic language's complex structure [SA10, ATK21, SAAES18], we have opted to preprocess the Arabic language reviews before incorporating them into the RSs. However, in this part, we aim to verify if adopting the preprocessing stage is necessary to apply these RSs on the Arabic datasets. To achieve it, we test the five RSs on different Arabic reviews datasets in both cases: using preprocessing stages and without using preprocessing tasks. The results are shown in Table 5.5 and Figure 5.13. By comparing the results achieved from the ALFM, A3NCF, PARL, CARL, and CARP in both cases, we can note that each of these RSs has maintained the same performance on each no-preprocessed reviews dataset (Arabic MI, Arabic PLG, Arabic Auto, and Arabic IV) and its preprocessed version (Arabic MI*, Arabic PLG*, Arabic Auto* and Arabic IV*), respectively. Those results confirm that the preprocessing stage is not essential for better accuracy and performance for the modern RSs when using Arabic texts. Such systems use effective feature extraction techniques to focus only on relevant information from the texts.

Table 5.5: Impact of the preprocessing phase on RSs in the Arabic context.

Datasets	ALFM	A3NCF	PARL	CARL	CARP
Arabic_MI	0.881	0.977	0.891	0.882	0.881
Arabic_MI*	0.883	0.978	0.891	0.882	0.881
Arabic_PLG	0.975	1.052	0.980	0.985	0.967
Arabic_PLG*	0.975	1.052	0.980	0.985	0.967
Arabic_Auto	0.895	0.993	0.914	0.900	0.894
Arabic_Auto*	0.895	0.993	0.914	0.900	0.894
Arabic_IV	0.978	1.031	1.020	0.990	0.986
Arabic_IV*	0.979	1.033	1.020	0.990	0.986

*: Preprocessed dataset

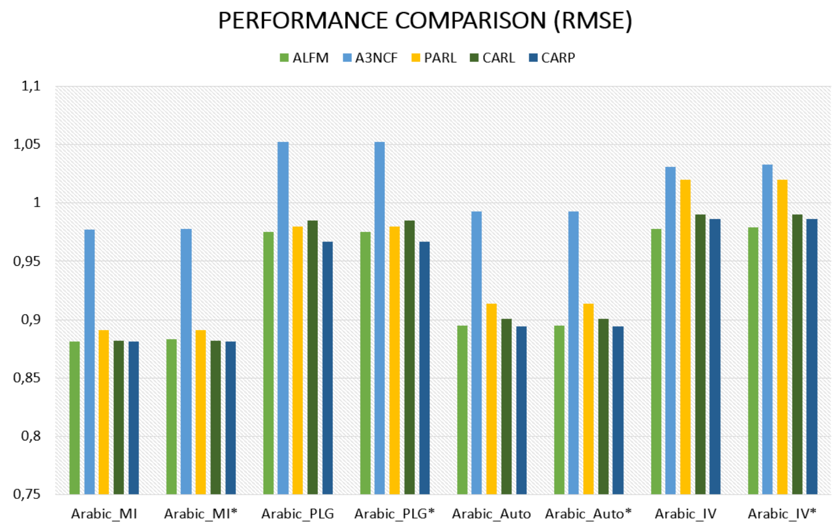


Figure 5.13: Impact of the preprocessing phase on RSs when applied to Arabic content.

5.5.3 Analyzing the performance of the recent RSs in the two contexts: English and Arabic

In this part, we compare the performance of the RSs (ALFM, A3NCF, PARL, CARL, and CARP) in two contexts: English and Arabic. In particular, we aim to verify whether the performance differences of the used recommendation models are statistically significant or not when changing the content's language. The experimental results are shown in Table 5.6 and Figure 5.14. From the experimental results, we can analyze the performances of the tested RSs. The results show that the RS ALFM has maintained very close performance for each English dataset and its Arabic version. The accuracy decrease on the four datasets pairs (MI and Arabic_MI, PLG and Arabic_PLG, Auto and Arabic_Auto, IV and Arabic_IV) is 0.21%, 0.21%, 0.11%, and 0.20%, respectively. Similarly, A3NCF has also maintained very nearly accuracy on each English dataset and its Arabic version. The accuracy decrease on the four datasets pairs is 0.21%, 1.06%, 1.12%, and 0.78%, respectively. Such accuracy differences are insignificant (very minor). We explain these because these RSs use bag-of-words techniques for review text processing, which is negatively impacted by the noise and irrelevant information introduced within the reviews during the translation phase. The results of this phase depend on the quality of the translation tool.

On the other hand, PARL, CARL, and CARP's performance do not degrade when changing the language of the review texts. Specifically, each of these three RSs has maintained the same rating prediction accuracy (accuracy decrease is 0%) on each English dataset and its Arabic version. We suggest that maintaining the same performance is due to processing the translated reviews by DL architectures, which help delete the noisy and unimportant information and develop appropriate feature representations for RSs. These results confirm the conclusions obtained in other studies for English content [KNSP21], which demonstrate that the modularity of neural networks allows handling heterogeneous and unstructured text content.

Considering those outputs, we can confirm that ALFM, A3NCF, PARL, CARL, and CARP do not lose performance when changing the application context (English to Arabic). Consequently, from these experimental results, we can conclude that applying the recent RSs to Arabic content provides good rating prediction accuracy as when using English content.

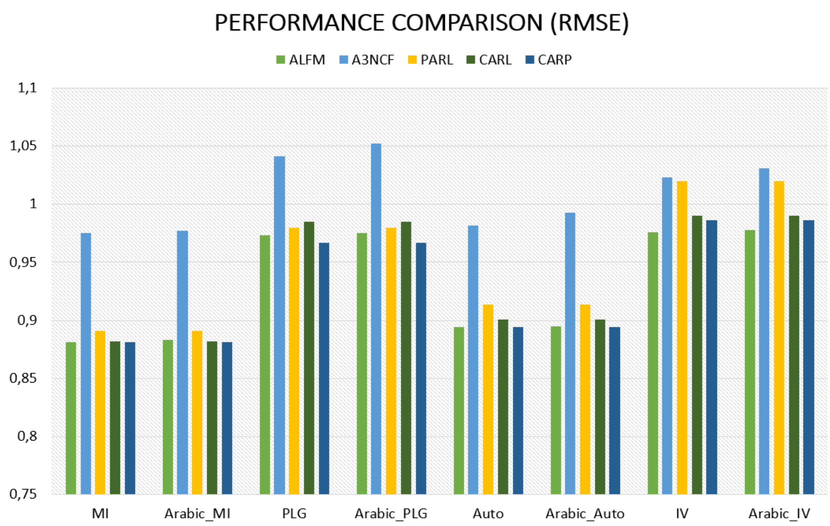


Figure 5.14: Comparing the adopted recommendation models in the English and Arabic contexts.

Table 5.6: Performance comparison on eight datasets in terms of RMSE.

Datasets	ALFM	A3NCF	PARL	CARL	CARP	$\triangledown\%$ (English vs Arabic)				
						ALFM	A3NCF	PARL	CARL	CARP
MI	0.881	0.975	0.891	0.882	0.881	0.21%	0.21%	0%	0%	0%
Arabic_MI	0.883	0.977	0.891	0.882	0.881					
PLG	0.973	1.041	0.980	0.985	0.967	0.21%	1.06%	0%	0%	0%
Arabic_PLG	0.975	1.052	0.980	0.985	0.967					
Auto	0.894	0.982	0.914	0.901	0.894	0.11%	1.12%	0%	0%	0%
Arabic_Auto	0.895	0.993	0.914	0.901	0.894					
IV	0.976	1.023	1.020	0.990	0.986	0.20%	0.78%	0%	0%	0%
Arabic_IV	0.978	1.031	1.020	0.990	0.986					

$\triangledown\%$: Accuracy decrease (pair of English-Arabic datasets)

5.6 Chapter Summary

This chapter provides a comprehensive evaluation of modern RSs when applied to Arabic content. For that, extensive experiments have been conducted using different Arabic reviews datasets. These datasets were built leveraging the English-language reviews regarding different product categories in the Amazon e-platform. Thus, we introduced Arabic_MI, Arabic_PLG, Arabic_Auto, and Arabic_IV, the largest Arabic datasets known to date for investigating RSs. We described the construction steps of those datasets, and then we provided their main properties. The experiments were performed utilizing five recent RSs mainly proposed to manage English content. Each of these systems was tested on different-language datasets: original English ones and their corresponding Arabic versions, which we have constructed. This study was conducted to achieve three objectives. The first aim is to verify the applicability of recent RSs to the Arabic content. According to our experimental results, we have shown that the used RSs could perform rating prediction tasks while exploiting Arabic data. The second goal was to determine if applying a particular preprocessing phase on the Arabic content before incorporating it into RSs. We tested each of the RSs on Arabic reviews in two cases: preprocessing and without preprocessing. The results showed that the preprocessing stage does not impact RSs' performance. The third objective of this work is to evaluate if the performance of these RSs varies when changing the application context from the original one (English) to Arabic. The experimental findings proved that the adopted RSs had maintained the same performance in both contexts.

Chapter 6

A Novel Deep learning-Based Recommender System Adapted to Arabic Content

Contents

6.1 Introduction

Recently, DL has been revolutionizing the recommendation paradigms dramatically and offering more capabilities to augment the performance of RSs. It is now widely proved that DL methods can capture the non-trivial and nonlinear user-item interactions [LZL+15] and codify more complicated abstractions as data representations in the higher layers [ZYST19]. Therefore, a prevailing trend in the RS field has leveraged those techniques to tackle diverse issues of the classical RS methods and achieve high accuracy performance [BYBK19]. Broadly, the success of DL in RSs is due to two main aspects: representation learning and nonlinear transformation. Concerning the representation learning aspect, the entity embeddings, i.e., users and items, can be substantially improved with modern tools in DL. For instance, CNNs are exploited to enhance the item and/or user representation learning from different sources, including textual reviews [KPO+16, ZNY17, CC17, WQLJ18, CZLM18, SHYL17]. For the other aspect of the nonLinear transformation, DL approaches often utilize single or multi-level neural architectures as the interaction function, allowing to generalize the classical MF methods to nonlinear settings, which are more expressive than the linear MF ones. Multi-Layer Perceptron (MLP) is the most popular among the various existing techniques due to its strong capability for capturing both the second-order and higher-order feature interactions.

DL techniques have progressively become the most largely adopted ones in the field of RS, thus achieving promising results on the preferences prediction task, matching, or even beating those human performance provides. However, these significant advancements have only been developed to work in the English context, particularly for managing information overload using English-language reviews. This leads to an overall lack of advanced and performant tools for exploiting reviews written in other languages like the Arabic one. Therefore, in this chapter, we aim to tackle this research gap by providing a novel Arabic RS while exploiting DL methods initially proposed for the English community. This system is also essentially developed to overcome the different limitations shared between the existing works in the Arabic RS field (refer to section) through the consideration of the following novel aspects:

- Using large-scale Arabic data for investigating the proposal.

- Using various and different datasets for the experimentations.

