
*School of Computer Science
The University of Auckland
New Zealand*

Personalized Risk Aware Recommender Systems

Danah Algawiaz

Supervisors:

Gillian Dobbie

Shafiq Alam

A THESIS SUBMITTED IN FULFILLMENT OF THE REQUIREMENTS OF
DOCTOR OF PHILOSOPHY IN COMPUTER SCIENCE, THE UNIVERSITY OF AUCK-
LAND, 2020.

Abstract

Recommender systems play a vital role in many sectors. Ease of use and availability increase the importance of these systems. E-commerce recommender systems provide recommendation lists based on user preferences and interest. The goal of e-commerce recommender systems from a user viewpoint is saving time and effort, and from a business viewpoint is increasing sales. Providing recommendations based only on user preferences can be risky because users' acceptance of recommendations may differ based on their situations. It is important to consider risks before providing recommendations to reduce the chance of irritating users with different items in the recommendation lists. Many studies have shown that the user intention to make a purchase can be employed to understand users' behaviour and provide different recommendations based on their intention. To decrease risks that may negatively affect system goals, assessing risks and controlling them are crucial steps. Risk assessment has two approaches: model driven and data driven. In this thesis, we consider the risk of the user not purchasing before providing a recommendation. The model driven approach is performed to obtain a better understanding behind the prediction. The disadvantage of the model driven approach is that the prediction does not consider a change in user behaviour. In order to address this shortcoming, we propose a data driven approach. Experimentally, we show

the proposed data driven model outperforms two state-of-the-art models on two publicly available data sets in regards to AUC of ROC area by 13% and 6%, respectively. Because different users can accept different levels of diversity in their recommendation lists, we also propose a risk assessment algorithm that predicts how many categories users can accept in the recommendation list based on their interaction with the system. Additionally, we propose a recommender system framework which includes the risk assessments of user purchase intention and user interaction with the system, the proposed framework can provide personalized recommendations with increased diversity in the recommendation lists. Experimentally, we show the resulting framework has higher coverage of items when compared with item-to-item collaborative filtering recommender systems and popularity-based recommender systems on two publicly available data sets, reaching 89.9% and 70.2%, respectively.

To my parents Haya & Abdullah

To my angels Rudina & Fayyadh

Acknowledgements

In the name of Allah, the Gracious, the Merciful. All praise belongs to Allah, Lord of all the worlds, who gave me the ability and patience to complete my PhD journey.

With great respect, I would like to express my sincere appreciation to my supervisor Prof. Gillian Dobbie for her support, insightful feedback and patience through the journey, she kept saying that her door is always open. To my co-supervisor Dr. Shafiq Alam for his support, advice and useful discussions through my PhD research.

Thanks to The Saudi Arabian Cultural Mission especially the former Cultural Attaché, Dr. Saud Altheyab, and current Cultural Attaché, Dr. Turki Almubrad, and my advisors for their support and advice during my PhD studying. Thanks to Mr. Mohsen Sawlan at the New Zealand Embassy in Riyadh, for his advice and information about studying in New Zealand. I would also like to thank Shaqra University for their financial support of my studying in New Zealand.

With my sincere appreciation, I dedicate this work to my mother for her enduring love and support during the hard time I had to go through to complete my PhD journey. To my father who supports my decisions even though studying in the farthest island around the world. To my kids, who chose to accompany

me on this tough journey. To my colleagues and friends at The University of Auckland and Shaqra University. I could not have completed this journey without your unconditional support and sweet words.

Thank you all from the bottom of my heart.

Contents

Abstract	i
Acknowledgements	iv
Contents	vi
List of Publications	x
List of Figures	xi
List of Tables	xiii
List of Algorithms	xiv
List of Symbols	xv
1 Introduction	1
1.1 Research Questions	6
1.2 Scope	7
1.3 Objectives	8
1.4 Contributions of the thesis	9
1.5 Structure of thesis	11

2	Background and Literature Survey	15
2.1	Recommender Systems	16
2.1.1	Similarity Metrics	18
	Pearson Correlation Metrics	18
	Cosine Metrics	19
2.1.2	Recommender Systems Approaches	20
	Content Based Approach	20
	Collaborative Filtering Approach	23
	Hybrid Approach	26
2.1.3	Limitations of the E-Commerce Recommender Systems	29
2.2	Role of Classifications in Building Recommendations	31
2.2.1	Issues of E-commerce Datasets	33
	Change of Users' Behaviours Over Time	34
	Imbalanced Datasets	35
2.2.2	Recommender Systems and Classifications	40
2.3	Dealing with Risks in Recommender Systems	41
2.3.1	Risk Management	42
2.3.2	Risk Analysis and Assessment	43
	Model Driven Approach	43
	Data Driven Approach	46
2.3.3	Risks in Recommender Systems	46
2.4	Conclusion	53
3	Risk Assessment based on a Model Driven Approach	55
3.1	Introduction	56
3.2	Contributions	58
3.3	Proposed Model	59
3.4	Experiments and Results	64
3.4.1	Experimental setup	64

3.4.2	Experimental results	66
3.5	Conclusion	72
4	Risk Assessment based on a Data Driven Approach	75
4.1	Introduction	76
4.2	Contributions	80
4.3	Proposed Model	80
4.4	Experiments and Results	84
4.4.1	Experimental setup	85
4.4.2	Experimental result	85
4.5	Conclusion	89
5	Risk Assessment of User Interaction based on a Data Driven Approach	92
5.1	Introduction	93
5.2	Contributions	95
5.3	Proposed Model	96
5.4	Experiments and Results	100
5.4.1	Experimental setup	101
5.4.2	Experimental result	102
5.5	Conclusion	108
6	Personalized Risk Aware Recommender System	111
6.1	Introduction	112
6.2	Contributions	114
6.3	Proposed Model	115
6.4	Experiments and Results	119
6.4.1	Experimental setup	120
6.4.2	Experimental result	122
6.5	Conclusion	130

7 Conclusions	133
7.1 Contributions	134
7.2 Limitations	135
7.3 Future Directions	135
Bibliography	139

List of Publications

Parts of Chapter 3 have been published in:

- Algawiaz, D., Dobbie, G., Alam, S. (2019). PRARS: Risk Assessment Model for Recommender Systems. In IEEE 18th International Conference on Intelligent Software Methodologies, Tools, and Techniques, pages 701-710.

Parts of Chapter 4 have been published in:

- Algawiaz, D., Dobbie, G., Alam, S. (2019). Predicting a User's Purchase Intention Using AdaBoost. In IEEE 14th International Conference on Intelligent Systems and Knowledge Engineering, pages 324-328.

Parts of Chapter 5 have been submitted to:

- Algawiaz, D., Dobbie, G., Alam, S. (2020). Risk Assessment to Predict Optimal Number of Item Category. In review at PAKDD 2021.

List of Figures

1.1	Recommender systems	4
1.2	Framework of proposed recommender system	4
1.3	Structure of thesis.	12
2.1	Recommender systems approaches and applications	21
2.2	Limitations of e-commerce recommender systems	32
2.3	Issues of e-commerce datasets	35
2.4	Recommender systems and risks	47
3.1	Assigning values of features	61
3.2	Sensitivity analysis of the features of the proposed model (Retail Rocket Dataset)	69
3.3	Sensitivity analysis of the features of the proposed model (Online Shoppers Dataset)	70
3.4	The ROC Curve of proposed model (RA-MD), MLP, and RNN	71
3.5	Evaluation of the proposed model(RA-MD), MLP, and RNN	71
3.6	User and population behaviours influence on the proposed model (RA-MD)	73
4.1	The change of behaviours over time for e-commerce datasets	79
4.2	Influence of recent sessions weight in F1 score	86

4.3	Influence of recent sessions threshold in F1 score	87
4.4	The ROC curve of the models	88
5.1	The change of behaviours over time for e-commerce datasets . . .	95
5.2	Influence of recent sessions weight in F1 score	103
5.3	Influence of recent sessions threshold in F1 score	104
5.4	Comparison of the proposed model and other recommender systems approaches	106
5.5	Comparison of the proposed model on different datasets	107
6.1	Personalized Risk Aware Recommender System sequence diagram	116
6.2	Precision of PRARS, Item-to-Item Collaborative Filtering and Popularity based recommender systems	123
6.3	Recall of PRARS, Item-to-Item Collaborative Filtering and Popularity based recommender systems	124
6.4	Coverage of PRARS, Item-to-Item Collaborative Filtering and Popularity based recommender system	126
6.5	long tail of items distribution of the proposed framework (PRARS), the Item-to-Item Collaborative Filtering recommender system and the Popularity based recommender system	127
6.6	Time to provide recommendation lists of the proposed framework (PRARS), the Item-to-Item Collaborative Filtering recommender system and the Popularity based recommender system	129

List of Tables

2.1	Personalization Risk Countermeasures	52
3.1	Risk features value of the Proposed Model Approach	62
3.2	Coefficient and the Exponent of the Proposed Model	62
3.3	Statistics of Datasets	65
4.1	Risk features types of RA-DD	83
5.1	Risk features types of RAI-DD	100
5.2	Statistics of Datasets	101
6.1	Statistics of Datasets	121

List of Algorithms

1	Risk Assessment based on Data Driven (RA-DD)	82
2	Risk Assessment of User Interaction (RAI-DD)	99
3	Diverse items List (DI)	117
4	Personalized Risk Aware Recommender System (PRARS)	120

List of Symbols

Symbol	Meaning
SN	Sensitivity of the validation set.
SP	Specificity of the validation set.
θ	Optimal threshold.
a	The coefficient of user classes.
bS	The exponent of user classes.
N	Number of items to be recommended.
I	List of all items.
DI	List of diverse items.
$s(u,i)$	Aggregation score for user u and item i .
$D(x)$	Weight of session x
Z_t	The sum of all weights by the classifier t
h_t	The predicted output by classifier t .
y_x	The correct output of session x .

1

Introduction

“Little strokes, fell great oaks.”

– Benjamin Franklin

Recommender systems play a vital role in many sectors such as e-commerce, medical, education and other sectors. The use of the web and modern lifestyles increase the importance of these systems due to convenience, because they can capture a user’s preferences explicitly or implicitly which saves users time, and availability because recommendations can be provided from anywhere, any time. Many retail stores have reduced their number of stores and moved to using e-commerce to provide and recommend their products and services. There

is no doubt that these systems are getting more attention, especially after the pandemic of the coronavirus. Recommender systems in e-commerce are basically providing items based on user preferences and interest. Recommender systems are used to provide the most relevant subset of items (Ricci et al., 2011). The benefit of recommender systems is providing users with a subset of items from a collection to save users time and effort which can increase conversion rate or sales (Lee and Hosanagar, 2016). There are many ways to capture users interests and this is how these systems have different approaches. Mainly there are three approaches of recommender systems to obtain user preferences: content based approach, collaborative filtering approach, and hybrid approach. The main idea behind content-based filtering is that if a user likes a certain item, the user will likely also prefer similar items. Figure 1.1, shows that the content-based filtering user will receive other colours of hexagons if she or he likes hexagon. However, for the collaborative filtering approach, if user A likes similar items that user B also likes (assuming they have similar preferences), then collaborate-filtering would assume that user A would like other items that user B prefers and vice versa. For instance, user B will receive other shapes based on the preferences of user A. The content based approach does not provide personalized recommendations while the collaborative filtering approach can provide personalized recommendations. A personalized recommendation means that the user may receive a recommendation for items that are not similar but are usually rated similarly or bought together. For example, recommending sugar when you buy coffee. The content based approach works as an information retrieval system, which recommends items to the user based on the items that the user already prefers and which have similar features (Jannach et al., 2010). For example, if the user prefers coffee then the recommender system will recommend coffee from other brands or coffee from the same brand with different density. It depends on the features of the items

and it compares the items based on these features. The second type of recommender system can provide personalization by building recommendations based on the users' collaboration with the system and does not rely on items features to provide recommendations. The collaborative filtering approach is one of the first and most common recommender systems that personalization can be provided through (Ricci et al., 2011). Some of the collaborative filtering recommender systems are based on items and others are based on users. The user based collaborative filtering approach finds the most similar users and provides their preferences to other users (Terveen and Hill, 2001). The second type is based on the items and recommends similar items that have been preferred by different users and this is called the item based collaborative filtering recommender system (Sarwar et al., 2001; Linden et al., 2003).

However, a pure personalization could be considered a risk because it may not be effective in all cases, some users preferring popularity based recommendations instead of personalization, especially because usually sparse data is used to build recommender systems (Zhang et al., 2013). Furthermore, these systems are lacking of diversity in recommendation lists (Said et al., 2013). The lack of diversity in recommendation lists is when the recommender system keeps recommending a subset of items based on the interaction with other items.

If a recommender system keeps recommending a subset of items to the user because they are more popular than other items this can affect the goal of the system from a business viewpoint which is increasing sales. From this point, there are risks of overwhelming or irritating users with many recommendations and a risk of just providing a subset of similar items through personalization. Several studies have focused on the importance of predicting the user's purchase intention or assessing the risk of the user not making a purchase for enhancing recommendations, services and personalization by understanding