- Comparing with solid and recent baselines in the field for performance evaluation.

In what follows, we present and detail our proposed Arabic RS. We first provide an overview of this proposal. Second, we give the followed steps to build it. Finally, we describe the results of our experiments and their interpretations.

6.2 Overview of the Proposed RS

The main aim of this study is to design a RS exploiting large-scale Arabic reviews datasets for rating prediction in the Arabic context. It predicts the users' interests according to their textual reviews written in the Arabic language. Thus, our system uses the Arabic reviews dataset as input. Specifically, it takes a training set T as input consisting of N Reviews. Each review can be represented as a $(u, i, r_{u,i}, d_{u,i})$ where $r_{u,i}$ is a numerical rating denoting user u's preference on item i, and $d_{u,i}$ is the accompanying textual review. Then, from the input data, the proposed RS predicts the unknown ratings of items that the users have not reviewed yet. The idea is to learn latent representations for users and items from their reviews' text such that the learned representations can estimate their missing ratings. This is achieved by first using a text processing technique for modeling users' preferences and items' characteristics from reviews, then incorporating the resultant models into a recommender module for rating prediction. The implementation of the two components of our system was carried out using DL paradigms. Figure 6.1 gives the overall architecture of the proposed system.

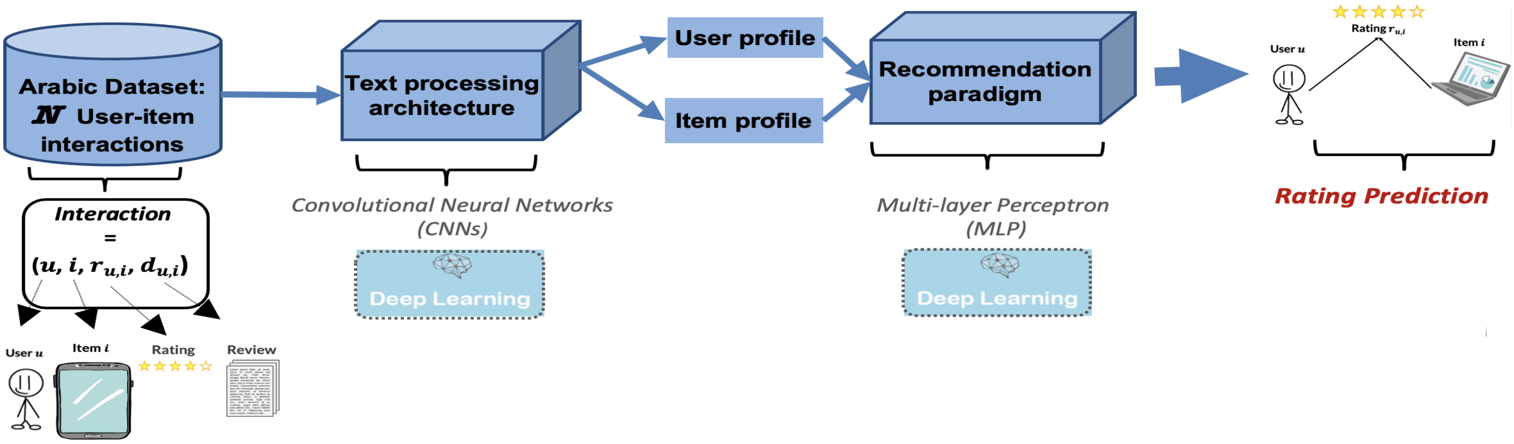


Figure 6.1: The overall architecture of the proposed system.

6.3 Steps of the Proposed RS

The proposed RS consists of two main steps: representation of users and items and generating predictions. These steps are shown in Figure 6.2.

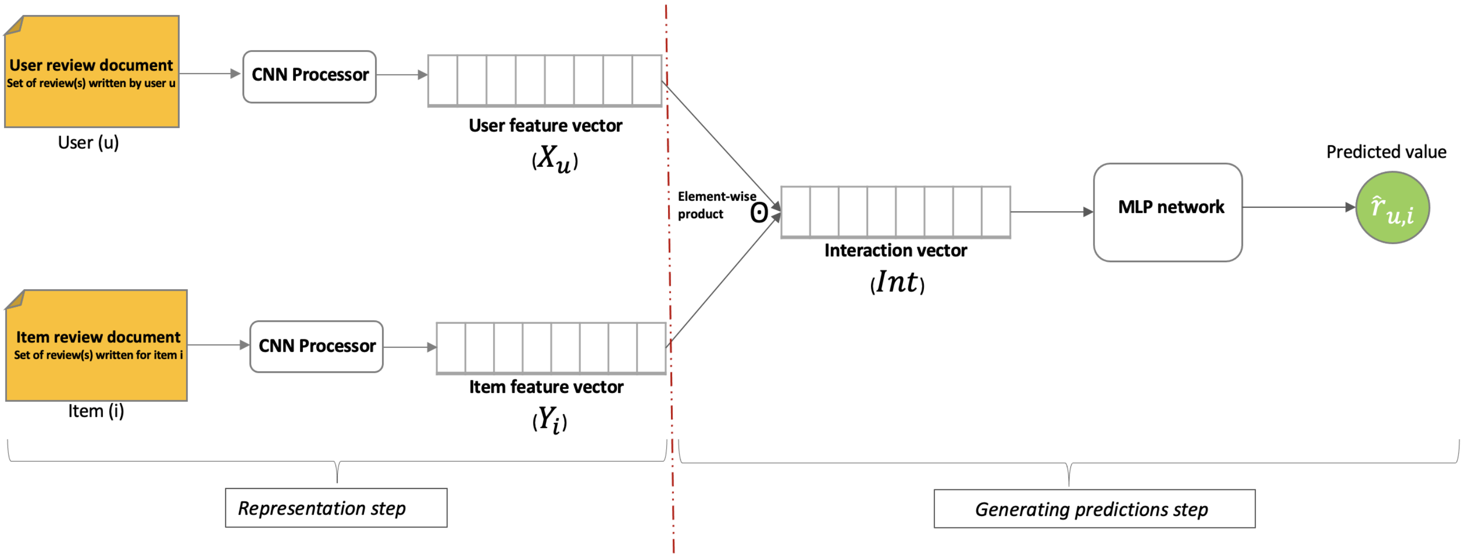


Figure 6.2: Steps of the proposed RS.

6.3.1 Representation of users and items

This first step of our RS aims to model the targeted user and item based on their corresponding reviews. Its main objective is to prepare the users' and items' representations for the rating prediction stage. As mentioned above, many works are proposed to model information from reviews for rating prediction using text processing methods based on DL. One of the pioneering works in this line of literature is [ZNY17]. It utilizes two parallel CNN Text processors, one for user modeling and another for item modeling. This method concatenates all reviews from the same user as a user review document. In the same way, it merges all reviews on the same item to form an item review document. Then, the constructed user and item review documents are passed to the user text processor and the item text processor, respectively, as inputs and corresponding representations are generated. In this work, the adopted CNN architecture has shown its effectiveness in modeling users and items from their reviews by considering semantic information. It has also proven to increase rating prediction accuracy better than traditional text processing techniques. However, this method has demonstrated its effectiveness only on English texts and has never been tested on Arabic texts. Motivated by those reasons, we opt to process our Arabic reviews using the same strategy exploited in [ZNY17]. In this context, to our knowledge, we are the first to use this text processing component for modeling users and items based on Arabic textual content. In what follows, we present the details about the user modeling process (to obtain the user's feature vector X_u). The same process is applied for modeling the item (to get the item's feature vector Y_u). Figure 6.3 gives the architecture of the adopted process to model both users and items.

Given a user review document [1] $RevDoc = [w_1, ..., w_n]$, an embedding lookup layer is exploited to project each word in $RevDoc$ into a d dimensional embedding vector $e_i \in \mathbb{R}^{1\times d}$. After that, an embedding matrix $RevDoc \in \mathbb{R}^{n\times d}$ is constructed by concatenating all the obtained embeddings:

[1]The user review document is constructed by merging all the reviews written by this target user. In the case of the item modeling process, we built the item review document by concatenating all the reviews made by the users for this target item.

$$RevDoc = [e_1..., e_k, ..., e_n]^T \tag{6.1}$$

where e_k represents the word embedding corresponding to the word at the k^{th} position in the document RevDoc.

The next layer is the convolutional layer. It consists of m neurons to produce contextual features by applying convolution operation on word vectors in the document matrix $RevDoc$. Each neuron in this layer utilizes a filter $K \in \mathbb{R}^{d \times t}$ with the same sliding window of size t. It generates its features as:

$$f^j = Relu(RevDoc * K_j + b_j) \tag{6.2}$$

where b_j is the bias, $*$ is the convolution operation and $Relu$ [NH10] is a nonlinear activation function.

After obtaining the features f^j corresponding to filter K_j, a max-pooling layer is adopted to capture the most informative contextual feature of each filter.

$$c_j = max(f^j_1, f^j_2, f^j_3, ..., f^j_{n-t+1}) \tag{6.3}$$

Finally, all captured features from the m neurons are concatenated as the output of the max-pooling layer:

$$Out = [c_1, c_2, c_3, ..., c_m] \tag{6.4}$$

After that, the output Out is feed into a fully connected layer f with a weight matrix $W \in \mathbb{R}^{m \times n}$ and a bias $s \in \mathbb{R}^n$ to derive the latent representation X_u of target user u as:

$$X_u = f(W \times Out + s) \tag{6.5}$$

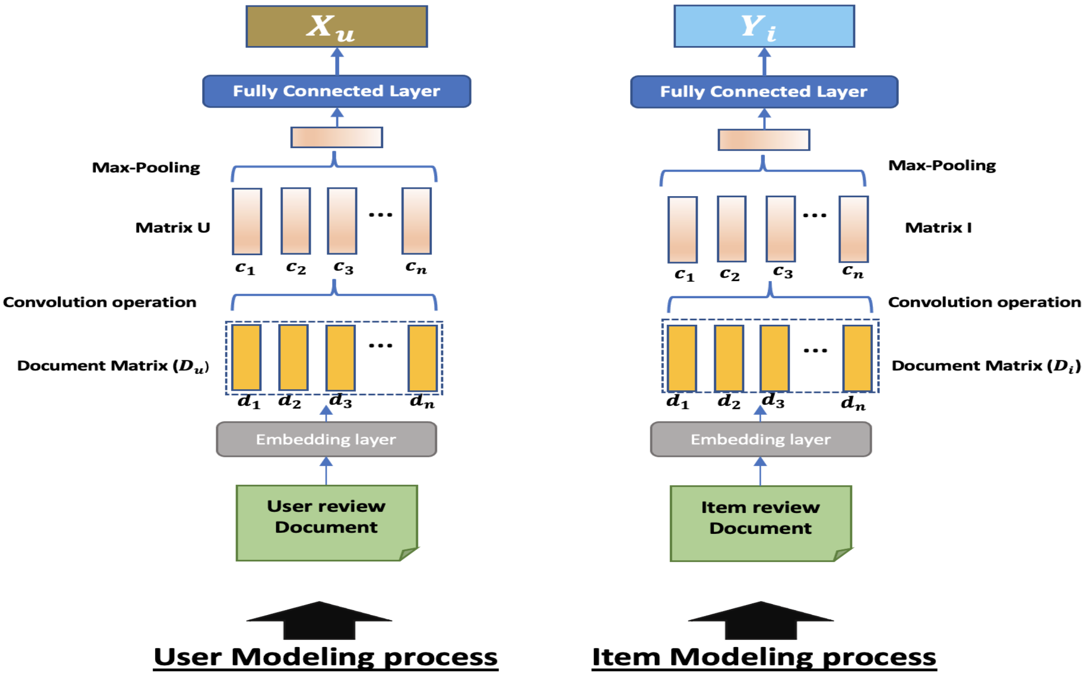


Figure 6.3: The adopted text processing architecture.