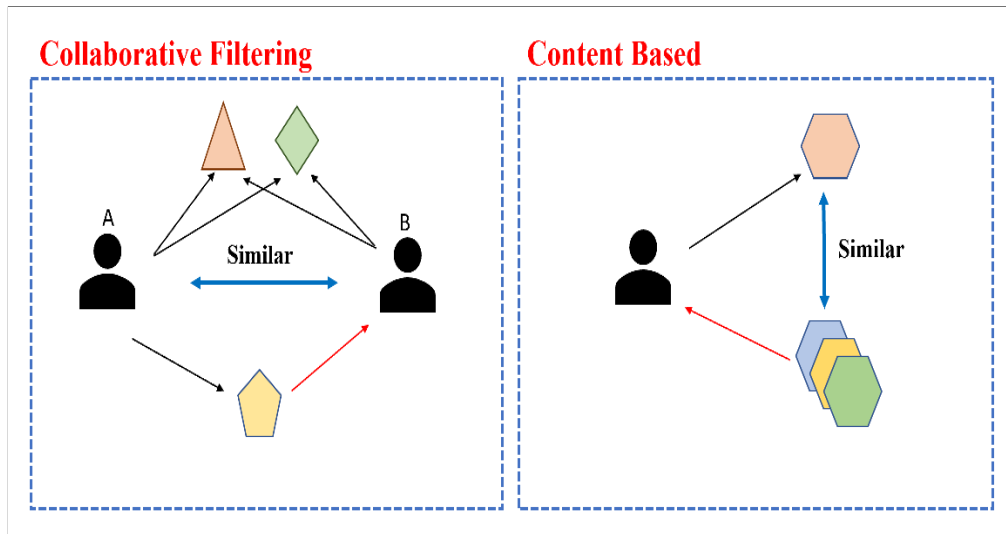


Figure 1.1: Recommender systems

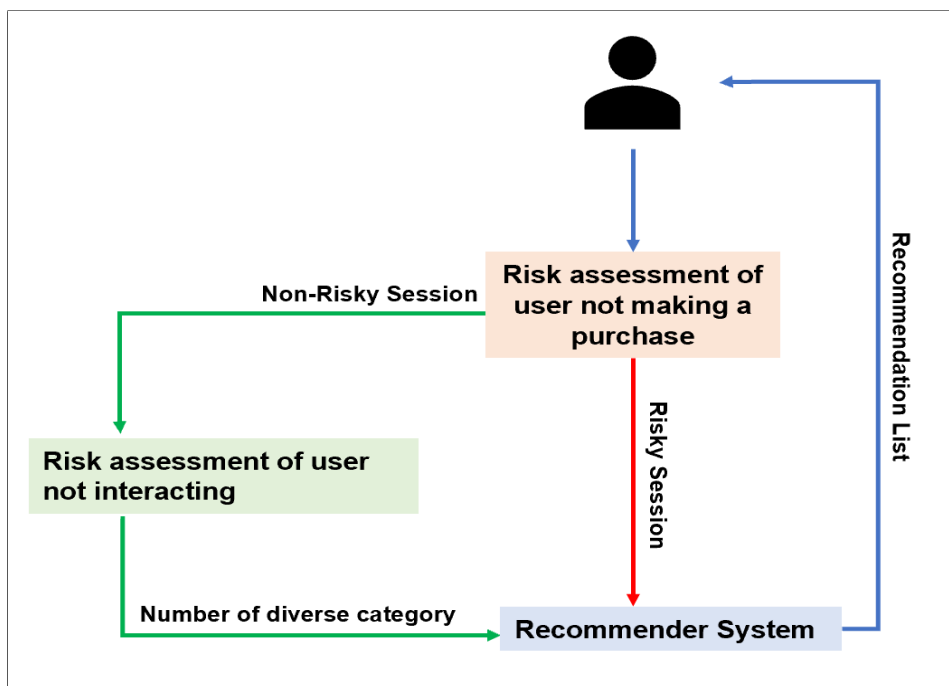


Figure 1.2: Framework of proposed recommender system

how users browse for specific items or how they search for related items. They concluded that different recommendations need to be based on a user's purchase intention (Morwitz and Schmittlein, 1992; Lo et al., 2016; Cheng et al., 2017). We apply two approaches for assessing users not purchasing: the model driven approach (RA-MD) and the data driven approach (RA-DD). The model driven approach is rule based while the data driven approach is built

through applying machine learning algorithms. **RA-MD** and **RA-DD** assess the risk of a user not making a purchase before providing the recommendation to provide personalized recommendations and reduce user irritation. The assessment is based on the users' behaviours which are implicit data.

There are many benefits of recommending different items to the user or increasing diversity in recommendation lists. Increasing diversity in recommendation list can be used to increase the chance of making purchases (Ziegler et al., 2005; Knijnenburg et al., 2012). Furthermore, it can be used to reduce issues of recommendation lists overfitting, or depending only on past preferences of users to build a recommendation list (Ziegler et al., 2005; Willemsen et al., 2011). However, another study shows that there is a risk of providing too many items in the recommendation list and too many diverse items can also affect a user's decisions negatively (Kapoor et al., 2015). We propose an algorithm to find how many different categories of items users may explore during sessions. The goal is to find the optimal number of different categories we can provide in the recommendation list to the user because each user may accept different levels of diversity in the recommendations list. Figure 1.2 presents a framework to control these risks which is a Personalized Risk Aware Recommender System (**PRARS**). We propose a framework for a recommender system that assesses risks, includes diverse items and recommends personalized items based on the assessment of two kinds of risks: risk of the user not purchasing (**RA-DD**), and risk of the user not interacting with the system (**RAI-DD**). Based on the assessments of these risks the proposed framework can include diverse items in the recommendation list or just provide personalized recommendations.

1.1 Research Questions

There are two issues that influence e-commerce recommender systems and affect the recommendation list negatively. They are bias towards a subset of items and lack of diversity in recommendation lists. The recommender systems try to be accurate by capturing the user's interests and preferences to make the recommendations more personalized. However, user behaviour may change over time and their acceptance of the recommendations depends on their situation. For instance, users who intend to make a purchase may explore more items than the users who do not want to make a purchase. The recommender systems may obtain benefits of assessing the behaviours that affect user acceptance before providing recommendation lists to users. Furthermore, the recommender system may have an issue with users preferences overfitting which as a result may affect the recommendation quality and tend to make the recommender system lose its personalization by providing a subset of items that is more popular. In this thesis we address these issues by investigating the role of risk assessment of e-commerce users' behaviours for providing personalized and diverse recommendations. The research questions that we are interested in are:

RQ1: How to build a risk assessment model based on different approaches to assess risks that affect the user decision to accept recommendations?

RQ2: What are the features that can be used to predict user behaviour for assessing risks and affect the user decision to accept recommendations?

RQ3: What are the challenges that can affect the assessment of these risks which can influence the user decision to accept recommendations and how to overcome them?

RQ4: How to employ the assessment of risks that can affect user decisions in

building a recommender system?

RQ5: How to build a recommender system that balances diversity and personalization based on users' behaviours?

1.2 Scope

The main goal of e-commerce recommender systems is to obtain user satisfaction by saving users time and effort searching for items and as a result increase sales. To the best of our knowledge there is no study that links user purchase intention, which is the factor that can help to achieve the goals of recommender systems, and build a recommendation list on different risk assessment approaches. In this thesis we analysis the risk of a user not purchasing. The goal is to reduce user irritation when the user intention to make a purchase is low. Furthermore, we assess risks of users not interacting with the system through user behaviour and their features to predict how many different categories of items the recommender systems can provide in the a recommendation list because different users may accept different levels of diversity. E-commerce datasets have some characteristics that may affect prediction and require to be addressed: imbalance issues and change of users' behaviours over time. We propose a recommender system framework that considers these risks before building a recommendation list to increase diversity that can reduce overfitting issues of user's preferences.

The proposed model assesses the risk of the user not making a purchase based on a model driven approach. However, it does not consider change over time of users' behaviours. Then, we propose a risk assessment model to adapt to change of users' behaviours over time based on a data driven approach. The proposed model classifies sessions as risky sessions and non-risky sessions based on an ensemble learning algorithm and considers the change of users'

behaviours over time by increasing the priority of the recent sessions.

Different users accept different levels of diversity, we assess the risk of the user not interacting with the systems. To find how many different categories we can provide to the user, we use a supervised learning algorithm for multiple class classification based on an ensemble learning algorithm and tackle the issue of the change over time of users' behaviours. These models require tuning more parameters than ensemble learning algorithms and does not consider the type of change. Finally, we propose a recommender system framework that considers the risk assessments before providing recommendations and introducing diverse recommendations. These diverse items are from the long tail distribution of items, without considering how diverse they are to a specific user.

1.3 Objectives

The main goal of this thesis is to consider risks based on users' behaviours before providing recommendations that may affect their decision to make a purchase to reduce effect of irritating users with diverse items. Furthermore, the e-commerce recommender systems face an issue which is lack of diversity due to over personalization. The objective is to assess risks then include diverse items in the recommendation lists if the user situation can accept these kinds of recommendations. To this end, the objectives of this thesis are:

- Develop two models that assess the risk of user not making a purchase based on a model driven approach and data driven approach.
- Specify the features that are used to in assessing the risk. These features are categorised to two behaviours which are user and population.
- Develop an algorithm that reflects the e-commerce purchasing pattern

for predicting a user's purchase intention and the number of diverse categories of items the recommender system can present in the recommendation list.

- Develop a recommender system framework that can recommend diverse and personalized items to users based on the models that are proposed to assess risks.

1.4 Contributions of the thesis

This research contributes to the body of work in recommender systems by proposing novel methods for recommending items that take the context of the user into account. The contributions of this thesis are as follows:

C1: Risk Assessment based on a Model Driven Approach (RA-MD),

a model that assesses the risk of a user not making a purchase by including two behaviours. This model assessed the risk of a user not purchasing by predicting and classifying sessions based on the concept of a model driven approach which is the Constructive Cost Model. We compare the model with state-of-the-art models that are used to predict and classify sessions based on the user's intention to make purchases. Experimentally, we show the effectiveness of RA-MD in assessing the risk of a user not purchasing and the AUC of the ROC curve is 89% and 79%, respectively, in two real world publicly available e-commerce datasets.

C2: Risk Assessment based on a Data Driven Approach (RA-DD),

an algorithm that predicts and assesses the risk of a user not purchasing using a data driven approach. We compare our algorithm with other machine learning algorithms to show the effectiveness of the model to handle issues of e-commerce data which are imbalanced datasets and data

that are changing over time. We compare the model with state-of-the-art models that are used to predict and classify session based on user's intention to make purchases. Experimentally, we show the effectiveness of RA-DD in assessing the risk of a user not purchasing and the AUC of the ROC curve is 91.5% and 90%, respectively, in two real world publicly available e-commerce datasets.

C3: Risk Assessment of User Interaction based on a Data Driven Approach (RAI-DD), a model that assesses the risk of a user not interacting with the system based on a data driven approach and to the best of our knowledge there is no study predicting how many diverse categories the recommender system can provide to users. This model can increase exploration of items, and reduce the risk of irritating users with many recommended items when the users' situations show that they may not interact with many diverse items. Experimentally, we show the effectiveness of RAI-DD in predicting the number of different categories the recommender system can present to users and the F1 score is 0.81 and 0.78, respectively, in two real world publicly available e-commerce datasets.

C4: Personalized Risk Aware Recommender System (PRARS), a recommender system framework that considers user behaviour by assessing risks of the user not purchasing and not interacting with the system. This recommender system may increase the diversity of the recommendation list based on the risk of a user not interacting with the system. To show the effectiveness of the proposed framework we compare it with other recommender system approaches and show how the proposed recommender system can decrease the long tail distribution issues of items by including items from different categories in the recommendation lists.

The proposed framework has the highest aggregate diversity and the coverage of PRARS is 89.9% and 70.2% , respectively, in two real world publicly available e-commerce datasets.

1.5 Structure of thesis

The aim of the thesis is to propose a framework that considers risk before providing recommendations to reduce user irritation when users are not intending to make a purchase by providing personalized recommendations and increasing diversity in the recommendations when their intention to make a purchase is high. Different users can accept different levels of diversity, assessing the risk of users not interacting can lead to predicting the level of diversity the recommender system can include in the recommendation lists. To assess risks two approaches have been performed: the model driven approach and the data driven approach. The model driven approach is a mathematical approach and the prediction can be explainable but rebuilding the model is required when there is a change of user behaviour. The data driven approach depends on the data and can adapt to changes of data. The proposed framework provides recommendations based on the assessment of risk. This thesis is structured as follows:

- *Chapter 2: Background and Literature Survey.*

We provide an overview of recommender systems and the role of risk in these systems. This chapter includes three parts. The first part provides a summary of the recommender systems approaches and the issues that these systems have. The second part provides reviews of the classification using supervised machine learning algorithms and the issue of e-commerce datasets that may affect learning in these kinds of the

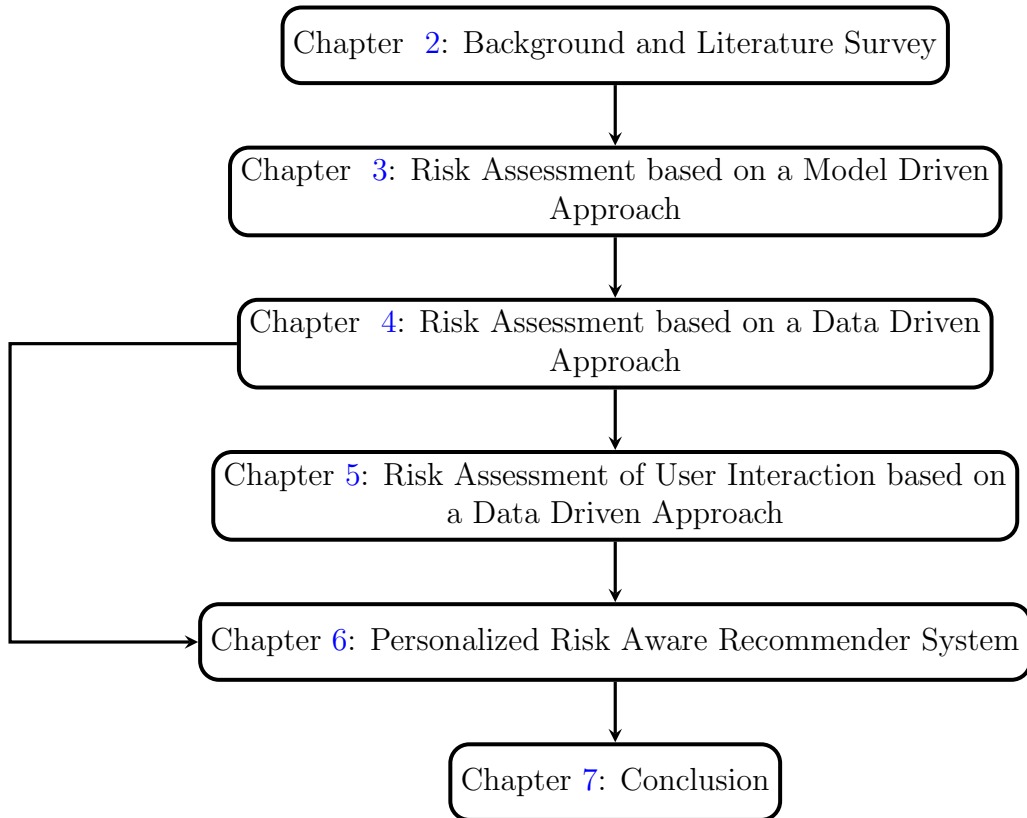


Figure 1.3: Structure of thesis.

datasets. The third part reviews risk management and the role of risk in recommender systems.

- *Chapter 3. Risk Assessment based on a Model Driven Approach.*
We propose a model that can be used to assess the risk of users not purchasing based on the Constructive Cost Model. The proposed model includes two behaviours to assess risk.
- *Chapter 4. Risk Assessment based on a Data Driven Approach.*
We introduce the issue that the model driven approach cannot handle and e-commerce datasets suffer from this issue which is changing of users' behaviours. We propose **RA-DD**, an algorithm that can handle the issue by increasing priority of the recent sessions.
- *Chapter 5: Risk Assessment of User Interaction based on a Data Driven Approach.*

We assess the risk of users not interacting with recommender systems by using an ensemble learning algorithm to find the optimal number of item categories users may explore during sessions. We propose **RAI-DD**, an algorithm that can apply multiclass classification algorithm to find how many different categories of items the recommender systems can provide in the recommendation lists, based on user interaction with the recommender systems.

- *Chapter 6: Personalized Risk Aware Recommender System.*

We propose **PRARS**, a recommender system framework that includes risks assessments before providing recommendations to the user. This framework can increase the diversity of items when the users' situations can accept this kind of recommendation which as a result decreases long tail distribution issues of items . This recommender system framework includes two types of risk: the user not making a purchase and the risk of recommending anarrow range of items.

- *Chapter 7: Conclusion.*

We present the contributions of the thesis, show the limitations, and provide future directions based on this research.