6.3.2 Generating predictions

After representing users and items, the next step is prediction generation. This step's fundamental objective is to integrate the learned representations of the target user and item, namely, X_u and Y_i, into a recommender model for predicting their unknown interactions.

Inspired by the successful applications of DL methods in the RS field, we opt to perform this task by adopting a neural network structure. Specifically, we decide to adopt the MLP architecture. MLP is a feed-forward neural network model consisting of one/more layer(s) and nonlinear activations. MLP comprises at least one hidden layer interconnected in a feed-forward direction (see Figure 6.4). It is used as the building block of most of the DL models [XCL$^+$16, XLC$^+$17, SYCX18]. The central assumption of MLP relies on introducing the nonlinear settings into the linear models for neural performance. It has been adopted in many RSs in the literature.

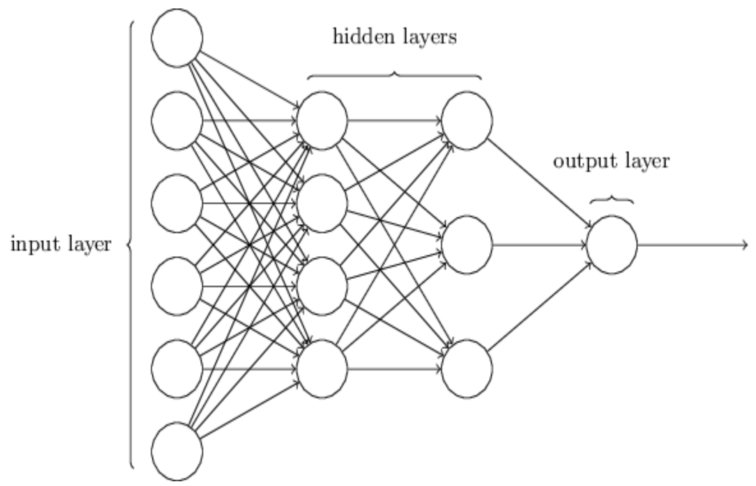


Figure 6.4: A structural architecture of the MLP network.

Specifically, given X_u and Y_i, we first model the interaction between user u and item i as follows:

$$Int = X_u \cdot Y_i \tag{6.6}$$

where $\cdot$ denotes the element-wise product of vectors. The output of Int represents n dimensional predictive vector. This vector can be viewed as the representation for the corresponding user-item pair.

Then, to predict the rating $\hat{r}_{u,i}$ for the item i that was not yet evaluated by the user u, we feed Int into a MLP network with one hidden layer (see Figure 6.5):

$$
\begin{aligned}
z_1 &= \Phi(X_u, Y_i) = W_1 \times Int + b_1 \\
\Phi_2(z_1) &= \sigma_2(W_2^T z_1 + b_2) \\
&\dots \\
\Phi_L(z_{L-1}) &= \sigma_{L-1}(W_L^T z_{L-1} + b_L)
\end{aligned}
\tag{6.7}
$$

$$
\hat{r}_{ui} = \underbrace{\sigma(h^T(\Phi_L(z_{L-1}))}_{\text{Pred}} + b_u + b_i
$$

where W_x , b_x , and σ_x denote the weight matrix, bias vector, and activation function for the x-th layer's perceptron, respectively. b_u and b_i refer to user bias and item bias, respectively.

These settings allow combining two independent components into one system: a CNN-text processor for representing users and items; and a neural network, in particular, a MLP for rating prediction.

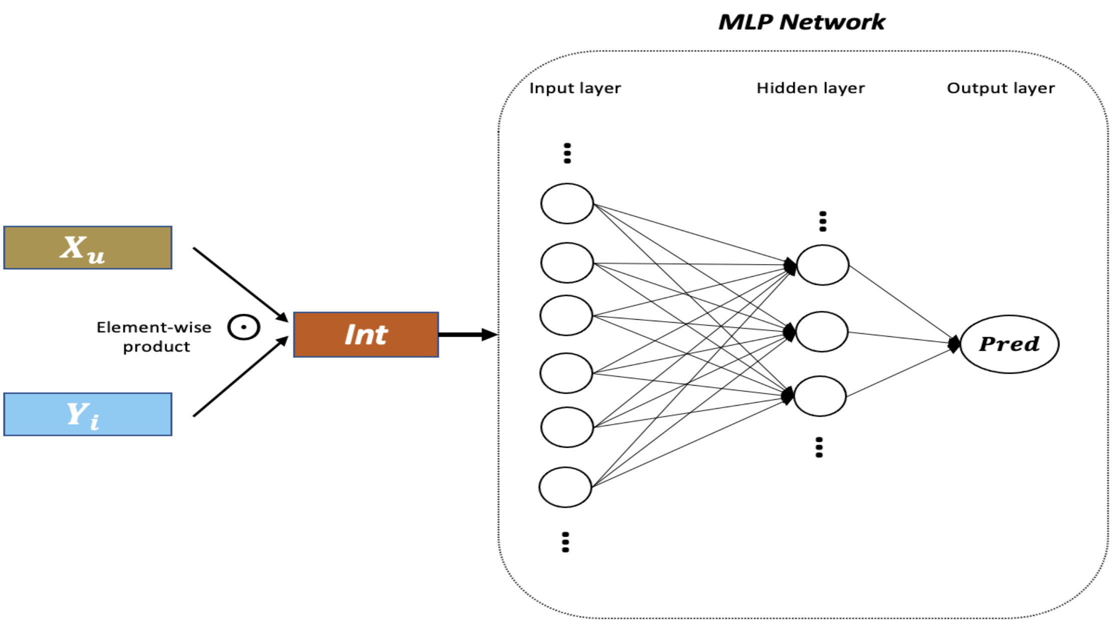


Figure 6.5: The architecture of the adopted MLP network.

6.3.3 RS optimization

The main aim of our proposed RS is to perform rating prediction. Since this task represents a regression problem, we utilize the following squared loss function to train our RS on the set of available ratings as follows:

$$L = \sum_{u,i \in T} (\hat{r}_{u,i} - r_{u,i})^2 + \mu(||x_u||^2 + ||y_i||^2) \tag{6.8}$$

where T denotes the set of instances for training, and $r_{u,i}$ is the real rating assigned by the user u to the item i. μ is a regularization parameter. x_u and y_i are bias vectors for user u and item i respectively. Our RS's main objective is to predict the unknown ratings with minimum prediction error (L). Equation (8) is used to optimize the parameters with Stochastic Gradient Descent (SGD) and backpropagation. To update the parameters, RM-Sprop [TH+12] is used over mini-batches. Besides, the dropout [SHK+14] strategy is adopted to the MLP layer for preventing overfitting. Figure 6.6 shows how the optimization process is performed.

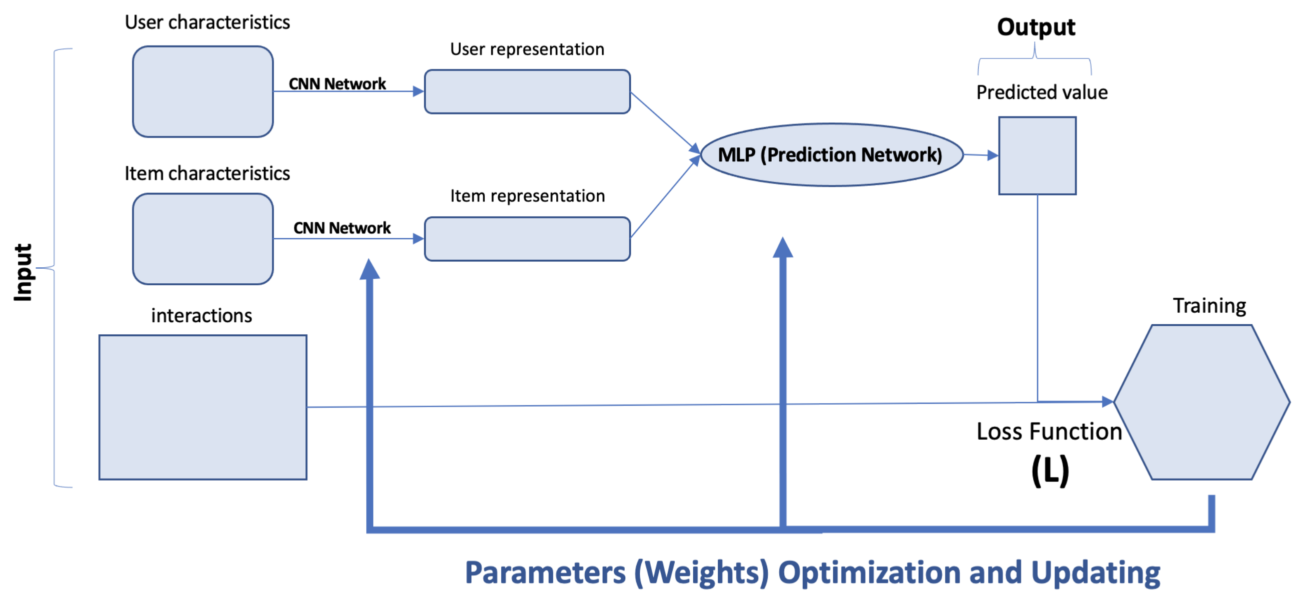


Figure 6.6: RS optimization.

6.4 Experimental Settings

This section presents the performed extensive experiments to evaluate the performance of our proposed RS compared to other state-of-the-art RSs. In what follows, we first introduce the datasets end the evaluation metrics used in our experiments. After that, we describe the baseline systems selected for comparisons. Finally, we discuss and analyze the experimental results.

6.4.1 Datasets and Evaluation metrics

To evaluate the performance of our RS, we conducted experiments on our four constructed and preprocessed Arabic datasets, namely: Arabic_MI, Arabic_PLG, Arabic_Auto, and Arabic_IV (see chapter 5).

Before using these datasets in our experiments, we preprocessed them based on the preprocessing scheme presented in chapter 5 (Section 5.2). We then form the review documents for all datasets by setting the length of each document to 300 words.

For evaluation, we divided each dataset into training and testing sets utilizing an $80 : 20$ division, respectively. We trained the models on the training sets and evaluated the performance on the test sets.

The performance of the used RSs is evaluated by two well-known metrics, namely MSE and MAE, which are described in section 3.5 (chapter 3). These two metrics are widely used to measure the prediction accuracy of RSs [ZNY17, ML13, PME19, FEAD21].

6.4.2 Baselines

We chose the following three state-of-the-art methods as our baselines. These methods were initially proposed for English content and have never been tested on Arabic content. This work compares the performance of these widely used methods when used in the Arabic context.

GMF [HLZ+17]: GMF generalizes the standard MF to a non-linear setting. It uses a neural layer above the element-wise of user's and item's embeddings (deduced based on the one-hot encoding strategy) to compute the rating that the user would score the item. It has been proved in many studies that GMF outperforms the standard MF methods. It is typically a representative baseline in CF RSs.

A3NCF [CDH+18]: utilizes the LDA topic model to represent user preferences and item characteristics from review documents. It then integrates them into a network framework to estimate the unknown interactions of users and items.

PARL [WQLJ18]: is an extension of DeepCoNN. It utilizes auxiliary reviews to relieve the sparsity issue concerning users. It adopts the same network architecture of DeepCoNN while both the user review vector and the auxiliary user vector are combined to form the final user latent vector.

We omit comparisons with early methods, such as HFT [ML13], RMR [LLK14], RBLT [TZLM16], ConvMF [KPO+16], DeepCoNN [ZNY17] and TransNets [CC17], because these models have been demonstrated to be inferior to our baselines A3NCF or PARL.

6.4.3 Implementation details

All methods were implemented using Python DL Frameworks. Specifically, the TensorFlow library [2] was used to develop our RS and PARL baseline. GMF and A3NCF were developed based on Keras library [3] using Theano [4] as backend.

We used 0.002 as the learning rate, 0.5 as the dropout probability, 0.01 as μ. We set the mini-batch size to 100 for all datasets. The $Relu$ activation function [NH10] is used in the layers of the MLP network. The number of MLP layers L in the rating prediction component of ou RS is fixed to 2. For the CNN architecture adopted in our RS, we utilized the same hyperparameter tuning as reported in

[2]https://www.tensorflow.org/?hl=fr
[3]https://keras.io/
[4]https://github.com/Theano/Theano

the corresponding paper [ZNY17] since changing it did not give any significant amelioration. We randomly initialized the word embeddings and fixed the number of convolution filters to 25.

For the baselines, we used the same hyper-parameter settings as the authors.

For all models, the dimensions for the user and item latent vectors were evaluated on $8, 16, 32, 64$.

6.5 Experimental Results

The rating prediction performance of our RS and the baselines are reported in terms of MSE and MAE in Tables 6.1 and 6.2. These Tables show the results on four large-scale Arabic datasets when the dimension for the user and item latent vectors in all systems is fixed to 8. The last column in the Tables indicates the improvement percentage of our RS over the baselines. From Tables 6.1 and 6.2, we make the following observations.

Table 6.1: Performance comparison on four datasets in terms of MSE.

Datasets	**GMF**	**A3NCF**	**PARL**	**Our RS**	**Improvement (%) Our RS vs Baselines**		
					GMF	**A3NCF**	**PARL**
Arabic_MI	0.811	0.806	0.782	0.774	4.56 %	3.97%	1.02 %
Arabic_PLG	1.104	1.020	0.958	0.950	13.95 %	6.86%	0.84 %
Arabic_Auto	0.883	0.827	0.801	0.782	11.44 %	5.75%	2.37 %
Arabic_IV	1.150	0.970	1.049	0.955	16.96 %	1.57%	8.96 %

Table 6.2: Performance comparison on four datasets in terms of MAE.

Datasets	GMF	A3NCF	PARL	Our RS	Improvement (%) Our RS vs Baselines		
					GMF	A3NCF	PARL
Arabic_MI	0.683	0.670	0.650	0.640	6 %	4.69 %	1.23 %
Arabic_PLG	0.840	0.779	0.735	0.734	12.62 %	5.78 %	0.14%
Arabic_Auto	0.722	0.668	0.647	0.623	13.71 %	7.22 %	3.71 %
Arabic_IV	0.835	0.738	0.766	0.737	11.74 %	0.14 %	3.79 %

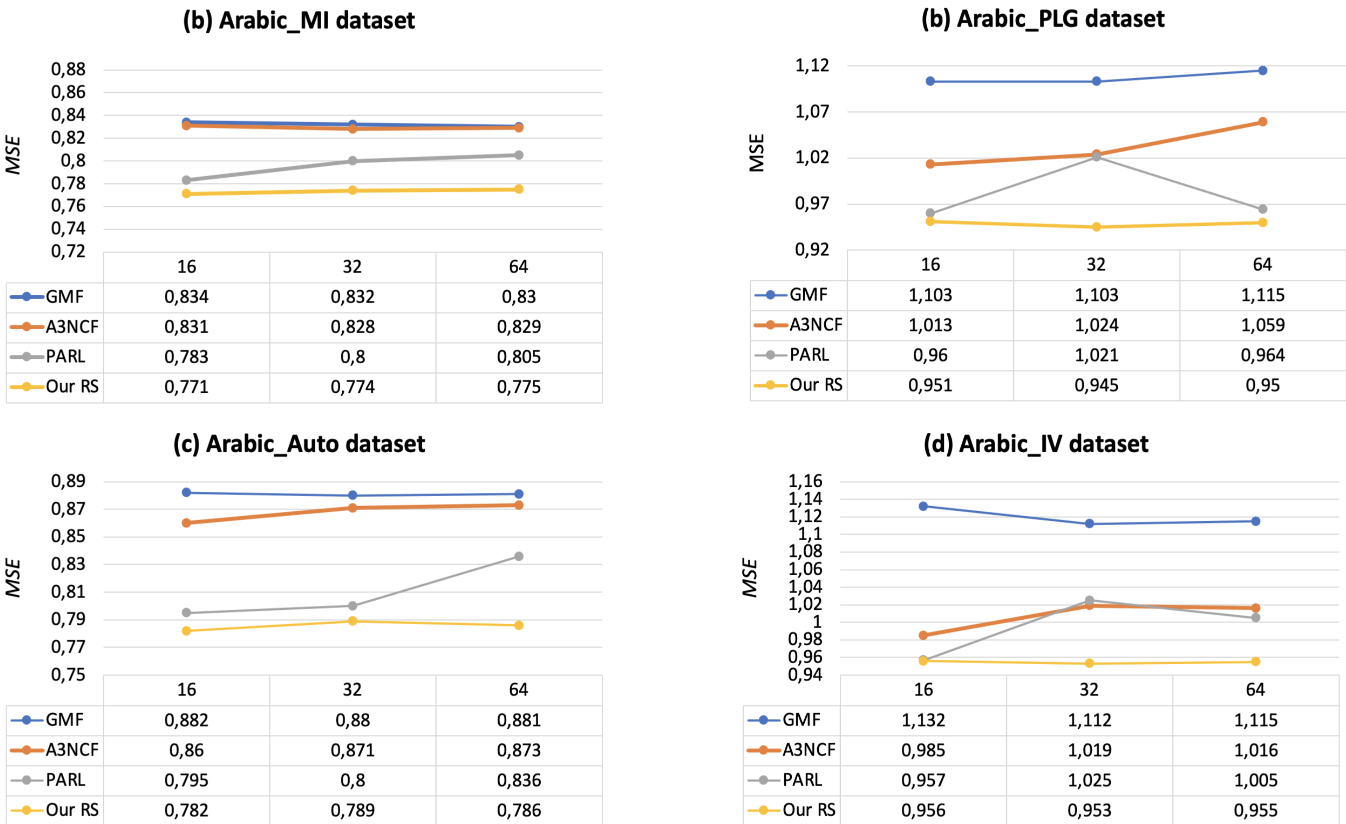
Firstly, we can note that for all datasets: Arabic_MI, Arabic_PLG, Arabic_Auto, and Arabic_IV, all used systems performed well, i.e., overall, it achieved good MSE and MAE scores which are comparabale with those reached by many state-of-the-art RSs devoted to English content [CDZK18, TLH18, DMD21]. These results can validate the applicability and effectiveness of modern RSs in the Arabic context.

Second, we can see that the methods using review information achieve better performance than GMF in all four datasets: Arabic_MI, Arabic_PLG, Arabic_Auto, and Arabic_IV. In particular, the MSE improvements of our RS as compared to GMF are very impressive, showing a 4.56%, 13.95%, 11.44%, and 16.96% improvement on the four datasets, respectively. Besides, our RS reaches the most significant MAE improvements (6%, 12.62%, 13.71%, and 11.74%) compared to GMF on the four datasets, respectively. Those results confirm that reviews can be used to learn fine-grained profiles of users/items and improve prediction quality. This observation is in line with the results reported in several review-based works [KPO$^+$16, ZNY17, CZLM18, LWP$^+$20].

Third, for the review-based baselines, PARL is the strongest one on all datasets. It achieves the second-best performance on Arabic_MI, Arabic_PLG, Arabic_Auto, and Arabic_IV datasets. We believe the reasons are as follows. First, its CNN text processing architecture ensures that semantic features discussed in reviews can be extracted successfully for modeling users and items. This helps to accurately reflect the user's preferences and item's properties hidden in the reviews. However, because A3NCF neglects the semantic information contained in reviews when building the topic representations of users and items, its performance is inferior to that of PARL. Besides, PARL exploits auxiliary reviews to deal with the sparsity of original reviews based on a CNN structure. By leveraging such supplementary information, extra informative features can be extracted, improving thus the prediction accuracy.

Fourth, our system reaches the best MSE and MAE scores across all datasets. From the last columns, we can observe that the improvements gained by our system over A3NCF and PARL baselines are consistent and stable. In particular, our system performs much better than the best baseline on the Arabic_MI, Arabic_PLG, Arabic_Auto, and Arabic_IV datasets in terms of MSE with improvements of 1.02%, 0.84%, 2.37%, and 8.96%, respectively. Also, we can observe that the relative improvement of the proposed RS against PARL in terms of MAE on the four datasets are, respectively, 1.23%, 0.14%, 3,71%, and 3.79%. Those results confirm that the proposed system is sufficiently appropriate and powerful for rating prediction on Arabic datasets with different characteristics.

Through another set of experiments, we also analyzed the behavior of the proposed RS and the baselines when changing the dimension size of users and items latent representations. Figures 6.7 and 6.8 plot the performance of all systems by varying the latent dimension size l in $\{16, 32, 64\}$. From the experimental results shown in those Figures, we can observe that compared to GMF, review-based methods (A3NCF and PARL) perform relatively well across all the range of l values on the four datasets. Also, PARL performs significantly better than A3NCF with all the varying l values in all the datasets. However, although, PARL baseline uses auxiliary reviews, which is the major extension of the adopted CNN architecture, it consistently performs worse than the proposed system. Contrary to our system, PARL uses a FM which is a linear model to capture interactions between user-item pairs. This fact makes it impossible to capture the non-trivial user-item interaction patterns and thus limits the modeling performance of such models.