2

Background and Literature Survey

Recommender systems have been gaining more attention in recent years due to convenience, capturing user preferences explicitly or implicitly which saves users time, and availability, providing recommendations from anywhere, any time for users. Building these systems mainly depends on how to capture users' preferences to provide personalized recommendations. These systems aim to understand user behaviour to save users time and effort, attract them, and persuade them to make a purchase. However, a user's decision to make a purchase or interact with the system could be predicted and employed to make recommendations more suitable for the user. There are three main areas in this chapter which are recommender systems and related facets which are

classification and risk. To address RQ1 and RQ2 in Section 1.1, we present literature related to risks in recommender systems and different approaches for assessing risk. To address RQ3, we inspect the role of machine learning and the challenges of learning from e-commerce datasets. To address RQ4 and RQ5, we present background of different recommender systems approaches and the limitations of these systems, especially the lack of diversity in recommendation lists.

2.1 Recommender Systems

There has been a lot of research into developing efficient and accurate recommender systems. In the e-commerce domain, understanding customers and the impact of recommender systems on the user are important for building recommendations (Jiang et al., 2015). Recommender systems are used to provide the most relevant subset of items (Ricci et al., 2011). The benefit of recommender systems is providing users with a list of recommended items by recommending a subset of items from a collection to save users time and effort which can increase conversion rate and as a result increase sales (Ricci et al., 2011; Lee and Hosanagar, 2016). Popular e-commerce recommender systems are used by various sites such as Amazon, Aliexpress and Ebuy.

The recommendations are based on the data collected from the user. The goal of collecting the data is to know how users behave which can lead to predicting their future preferences (Sapsford and Jupp, 1996). There are two main categories of data that are collected from a user for building preferences: explicit data and implicit data (Sapsford and Jupp, 1996). Explicit data are collected after asking the user about their feedback explicitly such as their rating of items or asking them about their preferences explicitly. For example, whether the user likes an item or not. Implicit data are collected indirectly and without

asking the user such as observing items that a user views or buys or analysing the time spent viewing an item.

Each category has its advantages and disadvantages. The advantage of explicit data is that it can be interpreted easily which can be used directly to predict user preferences (Ricci et al., 2011). However, there are some disadvantages of explicit data to build recommendation lists. The first is the trustability of data provided by the user or whether the user has provided the right information (Ricci et al., 2011). The second is that it depends on the user's psychological situation, which means that the rating may differ based on the user's circumstance (Rafter, 2010).

Implicit data also has its positives and negatives. This category of data can be collected easily and does not require any effort from users because it depends on their interaction with the system (Parsons et al., 2004). Implicit data has a low bias compared with explicit data, because it depends on user behaviour (Rafter, 2010). However this kind of data requires more analysis and infers from interpreting the users interaction to build users preferences (Parsons et al., 2004). The second disadvantage is the privacy issue as it gathers users' data without their permission (Rahman and Ramos, 2013).

A combination between explicit and implicit data collecting can be used to understand user behaviour and predict preferences. The system is not relying only on explicit feedback from the user but can also use implicit feedback to predict preferences. Combining the two categories has its positives and negatives as well. The advantage is that the system is not relying only on the data provided by the user and it does not require any explicit feedback from the user to predict their preferences because it can rely on their implicit data, and as a result it does not depend on data that may have reliability issues (Ricci et al., 2011). However, it also has a negative influence on privacy, similar to the issue of implicit data collecting which has privacy and inference issues (Rahman

and Ramos, 2013).

2.1.1 Similarity Metrics

To build a recommender system, it is crucial to find a relationship between users and items to provide recommendation lists. The relationship is based on exploring the data to find similar users or similar items the user might prefer. There are many similarity measurements that can be used to build recommender systems such as Pearson, Cosine and Euclidean. Based on Bagchi and Saikat (2015), Cosine and Pearson perform better than other similarity metrics for collaborative filtering recommender systems. The Group Lens project introduced the first recommender system based on finding similarities between users by using the Pearson correlation metric and suggested it as a suitable method (Kansala, 1997). There are some studies suggesting applying the Pearson correlation metric for user based recommender systems and the Cosine metric for item based recommender systems (CarlKadie, 1998; Herlocker et al., 2002; Jannach et al., 2011).

Pearson Correlation Metrics

The Pearson correlation metric was firstly proposed by the Group Lens project. They presented a recommender system that uses the Pearson correlation metric to find similarities between users, and since then many recommender systems have been built based on this metric (Konstan et al., 1997).

A high positive value of Pearson correlation means high similarity between users, a low negative value of Pearson correlation means that users preferences are different and a zero value of Pearson correlation means that there is no relationship between users. The values of Pearson correlation are scaled between $[+1,-1]$. The equation of the Pearson correlation metric is defined as follows

for users' similarities:

$$\mathbf{sim}(\mathbf{a}, \mathbf{b}) = \frac{\sum_{i \in I_{ab}} (\mathbf{r}_{ai} - \bar{\mathbf{r}}_a)(\mathbf{r}_{bi} - \bar{\mathbf{r}}_b)}{\sqrt{\sum_{i \in I_{ab}} (\mathbf{r}_{ai} - \bar{\mathbf{r}}_a)^2} \sqrt{\sum_{i \in I_{ab}} (\mathbf{r}_{bi} - \bar{\mathbf{r}}_b)^2}} \quad (2.1)$$

where $sim(a,b)$ represents the similarity between user a and user b , I_{ab} represents the items set rated by users a and b , r_{ai} and r_{bi} represent the ratings of item i from users a and b , \bar{r}_a and \bar{r}_b represent the mean rating of all items from users a and b .

There is another alternative metric to the Pearson correlation metric, Spearman rank correlation, which uses the rank of the rating instead of the rating value. The Spearman rank correlation does not require rating normalization as the Pearson correlation metric does ([Ricci et al., 2011](#)).

Cosine Metrics

The Cosine similarity compares two users or two items by measuring the angle between them to find the similarity. The Cosine similarity is more suitable for item based collaborative filtering than the Pearson metric ([Jannach et al., 2011](#)). It has a similar scale to the Pearson correlation metric, and the scale is between $[+1,-1]$ where zero represents no similarity. The Cosine similarity between two vectors \mathbf{a} and \mathbf{b} is defined as:

$$\mathbf{cos}(\mathbf{a}, \mathbf{b}) = \frac{\mathbf{a} \cdot \mathbf{b}}{\|\mathbf{a}\| \cdot \|\mathbf{b}\|} \quad (2.2)$$

There is another alternative metric to the Cosine similarity, the Adjusted Cosine similarity. This function was proposed to deal with the fact that items are rated differently, and it deals with this issue by subtracting the user's average rating ([Segaran, 2007](#)).

The recommender system requires a similarity metric to measure the similar-

ties between items. Cosine similarity and Pearson correlation are well-known metrics to measure similarities between users and items. The Pearson correlation is applied to measure the similarities between items and users and it works well with recommendations that are based on explicit feedback such as the rating of items while the Cosine similarity works well when we measure the similarities between items based on the implicit feedback such as the purchase of items (Lee et al., 2007).

2.1.2 Recommender Systems Approaches

There are many approaches for building recommender systems. In this thesis we focus on the e-commerce recommender systems that build the recommendations based on collecting user's preferences and finding the similarities between items and users. There are three approaches of building recommender systems based on users' preferences. They differ in how they predict users' preferences and are presented in Figure 2.1. These three approaches are content based approach, collaborative filtering approach and hybrid approach.

Content Based Approach

Content based filtering is an approach to build a recommender system based on the data collected from the item description and user's preferences. The main idea of building a recommender system based on this approach is to find relationships between users' preferences and features of items. This approach is used when there is not enough information about user preferences. For instance, if the user prefers an item the recommender system tries to find similar items based on the description or features of other items and recommend them to the user. The main comparison here is between items and those which have similar features to the item that the user already prefers. It is building a search engine based on the interaction of the users and their preferences to find the

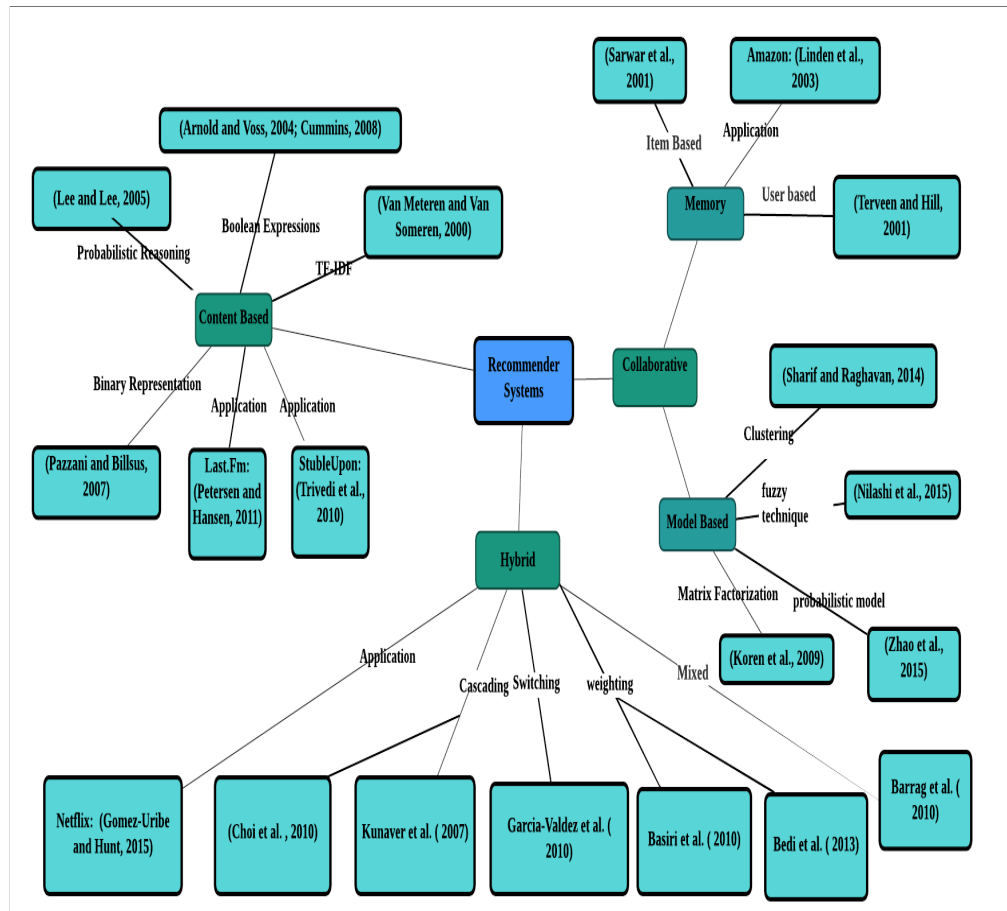


Figure 2.1: Recommender systems approaches and applications

most similar items that the system can provide or recommend to the users. Although, information retrieval is different than recommender systems, the content based recommender system has its root in information retrieval (Jan-nach et al., 2010). In a content based recommender system items and users' preferences are presented as vectors. The content based recommender system builds a vector for each user based on their preferences, and then compares similarities with other items to find the most similar items to the items that the user already prefers.

Term Frequency and Inverse Document Frequency (TF-IDF), which is used to calculate the importance of documents to users based on the terms that are used to describe the document. TF-IDF can be applied to build the content based filtering recommender system (Van Meteren and Van Someren, 2000).

TF refers to number of times the term is included in a document. IDF is used to reduce the effect of useless terms. Then the lengths of the document vectors are calculated as the square root of the sum of the squared weight of each term in the vector. A normalization step is done to normalize the vectors of the document by dividing the weight of the terms by length of the document. The final step is to calculate similarities between documents by using similarity metrics. Furthermore, a binary representation can be used to represent a vector of items and their features (Pazzani and Billsus, 2007). For instance, if the item has the feature it will be equal to one, otherwise it will be equal to zero. Building a content based recommender system based on binary representation has similar steps to TF-IDF. The steps are creating a vector for each item, creating a user profile, and creating a TF-IDF for each item based on the feature. The final stage is to calculate the sum product of these three vectors to find a score for each item and then provide the items that have highest scores to the user.

Some of the information retrieval techniques can also be employed on content based recommender systems such as combining terms with Boolean expressions (Arnold and Voss, 2004; Cummins, 2008). Another retrieval systems technique is using probabilistic reasoning instead of vector representation to find the probability of item relevancy to build recommender systems (Lee and Lee, 2005).

There are many successful applications that depend on the content based filtering approach that recommend documents to users based on their interest. For instance, StubleUpon is a content based recommender system that recommends web pages to users based on their web browsing (Trivedi et al., 2010). For the music domain, Last.Fm recommends songs to the user based on their preferences and song description (Petersen and Hansen, 2011).

The content-based filtering approach is more appropriate when there are a

detailed descriptions of items and the cold start issue for the items is rare because this recommender system is building the recommendation based on a description of the items (Watson et al., 2016). However, this system needs high volume interaction from the user to find the most relevant items based on their history interaction or it will have a cold start issue from the user side where the recommender system does not have enough information about the user profile and as a result it cannot recommend items based on user' preferences (Montaner et al., 2002). The second issue that this recommender system suffers from is that it can overfit user preferences because it only recommends similar items that the user already prefers, whereas preferences might change overtime (Abbassi et al., 2009).

Collaborative Filtering Approach

The collaborative filtering approach collects and analyzes data based on the users' actions or preferences and predicts what users might prefer based on other users who have similar preferences. This prediction is based on both implicit or explicit data (Ricci et al., 2011). For example, if two users purchased similar items, they might prefer similar items (implicit data) or if they have the same ranking of their favourite items they may prefer similar items (explicit data). The main advantage of this method is that it does not rely on item description and its recommendations are considered as accurate if the similarities between users are very high. However, it faces various issues such as the cold start problem and data sparsity (Lee et al., 2004). This recommender system provides recommendations based on the interaction between users and items to find similarities between users or items, and then suggests items based on users or items k-nearest neighbors algorithm. Many metrics have been used to measure similarities between users or items in collaborative filtering recommender systems such as Cosine and Pearson metrics.

Two of the earliest implementations of the collaborative filtering approach based on users similarities was proposed for music recommendations and showed its effectiveness to overcome weakness of content based recommender systems which lack serendipity (Shardanand, 1994). The second one is the GroupLens project which uses the Pearson metric to find similarities between users (Nguyen and Haddawy, 1998).

There are two main strategies for collaborative filtering recommender systems: model based and memory based. The memory based approach was first proposed to overcome the weakness of the content based approach and it has two categories: user based and item based recommender systems. The model based recommender system was proposed to improve the memory based approach by applying a machine learning algorithm for increasing accuracy of prediction or decreasing the complexity of the recommender system.

Memory Based Approach

The collaborative filtering memory based approach was first introduced to overcome drawbacks of content based recommender systems which are dependent on item descriptions and lack serendipity. The memory based approach of collaborative filtering is basically finding similarities between users based on the assumption that if user x prefers items $a, b,$ and c and user y prefers items a and b there is a high chance that user y will prefer item c . There are two kinds of memory based approaches: user similarities and item similarities. The user similarities build the user-to-user matrix to find similar users (Terveen and Hill, 2001). It is the first kind for building a collaborative filtering recommender system. The main advantage of this approach is that it does not rely on item descriptions, however, it has a scalability issue because every time a new user is added to the dataset the recommender system needs to rebuild the user-to-user matrix to include the preferences of the new user which also

results in data sparsity ([Ricci et al., 2011](#)).