(b) Arabic_MI dataset

	16	32	64
GMF	0,834	0,832	0,83
A3NCF	0,831	0,828	0,829
PARL	0,783	0,8	0,805
Our RS	0,771	0,774	0,775

(b) Arabic_PLG dataset

	16	32	64
GMF	1,103	1,103	1,115
A3NCF	1,013	1,024	1,059
PARL	0,96	1,021	0,964
Our RS	0,951	0,945	0,95

(c) Arabic_Auto dataset

	16	32	64
GMF	0,882	0,88	0,881
A3NCF	0,86	0,871	0,873
PARL	0,795	0,8	0,836
Our RS	0,782	0,789	0,786

(d) Arabic_IV dataset

	16	32	64
GMF	1,132	1,112	1,115
A3NCF	0,985	1,019	1,016
PARL	0,957	1,025	1,005
Our RS	0,956	0,953	0,955

Figure 6.7: Performance comparisons across different dimensions of latent vectors on the four Arabic datasets in terms of MSE.

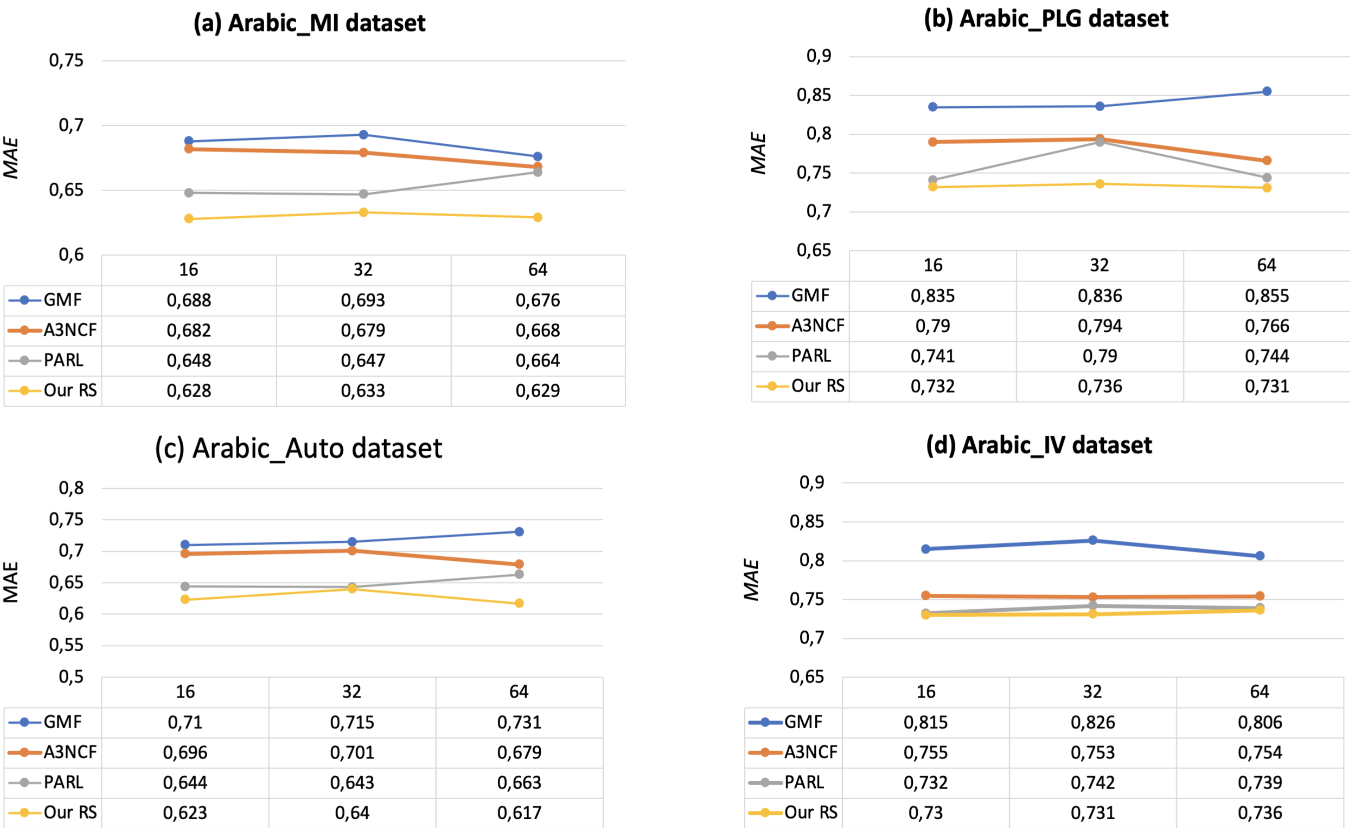


Figure 6.8: Performance comparisons across different dimensions of latent vectors on the four Arabic datasets in terms of MAE.

The proposed RS achieves better performance (lowest MSE and MAE values) than the baseline models across all l values range (i.e.,[16,64]). Even when varying the size of latent vectors, our RS maintains relatively stable performance. We believe that the substantial improvement of our model over the baselines could be credited to the fact that our RS reaps the benefits from an effective combination of two components: a CNN-based representation learning and a MLP network for capturing interactions between user-item pairs. From all the experimental results in Tables 6.1, 6.2, and Figures 6.7, 6.8, we can confirm that these two independent components complement each other. By leveraging convolution and pooling layers in the CNN component, latent features of users and items were efficiently captured from the Arabic-language reviews. This component allows capturing relevant semantic features from the texts by considering contextual information, which are essential for improving the understanding and modeling in the RS field. Using the generated representations as inputs of the MLP component, deep user-item interactions were captured by stacking two neural layers. This can efficiently capture the complex nonlinear relations between user and item representations, which conducts to a good prediction quality. Therefore, it is reasonable to conclude that the realized combination effectively predicts rating in the Arabic context. More precisely, it is suitable for exploiting a significant volume of Arabic data.

6.6 Chapter Summary

This chapter presented an Arabic RS designed for rating prediction in the Arabic context. Its main goal is to leverage user reviews written in the Arabic language for generating predictions. We have combined a CNN text processing method with a MLP-based structure to develop our RS. These models were never fused or used before to build a RS for rating prediction exploiting Arabic content. The CNN architecture was employed to represent user preferences and item characteristics according to their corresponding textual reviews. The obtained user's and item's representations were used as inputs to a MLP network with a single hidden layer to predict user preferences on items. The experimental results show that the proposed RS consistently outperforms state-of-the-art alternatives over four

Arabic datasets. Additionally, in the experiments, we demonstrated that our RS maintains the most stable and strong performance in terms of prediction accuracy by varying the dimension of feature vectors of users and items. This proved that the realized combination (CNN+MLP) is promising in prediction quality in the Arabic context, motivating us to continue working in this direction to advance the Arabic RS field.

Part IV

Closing

Chapter 7

Conclusion and Perspectives

Contents

7.1 Conclusion

The RS field has recently become one of the growing research areas. Research on Arabic RSs has received minimal attention compared to studies on English RSs. This thesis aims to boost and advance this field by exploring and examining several major research questions summarized as follows:

• How about the Arabic content evolution and presence on the Internet?

• How about the importance of proposing and exploring new functionalities to exploit the Arabic content?

• Are there RSs in the literature dedicated to handling Arabic content? If so, how about their performance and efficiency compared to available RSs developed to manage content in other languages like English?

• Are there available datasets containing Arabic content for exploring RSs in an Arabic context?

• Is it possible to apply recent RSs devoted primarily to English content to the Arabic content? If so, how about their performance and efficiency when exploiting such content?

• How should Arabic content be treated when integrated into RSs?

• How about exploiting recent paradigms in RS to propose a novel and modern Arabic RS?

To answer those major questions, this thesis:

1) highlights the presence and the rapid increase of Arabic content on the web recently. It also describes the importance of exploiting such content.

2) presents the state-of-the-art on RS field. It explains different paradigms and concepts in this field, sheds light on the lack of researches and accessible resources in the Arabic RSs, and exposes the various limitations of the very few available works in this direction.

3) provides and makes accessible the four most large Arabic datasets to date, devoted to RSs exploration and investigation in the Arabic context. Each dataset consists of users reviews regarding a specific category of products in the Amazon e-platform. Each record in each dataset contains the review text in the Arabic language, the reviewer's score on a scale of 1 to 5, and other characteristics about the product/reviewer. The correctness of those datasets was implicitly proved through the set of experiments realized in this thesis.

4) realizes a comprehensive experimental study about applying the recent recommendation paradigms in the Arabic context. Several RSs have been proposed in the field of RSs. Such systems were devoted originally to English content and have never been explored on Arabic content. Therefore, this proposed study

explores applying those systems to Arabic content for clarifying and answering particular research questions. The results demonstrated the promising potential of the recent paradigms in the RS field when exploiting Arabic content in terms of accuracy and performance. This study may lead to immediate improvements in the Arabic RS field.

5) presents a modern RS for performing and improving rating prediction using Arabic content. The proposed system was built based on a combination of neural network technologies, namely CNN text processing techniques for modeling users and items from the text of reviews and a MLP network for predicting their interactions. Those techniques were never used before for performing those tasks in the Arabic context. The results of experiments conducted on our developed Arabic datasets showed the efficiency and effectiveness of this proposed combination in terms of prediction accuracy over different state-of-the-art baselines.

The above steps represent the entire work done in this research. Such as the previous studies, this work is neither perfect nor comprehensive. The Advancement in science will never cease. However, we expect this work to encourage the researcher community and serve as a starting point for further studies to improve the Arabic RS field. The Arabic RS field is still very little explored. Therefore, many areas of improvement exist for boosting this field. The following section presents the limitations of our work and different future directions and perspectives.

7.2 Limitations and Perspectives

This section presents several future research lines that can be explored to improve the new aspects presented in this thesis.

• Concerning the built datasets: in this work, we have used an MT tool to translate reviews' text from English into Arabic to create our datasets. Therefore, the obtained datasets represent the Arabic versions of existing English datasets in the Amazon repository. A possible next step is to build and make accessible Arabic language datasets by collecting reviews from the online Arabic stores. Such datasets will allow exploring RSs in a real Arabic context because they contain the textual reviews written by Arabic users.

• Concerning the exploratory study: In this study, we have evaluated the performance of all adopted RSs using only one evaluation metric designed to measure predictions' accuracy. However, it would be interesting to investigate their performance using other metrics to better understand those systems' behavior when used in the Arabic context. On the other hand, through the realized experiments, the adopted RSs have been explored in the Arabic context for a specific recommendation task, notably the rating prediction. However, it would be interesting in future work to investigate them on other recommendation problems such as the usage prediction or personalized ranking where the system's goal is not to predict missing ratings but to suggest items or find the best-ranked list for the target user, respectively.