The item based recommender systems build the item-to-item matrix to find similar items that have been preferred by different users ([Sarwar et al., 2001](#)). It is based on the assumption that if items a and b have been preferred by different users this means that item a is similar to item b and the user who prefers items a will most likely prefer item b . The main advantages of this kind of collaborative filtering approach is that it reduces the dimensionality of the user-to-user collaborative filtering approach, and it does not require any changes on matrix dimensions when there is a new user which is supposed to be more often than new items. However, in some recommender systems such as news recommendations the changes of the items are faster than the changes of the users, and the user-to-user based is more appropriate than item-to-item based. The choice basically depends on the domain of the recommender system and which one has more rate of change than the other.

The main application of this approach is the Amazon recommender systems that depend on item to item similarities to overcome scalability issues of user-to-user similarities ([Linden et al., 2003](#)).

In general the main advantages of the memory based collaborative filtering approach is that it can provide explanations about why it recommends items of both kind and it does not rely on item descriptions to recommend it to users . The main disadvantage of this approach is that it requires rebuilding every time there is a new user or new items which causes scalability issues. The second issue is that it requires the whole matrix to be in the memory to measure the similarities every time the recommendation is performed.

Model Based Approach

The model based approach is the second kind of collaborative filtering approach that uses machine learning algorithms to provide recommendations.

The main goal of the model based recommender system is to reduce the issue of data sparsity and increase the accuracy of the recommender system.

There are many machine learning approaches that have been used in recommender systems such as clustering, fuzzy, probabilistic model and matrix operations. The model based collaborative filtering recommender system does not require loading the similarities matrix in the memory to provide recommendations. The first recommender system that was proposed based on this approach was the Netflix competition and it is based on the matrix factorization techniques that improve root mean square error (Koren et al., 2009). The probabilistic model which is the Latent Dirichlet Allocation matrix technique has been used to predict the user's rating of an item (Zhao et al., 2015). Reducing the dimensionality is used to overcome data sparsity of the collaborative filtering recommender system by using a fuzzy technique (Nilashi et al., 2015). A clustering approach has been used to improve accuracy and decrease the data sparsity of recommender systems by clustering data points to identify them before processing (Sharif and Raghavan, 2014).

Although model based recommender systems have been used to reduce the issue of data sparsity or to increase accuracy of recommender systems, they suffer from the critical issue of the essentiality to retrain the model when there is a change on the user or items matrix which causes the need to rebuild the model.

Hybrid Approach

The hybrid approach is a combination of collaborative and content based recommender systems, and could be used to overcome weaknesses of the previous methods by providing an accurate recommendation (Adomavicius and Tuzhilin, 2005). For example, Netflix uses a hybrid recommender system to make recommendations by providing preferences of similar users (collaborative

filtering) as well as by suggesting movies that have similar characteristics to the movies that the user has liked and are highly rated by the user (content based filtering) (Gomez-Uribe and Hunt, 2015). The main advantage of this method is the ability to overcome the cold start problem of item and user. However, this method suffers from coping with frequent changes of user preferences (Zhang et al., 2013).

The hybrid recommender systems are used to overcome a weakness of a specific recommender system, reduce issues of recommender systems in general such as the cold start problem or increase the accuracy of the system. Hybrid recommender systems can be based on one recommender system approach or many recommender systems approaches. There are many strategies to apply a hybrid recommender system such as cascading, switching, weighting, features, and mixed.

The cascading strategy is an order sensitive technique that employs two recommender systems to provide recommendation lists. The first recommender system generates a recommendation list and the second recommender system filters the list. Kunaver et al. (2007), apply two collaborative filtering recommender systems with different similarity metrics based on the user rating. Choi et al. (2010), apply two different recommender systems, collaborative filtering recommender systems and content based recommender systems, to eliminate the scalability issue in the content based recommender system by applying the collaborative filtering approach to find the nearest neighbour of the user and ignores the rest of the users, then feeds this information to the content based recommender systems to produce the recommendation lists.

The switching strategy is a rule-based technique which employees two recommender systems for providing recommendation lists and it is not an order sensitive technique. Garcia-Valdez et al. (2010), apply two collaborative filtering recommender systems to solve the cold start issue of a new item. They

apply a short term collaborative filtering approach for the recent preferences of users, but if there is a new item they switch to the long term collaborative filtering to obtain the preferences in general.

The weighting strategy is a technique that aggregates two recommender systems and provides recommendation lists after computing the score of each recommender system. Basiri et al. (2010), apply the weighted linear function for collaborative filtering recommender system scores and content based recommender system scores, then change the weight based on user feedback.

The features strategy is a technique that the output of one recommender system considers as a feature of the second recommender system and it is an order sensitive technique. Bedi et al. (2013), use the collaborative filtering approach to obtain the preferences of the user, then apply clustering to find the categories of the books based on the preferences of the user and the type of users, this information is added to the content based recommender system to recommend the most relevant books to the user.

The mixed strategy is a technique that combines the recommendation lists from different recommender systems approaches and it is the simplest form of hybrid recommender system. Barrag et al. (2010), apply two collaborative filtering approaches, item based and user based, the item based to find the items that the user prefers and the user based to find the nearest neighbour of the user, then they obtain the recommendations from both recommender systems and those which are recommended by both systems are called star recommendations.

Hybrid recommender systems have been used to solve issues of the single recommender system such as scalability and cold start. However, it can increase the complexity of the recommender systems and reduce the ability of some recommender systems to be interpreted.

2.1.3 Limitations of the E-Commerce Recommender Systems

There are many issues of e-commerce recommender systems. Some of them are related to the data that is collected from the user, while others are related to the mechanism of the recommender systems' algorithms such as low accuracy, lack of diversity and novelty as shown in Figure 2.2 .

There are some issues that are related to data that are collected from the user such as scalability, data sparsity and cold start. The scalability issue is related to the number of users and items. Data sparsity is an issue of the low number of items ratings. The cold start issue is when there are new users or new items and the recommender system cannot obtain the preferences of the users or cannot recommend items due to low interactions of item or user. Some research applied hybrid recommender systems of collaborative filtering and content based to overcome these issues. They obtain the benefit of content based to solve cold start issues of collaborative filtering (Sanchez et al., 2012; Chughtai et al., 2014). Choi et al. (2010), improved the scalability issue and data sparsity by applying hybrid recommender systems of collaborative filtering and content based filtering. They compressed data by finding the nearest neighbour of each user and the furthest neighbour.

There are some issues that are related to the mechanism of the recommender systems algorithms. The accuracy of the recommender systems has been addressed in many studies (Deng et al., 2010; De Campos et al., 2010; Wen et al., 2012). Accuracy can be improved based on the data that are collected from the user (Deng et al., 2010). Others suggest to increase accuracy by hybridize two recommender systems' algorithms to obtain benefits from both (De Campos et al., 2010). However, evaluating the recommender systems based on accuracy only may have several limitations. Many studies have focused on how to the increase the accuracy of the model and this is mainly done by trying to capture

the similarities between items. This can increase bias toward a subset of items, popular items, and as a result this can affect the diversity of the e-commerce recommender systems. A study shows that the bias towards a subset of items in different recommender systems such as collaborative filtering recommender systems is increasing over time and as a result aggregate diversity is decreasing overtime in recommendation lists (Masoud et al., 2020). Diversity in recommendation can enhance consumer's desire to purchase and increase purchase rates (Hao et al., 2020; Kwon et al., 2020).