• Concerning the proposed RS: It has to be noted that the proposed system has provided high-quality results for the rating prediction task compared to different state-of-the-art models. However, our proposal has been employed for a specific content language (Modern Standard Arabic). Future research will focus on extending our RS for managing other content languages such as Moroccan Arabic and Amazigh. Also, the proposed RS has exploited two deep learning approaches to perform text analysis and rating prediction tasks. In the future, we will extend the proposed system by adopting other DL-based architectures and studying their impacts on prediction quality. Another promising direction is to explore the proposed RS by considering the case of cold-start or data sparsity. In such settings, it will be essential to integrate into the proposal an appropriate mechanism that can mitigate the negative impact of such issues on the rating prediction task. By taking all the aspects above into consideration, it will be interesting to commercialize our proposal.

Bibliography

[A+16] Charu C Aggarwal et al. *Recommender systems*, volume 1. Springer, 2016.

[Aiz03] Akiko Aizawa. An information-theoretic perspective of tf–idf measures. *Information Processing & Management*, 39(1):45–65, 2003.

[AJAP+19] Gharbi Alshammari, Jose L Jorro-Aragoneses, Nikolaos Polatidis, Stelios Kapetanakis, Elias Pimenidis, and Miltos Petridis. A switching multi-level method for the long tail recommendation problem. *Journal of Intelligent & Fuzzy Systems*, 37(6):7189–7198, 2019.

[AKGA+14] Mohammed N Al-Kabi, Amal H Gigieh, Izzat M Alsmadi, Heider A Wahsheh, and Mohamad M Haidar. Opinion mining and analysis for arabic language. *IJACSA) International Journal of Advanced Computer Science and Applications*, 5(5):181–195, 2014.

[AT05] 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.

[ATK21] Amal Alharbi, Mounira Taileb, and Manal Kalkatawi. Deep learning in arabic sentiment analysis: An overview. *Journal of Information Science*, 47(1):129–140, 2021.

[AU19] Taushif Anwar and V Uma. Cd-spm: cross-domain book recommendation using sequential pattern mining and rule mining. *Journal of King Saud University-Computer and Information Sciences*, 2019.

[AZH+21] Laith Alzubaidi, Jinglan Zhang, Amjad J Humaidi, Ayad Al-Dujaili, Ye Duan, Omran Al-Shamma, J Santamaría, Mohammed A Fadhel, Muthana Al-Amidie, and Laith Farhan. Review of deep learning: Concepts, cnn architectures, challenges, applications, future directions. *Journal of big Data*, 8(1):1–74, 2021.

[BFZ14] Yang Bao, Hui Fang, and Jie Zhang. Topicmf: Simultaneously exploiting ratings and reviews for recommendation. In *Twenty-Eighth AAAI conference on artificial intelligence*, 2014.

[BKBH21] Marouane Birjali, Mohammed Kasri, and Abderrahim Beni-Hssane. A comprehensive survey on sentiment analysis: Approaches, challenges and trends. *Knowledge-Based Systems*, page 107134, 2021.

[BNJ03] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[BOHG13] Jesús Bobadilla, Fernando Ortega, Antonio Hernando, and Abraham Gutiérrez. Recommender systems survey. *Knowledge-based systems*, 46:109–132, 2013.

[BYBK19] 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.

[CBdG+17] Annalina Caputo, Pierpaolo Basile, Marco de Gemmis, Pasquale Lops, Giovanni Semeraro, and Gaetano Rossiello. Sabre: A sentiment aspect-based retrieval engine. In *Information Filtering and Retrieval*, pages 63–78. Springer, 2017.

[CC05] Hung-Chen Chen and Arbee LP Chen. A music recommendation system based on music and user grouping. *Journal of Intelligent Information Systems*, 24(2):113–132, 2005.

[CC17] Rose Catherine and William Cohen. Transnets: Learning to transform for recommendation. In *Proceedings of the eleventh ACM conference on recommender systems*, pages 288–296, 2017.

[CCW15] Li Chen, Guanliang Chen, and Feng Wang. Recommender systems based on user reviews: the state of the art. *User Modeling and User-Adapted Interaction*, 25(2):99–154, 2015.

[CCZ+19] Zhiyong Cheng, Xiaojun Chang, Lei Zhu, Rose C Kanjirathinkal, and Mohan Kankanhalli. Mmalfm: Explainable recommendation by leveraging reviews and images. *ACM Transactions on Information Systems (TOIS)*, 37(2):1–28, 2019.

[CDH+18] Zhiyong Cheng, Ying Ding, Xiangnan He, Lei Zhu, Xuemeng Song, and Mohan S Kankanhalli. Aˆ 3ncf: An adaptive aspect attention model for rating prediction. In *IJCAI*, pages 3748–3754, 2018.

[CDZK18] Zhiyong Cheng, Ying Ding, Lei Zhu, and Mohan Kankanhalli. Aspect-aware latent factor model: Rating prediction with ratings and reviews. In *Proceedings of the 2018 world wide web conference*, pages 639–648, 2018.

[CHC+18] Rui Chen, Qingyi Hua, Yan-Shuo Chang, Bo Wang, Lei Zhang, and Xiangjie Kong. A survey of collaborative filtering-based recommender systems: From traditional methods to hybrid methods based on social networks. *IEEE Access*, 6:64301–64320, 2018.

[CHN+17] Da Cao, Xiangnan He, Liqiang Nie, Xiaochi Wei, Xia Hu, Shunxiang Wu, and Tat-Seng Chua. Cross-platform app recommendation by jointly modeling ratings and texts. *ACM Transactions on Information Systems (TOIS)*, 35(4):1–27, 2017.

[CM06] Judith A Chevalier and Dina Mayzlin. The effect of word of mouth on sales: Online book reviews. *Journal of marketing research*, 43(3):345–354, 2006.

[CN21] James Chambua and Zhendong Niu. Review text based rating prediction approaches: preference knowledge learning, representation and utilization. *Artificial Intelligence Review*, 54(2):1171–1200, 2021.

[CZLM18] Chong Chen, Min Zhang, Yiqun Liu, and Shaoping Ma. Neural attentional rating regression with review-level explanations. In *Proceedings of the 2018 World Wide Web Conference*, pages 1583–1592, 2018.

[DDGR07] Abhinandan S Das, Mayur Datar, Ashutosh Garg, and Shyam Rajaram. Google news personalization: scalable online collaborative filtering. In *Proceedings of the 16th international conference on World Wide Web*, pages 271–280, 2007.

[DK11] Christian Desrosiers and George Karypis. A comprehensive survey of neighborhood-based recommendation methods. *Recommender systems handbook*, pages 107–144, 2011.

[DMD21] Parisa Abolfath Beygi Dezfouli, Saeedeh Momtazi, and Mehdi Dehghan. Deep neural review text interaction for recommendation systems. *Applied Soft Computing*, 100:106985, 2021.

[DQW+14] Qiming Diao, Minghui Qiu, Chao-Yuan Wu, Alexander J Smola, Jing Jiang, and Chong Wang. Jointly modeling aspects, ratings and sentiments for movie recommendation (jmars). In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 193–202, 2014.

[DQZ18] Bin Ding, Huimin Qian, and Jun Zhou. Activation functions and their characteristics in deep neural networks. In *2018 Chinese control and decision conference (CCDC)*, pages 1836–1841. IEEE, 2018.

[DSRO20a] Aminu Da'u, Naomie Salim, Idris Rabiu, and Akram Osman. Recommendation system exploiting aspect-based opinion mining with deep learning method. *Information Sciences*, 512:1279–1292, 2020.

[DSRO20b] Aminu Da'u, Naomie Salim, Idris Rabiu, and Akram Osman. Weighted aspect-based opinion mining using deep learning for recommender system. *Expert Systems with Applications*, 140:112871, 2020.

[EE16] Ashraf Elnagar and Omar Einea. Brad 1.0: Book reviews in arabic dataset. In *2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA)*, pages 1–8. IEEE, 2016.

[FEAD21] Mohammadsadegh Vahidi Farashah, Akbar Etebarian, Reza Azmi, and Reza Ebrahimzadeh Dastjerdi. A hybrid recommender system based-on link prediction for movie baskets analysis. *Journal of Big Data*, 8(1):1–24, 2021.

[GBC16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[GD98] Matt W Gardner and SR Dorling. Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. *Atmospheric environment*, 32(14-15):2627–2636, 1998.

[GGM+19] Diksha Garg, Priyanka Gupta, Pankaj Malhotra, Lovekesh Vig, and Gautam Shroff. Sequence and time aware neighborhood for session-based recommendations: Stan. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1069–1072, 2019.

[GWK+18] Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Gang Wang, Jianfei Cai, et al. Recent advances in convolutional neural networks. *Pattern Recognition*, 77:354–377, 2018.

[HASA20] Fouzi Harrag, Abdulmalik Salman Al-Salman, and Alaa Alquahtani. Arabic opinion mining using a hybrid recommender system approach. *arXiv preprint arXiv:2009.07397*, 2020.

[HB06] A Ben Hamza and David J Brady. Reconstruction of reflectance spectra using robust nonnegative matrix factorization. *IEEE Transactions on Signal Processing*, 54(9):3637–3642, 2006.

[HFT20] S Hasanzadeh, SM Fakhrahmad, and M Taheri. Based recommender systems: a proposed rating prediction scheme using word embedding representation of reviews. *The Computer Journal*, 2020.

[HKBR99] Jonathan L Herlocker, Joseph A Konstan, Al Borchers, and John Riedl. An algorithmic framework for performing collaborative filtering. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 230–237, 1999.

[HKTR04] Jonathan L Herlocker, Joseph A Konstan, Loren G Terveen, and John T Riedl. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1):5–53, 2004.

[HKV08] 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.

[HL04] Minqing Hu and Bing Liu. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177, 2004.

[HLZ+17] 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, 2017.

[Hof01] Thomas Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Machine learning*, 42(1):177–196, 2001.

[HRCB19] María Hernández-Rubio, Iván Cantador, and Alejandro Bellogín. A comparative analysis of recommender systems based on item aspect opinions extracted from user reviews. *User Modeling and User-Adapted Interaction*, 29(2):381–441, 2019.

[IFO15] Folasade Olubusola Isinkaye, YO Folajimi, and Bolande Adefowoke Ojokoh. Recommendation systems: Principles, methods and evaluation. *Egyptian informatics journal*, 16(3):261–273, 2015.

[JZFF10] Dietmar Jannach, Markus Zanker, Alexander Felfernig, and Gerhard Friedrich. *Recommender systems: an introduction*. Cambridge University Press, 2010.

[KAU16] Shah Khusro, Zafar Ali, and Irfan Ullah. Recommender systems: issues, challenges, and research opportunities. In *Information Science and Applications (ICISA) 2016*, pages 1179–1189. Springer, 2016.

[KBGM09] Georgia Koutrika, Benjamin Bercovitz, and Hector Garcia-Molina. Flexrecs: expressing and combining flexible recommendations. In *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data*, pages 745–758, 2009.

[KBV09] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[KMLD21] Georgia Koppe, Andreas Meyer-Lindenberg, and Daniel Durstewitz. Deep learning for small and big data in psychiatry. *Neuropsychopharmacology*, 46(1):176–190, 2021.

[KNSP21] Zahid Younas Khan, Zhendong Niu, Sulis Sandiwarno, and Rukundo Prince. Deep learning techniques for rating prediction: a survey of the state-of-the-art. *Artificial Intelligence Review*, 54(1):95–135, 2021.

[Kor08] Yehuda Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 426–434, 2008.

[KPO+16] 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.

[KPOY17] Donghyun Kim, Chanyoung Park, Jinoh Oh, and Hwanjo Yu. Deep hybrid recommender systems via exploiting document context and statistics of items. *Information Sciences*, 417:72–87, 2017.

[KS07] Young Ae Kim and Jaideep Srivastava. Impact of social influence in e-commerce decision making. In *Proceedings of the ninth international conference on Electronic commerce*, pages 293–302, 2007.

[LCC06] Cane WK Leung, Stephen CF Chan, and Fu-lai Chung. Integrating collaborative filtering and sentiment analysis: A rating inference approach. In *Proceedings of the ECAI 2006 workshop on recommender systems*, pages 62–66, 2006.

[LCT21] Alexander Ligthart, Cagatay Catal, and Bedir Tekinerdogan. Systematic reviews in sentiment analysis: a tertiary study. *Artificial Intelligence Review*, pages 1–57, 2021.

[LDC20] Hsiao-Chi Li, Zong-Yue Deng, and Hsin-Han Chiang. Lightweight and resource-constrained learning network for face recognition with performance optimization. *Sensors*, 20(21):6114, 2020.

[LHM+14] Haifeng Liu, Zheng Hu, Ahmad Mian, Hui Tian, and Xuzhen Zhu. A new user similarity model to improve the accuracy of collaborative filtering. *Knowledge-based systems*, 56:156–166, 2014.

[LLK14] Guang Ling, Michael R Lyu, and Irwin King. Ratings meet reviews, a combined approach to recommend. In *Proceedings of the 8th ACM Conference on Recommender systems*, pages 105–112, 2014.

[LLX+20] Xiaofeng Liao, Xiangjun Li, Qingyong Xu, Hu Wu, and Yongji Wang. Improving ant collaborative filtering on sparsity via dimension reduction. *Applied Sciences*, 10(20):7245, 2020.

[LMZ11] Yue Lu, Qiaozhu Mei, and ChengXiang Zhai. Investigating task performance of probabilistic topic models: an empirical study of plsa and lda. *Information Retrieval*, 14(2):178–203, 2011.

[LQP+19] Chenliang Li, Cong Quan, Li Peng, Yunwei Qi, Yuming Deng, and Libing Wu. A capsule network for recommendation and explaining what you like and dislike. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*, pages 275–284, 2019.

[LSY03] Greg Linden, Brent Smith, and Jeremy York. Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet computing*, 7(1):76–80, 2003.

[LWL+17] Weibo Liu, Zidong Wang, Xiaohui Liu, Nianyin Zeng, Yurong Liu, and Fuad E Alsaadi. A survey of deep neural network architectures and their applications. *Neurocomputing*, 234:11–26, 2017.

[LWM+15] Jie Lu, Dianshuang Wu, Mingsong Mao, Wei Wang, and Guangquan Zhang. Recommender system application developments: a survey. *Decision Support Systems*, 74:12–32, 2015.

[LWP+20] Hongtao Liu, Yian Wang, Qiyao Peng, Fangzhao Wu, Lin Gan, Lin Pan, and Pengfei Jiao. Hybrid neural recommendation with joint deep representation learning of ratings and reviews. *Neurocomputing*, 374:77–85, 2020.

[LZL+15] Xin Luo, MengChu Zhou, Shuai Li, Zhuhong You, Yunni Xia, and Qingsheng Zhu. A nonnegative latent factor model for large-scale sparse matrices in recommender systems via alternating direction method. *IEEE transactions on neural networks and learning systems*, 27(3):579–592, 2015.

[MCCD13] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[MCW17] Yue Ma, Guoqing Chen, and Qiang Wei. Finding users preferences from large-scale online reviews for personalized recommendation. *Electronic Commerce Research*, 17(1):3–29, 2017.

[MdGSL17] Cataldo Musto, Marco de Gemmis, Giovanni Semeraro, and Pasquale Lops. A multi-criteria recommender system exploiting aspect-based sentiment analysis of users' reviews. In *Proceedings of the eleventh ACM conference on recommender systems*, pages 321–325, 2017.

[Mit16] Nupur Mittal. *Data, learning and privacy in recommendation systems*. PhD thesis, Rennes 1, 2016.

[MJZ03] Bamshad Mobasher, Xin Jin, and Yanzan Zhou. Semantically enhanced collaborative filtering on the web. In *European Web Mining Forum*, pages 57–76. Springer, 2003.

[ML13] Julian McAuley and Jure Leskovec. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 165–172, 2013.

[MLF13] Claudiu Cristian Musat, Yizhong Liang, and Boi Faltings. Recommendation using textual opinions. In *Twenty-third international joint conference on artificial intelligence*, 2013.

[MR17] Rachana Mehta and Keyur Rana. A review on matrix factorization techniques in recommender systems. In *2017 2nd International Conference on Communication Systems, Computing and IT Applications (CSCITA)*, pages 269–274. IEEE, 2017.

[MS08] Andriy Mnih and Russ R Salakhutdinov. Probabilistic matrix factorization. In *Advances in neural information processing systems*, pages 1257–1264, 2008.

[NH10] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Icml*, 2010.

[OBLB18] Ahmed Oussous, Fatima-Zahra Benjelloun, Ayoub Ait Lahcen, and Samir Belfkih. Big data technologies: A survey. *Journal of King Saud University-Computer and Information Sciences*, 30(4):431–448, 2018.

[OBLB20] Ahmed Oussous, Fatima-Zahra Benjelloun, Ayoub Ait Lahcen, and Samir Belfkih. Asa: A framework for arabic sentiment analysis. *Journal of Information Science*, 46(4):544–559, 2020.

[OLB18] Ahmed Oussous, Ayoub Ait Lahcen, and Samir Belfkih. Improving sentiment analysis of moroccan tweets using ensemble learning. In *International Conference on Big Data, Cloud and Applications*, pages 91–104. Springer, 2018.

[PBD18] Iván Palomares, Fiona Browne, and Peadar Davis. Multi-view fuzzy information fusion in collaborative filtering recommender systems: Application to the urban resilience domain. *Data & Knowledge Engineering*, 113:64–80, 2018.

[PCH11] Pearl Pu, Li Chen, and Rong Hu. A user-centric evaluation framework for recommender systems. In *Proceedings of the fifth ACM conference on Recommender systems*, pages 157–164, 2011.

[PFT10] Damien Poirier, Françoise Fessant, and Isabelle Tellier. Reducing the cold-start problem in content recommendation through opinion classification. In *2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, volume 1, pages 204–207. IEEE, 2010.

[PMDC19] Dimitri Palaz, Mathew Magimai-Doss, and Ronan Collobert. End-to-end acoustic modeling using convolutional neural networks for hmm-based automatic speech recognition. *Speech Communication*, 108:15–32, 2019.

[PME19] Hashem Parvin, Parham Moradi, and Shahrokh Esmaeili. Tcfaco: Trust-aware collaborative filtering method based on ant colony optimization. *Expert Systems with Applications*, 118:152–168, 2019.

[PPK05] Manos Papagelis, Dimitris Plexousakis, and Themistoklis Kutsuras. Alleviating the sparsity problem of collaborative filtering using trust inferences. In *International conference on trust management*, pages 224–239. Springer, 2005.

[PSM14] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[Qub15] Rana Chamsi Abu Quba. *On enhancing recommender systems by utilizing general social networks combined with users goals and contextual awareness*. PhD thesis, Université Claude Bernard-Lyon I, 2015.

[Ren10] Steffen Rendle. Factorization machines. In *2010 IEEE International conference on data mining*, pages 995–1000. IEEE, 2010.

[RNBY99] Berthier Ribeiro-Neto and Ricardo Baeza-Yates. Modern information retrieval. *Addison-Wesley*, 4:107–109, 1999.

[RRS11] Francesco Ricci, Lior Rokach, and Bracha Shapira. Introduction to recommender systems handbook. In *Recommender systems handbook*, pages 1–35. Springer, 2011.

[RRS15] Francesco Ricci, Lior Rokach, and Bracha Shapira. Recommender systems: introduction and challenges. In *Recommender systems handbook*, pages 1–34. Springer, 2015.

[RS20] Masoumeh Riyahi and Mohammad Karim Sohrabi. Providing effective recommendations in discussion groups using a new hybrid recommender system based on implicit ratings and semantic similarity. *Electronic Commerce Research and Appli-*

cations, 40:100938, 2020.

[RSDO20] Idris Rabiu, Naomie Salim, Aminu Da'u, and Akram Osman. Recommender system based on temporal models: a systematic review. *Applied Sciences*, 10(7):2204, 2020.

[RSMVULPO11] Mohammed Rushdi-Saleh, M Teresa Martín-Valdivia, L Alfonso Ureña-López, and José M Perea-Ortega. Oca: Opinion corpus for arabic. *Journal of the American Society for Information Science and Technology*, 62(10):2045–2054, 2011.

[S+00] Seung H Sebastian et al. Algorithms for non-negative matrix factorization. 2000.

[SA10] Motaz K Saad and Wesam M Ashour. Arabic morphological tools for text mining. In *Corpora, 6th ArchEng International Symposiums, EEECS'10 the 6th International Symposium on Electrical and Electronics Engineering and Computer Science*, volume 18, 2010.

[SAAES18] Said A Salloum, Ahmad Qasim AlHamad, Mostafa Al-Emran, and Khaled Shaalan. A survey of arabic text mining. In *Intelligent Natural Language Processing: Trends and Applications*, pages 417–431. Springer, 2018.

[SCS20] Amit Kumar Sharma, Sandeep Chaurasia, and Devesh Kumar Srivastava. Sentimental short sentences classification by using cnn deep learning model with fine tuned word2vec. *Procedia Computer Science*, 167:1139–1147, 2020.

[SDS+19] Camila Vaccari Sundermann, Marcos Aurélio Domingues, Roberta Akemi Sinoara, Ricardo Marcondes Marcacini, and Solange Oliveira Rezende. Using opinion mining in context-aware recommender systems: A systematic review. *Informa-*

tion, 10(2):42, 2019.

[SHK$^+$14] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[SHYL17] Sungyong Seo, Jing Huang, Hao Yang, and Yan Liu. Interpretable convolutional neural networks with dual local and global attention for review rating prediction. In *Proceedings of the eleventh ACM conference on recommender systems*, pages 297–305, 2017.

[Sid18] Sumit Sidana. *Recommendation systems for online advertising*. PhD thesis, Université Grenoble Alpes, 2018.

[SKKR00] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Application of dimensionality reduction in recommender system-a case study. Technical report, Minnesota Univ Minneapolis Dept of Computer Science, 2000.

[SKR01] J Ben Schafer, Joseph A Konstan, and John Riedl. E-commerce recommendation applications. *Data mining and knowledge discovery*, 5(1):115–153, 2001.

[SLH14] Yue Shi, Martha Larson, and Alan Hanjalic. Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges. *ACM Computing Surveys (CSUR)*, 47(1):1–45, 2014.

[SM95] Upendra Shardanand and Pattie Maes. Social information filtering: Algorithms for automating “word of mouth”. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 210–217, 1995.

[SOALM20] Mehdi Srifi, Ahmed Oussous, Ayoub Ait Lahcen, and Salma Mouline. Recommender systems based on collaborative filtering using review texts—a survey. *Information*, 11(6):317, 2020.

[SOLM21] Mehdi Srifi, Ahmed Oussous, Ayoub Ait Lahcen, and Salma Mouline. Evaluation of recent advances in recommender systems on arabic content. *Journal of Big Data*, 8(1):1–19, 2021.

[SS17] H Sankar and V Subramaniyaswamy. Investigating sentiment analysis using machine learning approach. In *2017 International Conference on Intelligent Sustainable Systems (ICISS)*, pages 87–92. IEEE, 2017.

[SYCX18] Bo Song, Xin Yang, Yi Cao, and Congfu Xu. Neural collaborative ranking. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 1353–1362, 2018.

[SZL+19] Thiago Silveira, Min Zhang, Xiao Lin, Yiqun Liu, and Shaoping Ma. How good your recommender system is? a survey on evaluations in recommendation. *International Journal of Machine Learning and Cybernetics*, 10(5):813–831, 2019.

[SZYM19] Rong-Ping Shen, Heng-Ru Zhang, Hong Yu, and Fan Min. Sentiment based matrix factorization with reliability for recommendation. *Expert Systems with Applications*, 135:249–258, 2019.

[TH05] Kar Yan Tam and Shuk Ying Ho. Web personalization as a persuasion strategy: An elaboration likelihood model perspective. *Information systems research*, 16(3):271–291, 2005.

[TH+12] Tijmen Tieleman, Geoffrey Hinton, et al. Lecture 6.5-rmsprop:

Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31, 2012.

[TLH18] Yi Tay, Anh Tuan Luu, and Siu Cheung Hui. Multi-pointer co-attention networks for recommendation. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2309–2318, 2018.

[TSK+20] Amirsina Torfi, Rouzbeh A Shirvani, Yaser Keneshloo, Nader Tavaf, and Edward A Fox. Natural language processing advancements by deep learning: A survey. *arXiv preprint arXiv:2003.01200*, 2020.

[TZLM16] Yunzhi Tan, Min Zhang, Yiqun Liu, and Shaoping Ma. Rating-boosted latent topics: Understanding users and items with ratings and reviews. In *IJCAI*, volume 16, pages 2640–2646, 2016.

[VKVTL05] Ellen Van Kleef, Hans CM Van Trijp, and Pieternel Luning. Consumer research in the early stages of new product development: a critical review of methods and techniques. *Food quality and preference*, 16(3):181–201, 2005.

[WB11] Chong Wang and David M Blei. Collaborative topic modeling for recommending scientific articles. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 448–456, 2011.

[WBC07] Ryen W White, Mikhail Bilenko, and Silviu Cucerzan. Studying the use of popular destinations to enhance web search interaction. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 159–166, 2007.

[WGS+20] Wei Wang, Bing Guo, Yan Shen, Han Yang, Yaosen Chen, and Xinhua Suo. Twin labeled lda: a supervised topic model for document classification. *Applied Intelligence*, 50(12):4602–4615, 2020.

[WHC+17] 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.

[WQL+19] Libing Wu, Cong Quan, Chenliang Li, Qian Wang, Bolong Zheng, and Xiangyang Luo. A context-aware user-item representation learning for item recommendation. *ACM Transactions on Information Systems (TOIS)*, 37(2):1–29, 2019.

[WQLJ18] Libing Wu, Cong Quan, Chenliang Li, and Donghong Ji. Parl: Let strangers speak out what you like. In *Proceedings of the 27th ACM international conference on information and knowledge management*, pages 677–686, 2018.

[XCL+16] Zhenghua Xu, Cheng Chen, Thomas Lukasiewicz, Yishu Miao, and Xiangwu Meng. Tag-aware personalized recommendation using a deep-semantic similarity model with negative sampling. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 1921–1924, 2016.

[XLC+17] Zhenghua Xu, Thomas Lukasiewicz, Cheng Chen, Yishu Miao, and Xiangwu Meng. Tag-aware personalized recommendation using a hybrid deep model. AAAI Press/International Joint Conferences on Artificial Intelligence, 2017.

[YHPC18] Tom Young, Devamanyu Hazarika, Soujanya Poria, and Erik

Cambria. Recent trends in deep learning based natural language processing. *ieee Computational intelligenCe magazine*, 13(3):55–75, 2018.

[YST+04] Kai Yu, Anton Schwaighofer, Volker Tresp, Xiaowei Xu, and H-P Kriegel. Probabilistic memory-based collaborative filtering. *IEEE Transactions on Knowledge and Data Engineering*, 16(1):56–69, 2004.

[ZAS+17] Amel Ziani, Nabiha Azizi, Didier Schwab, Monther Aldwairi, Nassira Chekkai, Djamel Zenakhra, and Soraya Cheriguene. Recommender system through sentiment analysis. In *2nd international conference on automatic control, telecommunications and signals*, 2017.

[Zha19] Shuai Zhang. *Personalized recommendation: neural architectures and beyond.* PhD thesis, University of New South Wales, Sydney, Australia, 2019.

[ZLZ+14] Yongfeng Zhang, Guokun Lai, Min Zhang, Yi Zhang, Yiqun Liu, and Shaoping Ma. Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, pages 83–92, 2014.

[ZNY17] 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, 2017.

[ZW15] Ye Zhang and Byron Wallace. A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. *arXiv preprint arXiv:1510.03820*, 2015.

[ZWFM06] Sheng Zhang, Weihong Wang, James Ford, and Fillia Makedon. Learning from incomplete ratings using non-negative matrix factorization. In *Proceedings of the 2006 SIAM international conference on data mining*, pages 549–553. SIAM, 2006.

[ZYST19] 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.

Résumé

À l'ère du big data, les Systèmes de Recommandation (Recommender Systems, RSs) sont devenus des outils de plus en plus utilisés. Ils constituent un type important d'algorithmes d'apprentissage automatique (Machine Learning, ML) qui contribuent principalement à préserver la fidélité des internautes, en mettant à leur disposition un contenu personnalisé sur les différentes plateformes électroniques telles que Amazon, Netflix, YouTube et Facebook. Les RSs sont bénéfiques à la fois pour les utilisateurs et les entreprises. Ils aident les utilisateurs à prendre des décisions, et les entreprises à faire plus de bénéfices. En fait, une grande partie des revenus de nombreuses entreprises est générée uniquement par les recommandations.

Plusieurs RSs ont été proposés dans la littérature, dont la plupart se sont principalement concentrés sur le contenu anglais. La recherche et les ressources concernant les RSs en d'autres langues notamment l'arabe sont restreintes. Ces derniers temps, le contenu arabe sur le Web a considérablement augmenté, et ce en raison du nombre croissant d'internautes arabes; l'arabe classée quatrième parmi les dix premières langues utilisées sur internet. Ce qui sollicite la nécessité de conduire des études portant sur le contenu arabe, plus particulièrement dans le domaine des RSs.

Cette thèse aborde les récents travaux réalisés dans le domaine des RSs, tout en mettant l'accent sur le manque de recherches concernant les RSs arabes. Par ailleurs, elle présente d'autres nouvelles contributions visant à améliorer ce domaine peu exploré jusqu'à présent.

La première contribution porte sur l'exploration et l'investigation des RSs récents initialement consacrés au contenu anglais, lorsqu'ils sont appliqués au contenu arabe. Elle envisage d'expérimenter des RSs de pointe à partir de trois aspects à savoir, l'applicabilité, l'impact du prétraitement et la performance lors du changement de la langue du contenu. En réalisant cet ensemble d'expérimentations, l'apport de cette première contribution réside aussi dans la construction et la mise à disposition de quatre datasets arabes de taille permettant l'exploration des RSs dans un contexte arabe. Elle permet ainsi de combler le manque énorme de ressources arabes dédiées aux RSs.

La deuxième contribution est consacrée à l'enrichissement du domaine des RSs arabes en proposant un nouveau système de recommandation moderne, adapté au contenu arabe. La mise en œuvre du système proposé a été réalisé en combinant des modèles indépendants d'apprentissage en profondeur (Deep Learning, DL) en un seul système, à savoir des Réseaux de Neurones Convolutifs (Convolutional Neural Networks, CNNs) avec un Perceptron Multi-couche (Multi-Layer Perceptron, MLP). Ce système a pour principaux avantages, la capacité d'améliorer efficacement la précision des prédictions, et à traiter un grand volume de données.

Des expérimentations approfondies ont été menées en exploitant les datasets arabes construits dans cette thèse. Les résultats obtenus de notre exploration et de notre proposition garantissent des aboutissements prometteurs, pouvant inspirer la communauté des chercheurs à mener des études supplémentaires dans ce sens.

Mots-clefs (5) : Big data, Systèmes de Recommandation, Arabe, Prédiction de notes, Apprentissage profond.

Abstract

In the era of big data, Recommender Systems (RSs) have become growing essential tools. They represent important Machine Learning (ML) algorithms that keep users engaged with personalized content in different e-platforms like Amazon, Netflix, YouTube and Facebook. RSs are beneficial to both users and businesses. They help users make decisions, besides assisting companies to make more profits. A large chunk of many businesses' revenue is generated from recommendations alone.

Several RSs have been proposed in the literature, and most of them have primarily focused on English content. However, research and existing resources remain very limited for content in other languages like Arabic. Recently, the Arabic content on the Web has significantly increased because of the growing number of Arabic web users. Arabic came fourth in the top ten Internet languages. This highlights the need for exhaustive in-depth work on Arabic content, especially in the RS field.

This thesis describes the different works about RSs in general while emphasizing the lack of research on Arabic RSs. Furthermore, it presents several new contributions aiming to improve this field, little explored until now.

The first contribution explores and investigates recent RSs initially devoted to English content when applied to Arabic content. It plans to experiment with state-of-the-art RSs from three aspects: the applicability, the impact of the preprocessing, and the performance when changing the language of the content. By conducting this set of experiments, this contribution also makes four Arabic datasets available to explore RSs in an Arabic context. It thus allows addressing the considerable lack of Arabic resources devoted to RSs.

The second contribution is devoted to enriching the field of Arabic RSs by proposing a new modern recommendation system adapted to Arabic content. The implementation of the proposed system has been achieved by combining independent Deep Learning (DL) models into one system, namely Convolutional Neural Networks (CNNs) with a Multi-Layer Perceptron (MLP). The main advantages of this system are its capacity to effectively improve the accuracy of predictions and deal with a large volume of data.

Extensive experiments have been carried out exploiting the Arabic datasets constructed in this thesis. The results obtained from our exploration and our proposal ensure promising findings, inspiring the research community to carry out additional studies in this direction.

Keywords (5): Big data, Recommender Systems, Arabic, Rating prediction, Deep Learning.