Some research projects tackle the lack of diversity in collaborative filtering recommender systems. One study proposed a K-Furthest Neighbour Collaborative Filtering Recommender Algorithm to decrease bias toward the popularity of items by recommending the items based on dissimilar users and all the users will receive recommendations from the least similar users to improve the diversity of the model (Said et al., 2013). This recommender influences the personalization that the collaborative filtering recommender systems are based on which is finding the most similar users to build the recommendations. Another study proposed building hybrid recommender systems which build clusters of users and items and the clusters that are similar to other clusters are deleted, then the recommender system builds the recommendation to choose the best cluster (Li and Murata, 2012). The Youtube recommender system uses association rule mining by adding some rules to the recommender system which is removing the videos that are too similar to the videos that are going to be included in the recommendation list and limiting the number of videos from the same channel (Davidson et al., 2010). Furthermore, collaborative filtering recommender systems may lack novelty. Novelty in recommendations is important when there are a few items which are extremely popular and the rest are much less well-known (Anderson, 2006). One study identified some users as experts of the population, and sent their recommendations to

other users in the same population, the population being identified based on the users' ratings, the goal being to decrease bias of the popularity and increase the novelty in the recommendation list (Lee and Lee, 2014). Uber Eats faced the same issue with their recommender system which is biased towards known restaurants, and they applied a multi-armed bandits method by setting the upper confidence bound high when the restaurant was new which gives it more chance to be recommended to users and this upper confidence bound would reach its actual value in time (Zhang et al., 2018). A study shows how to increase diversity in a recommendation list by applying a post-processing method after producing a longer recommendation list, identify items that have low visibility, then apply bipartite graph between items and users to generate shorter recommendation lists (Mansoury et al., 2020). Furthermore, balancing the training samples between the popular items and unobserved items could be applied to increase diversity in the recommendation lists (Boratto et al., 2021).

To the best of our knowledge all the suggestions of increasing the diversity in recommender systems do not consider user behaviour. Kapoor et al. (2015), show that there is a risk of providing too many items in the recommendation list and too many diverse items can also affect the user's decisions negatively. So, it is important to understand the user's behaviour before providing diverse items in the recommendation list.

2.2 Role of Classifications in Building Recommendations

Classifications have been employed before and through building recommendations. There are three main types of supervised machine learning approaches

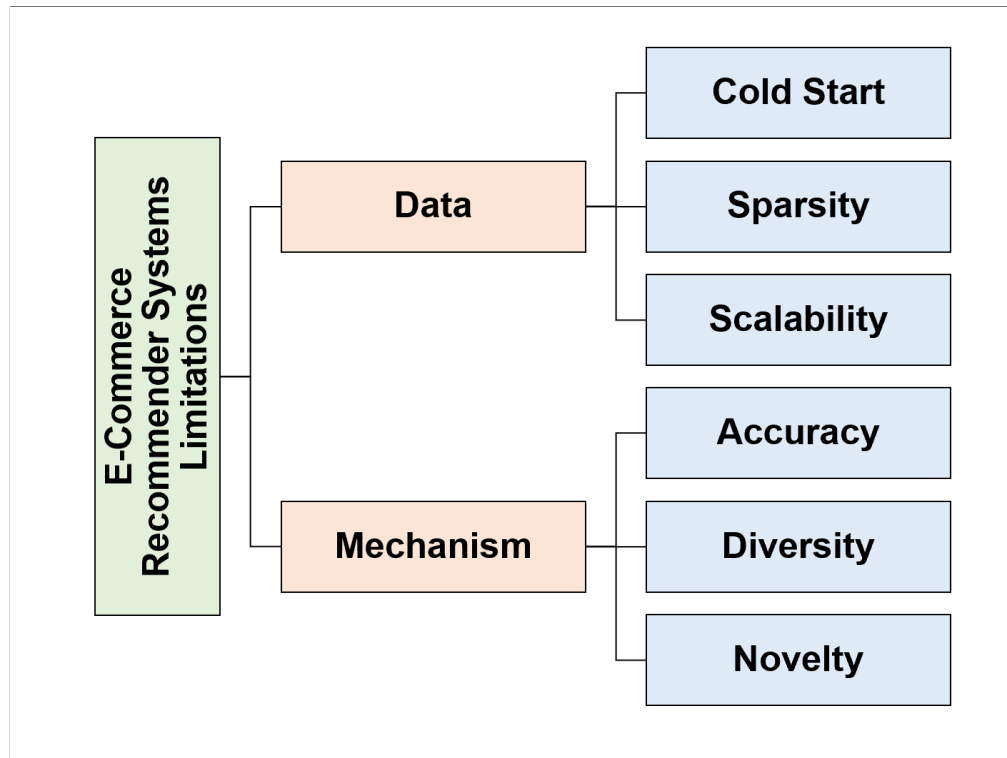


Figure 2.2: Limitations of e-commerce recommender systems

for classification: classical machine learning, deep learning, and ensemble learning. The classical machine learning algorithms, such as decision trees and the k-nearest neighbors algorithm, use a single base learner that is trained by training sets, and then test the model on unseen data. Ensemble learning uses multiple base learners to obtain higher accuracy than a single base learner. Deep learning is a subset of machine learning algorithms that uses multiple layers of neural networks as a learner to predict on unseen data. Based on the literature they all have their advantages and disadvantages. In summary, the classical approach can provide interpretation but it cannot handle the imbalance issue or changes of behaviour over time (He and Garcia, 2009). It is suitable for static data and balanced data sets. Deep learning can be more accurate than other machine learning algorithms but cannot handle changes of behaviour over time and the imbalance issue, and it is difficult to interpret (Nicolas, 2015). Furthermore, it also needs a huge volume of data and time

for training purposes. Ensemble learning can be interpreted (Friedman et al., 2000), but it is not as easy to interpret as some of the classical machine learning algorithms because it is based on many base learners. Furthermore, some of the ensemble learning can handle the imbalance issue such as the boosting algorithm.

Ensemble learning uses many base learners to build the model. Some of the base learners of the ensemble learning algorithm learn in parallel and others learn sequentially. Ensemble learning algorithms have three different approaches: bagging, boosting and stacking. The bagging algorithm is basically training different base learners in parallel on different instances of training sets. The instances are chosen randomly from the training data sets then a vote is used to choose the more suitable base learner (Breiman, 1996). The boosting algorithm applies the same concept as bagging but it learns sequentially and, instead of choosing the instances from the training sets randomly, the algorithms choose them based on their difficulty to classify, so those that are mis-classified will have higher priority and will be more likely to be chosen in the next iteration of the training process (Friedman et al., 2000). The stacking algorithm uses the different base learners as input then aggregates the output of the base learners to make new learners based on those base learners (Zhou, 2012).

2.2.1 Issues of E-commerce Datasets

Employing classifications for building recommendations mainly depends on datasets. The e-commerce datasets have different characteristics than other datasets. This is because e-commerce datasets are mainly based on users' behaviours and these behaviours are affected by outsource factors and contextual factors of users. Furthermore, e-commerce datasets can be imbalanced. Figure 2.3 shows the issues of e-commerce datasets and illustrates how to handle

them. In this section, we introduce these issues and describe how to handle them.

Change of Users' Behaviours Over Time

Machine learning algorithms depend on data to learn and any changes in data will affect their performance. Several studies have shown that for e-commerce datasets, the recent instances of the datasets are more important than previous instances (Whittington, 2013; Zhao and Cen, 2013). There are mainly three strategies to handle the change of behaviour over time or concept drift: static handling, instances weight, and dynamic handling.

Static handling is performed by manually retraining the model. The machine learning algorithms retrain the dataset after a period of time to include the new instances in the training process (Klinkenberg and Joachims, 2000; Gama et al., 2004; Zeira et al., 2004). However, this method has a risk of including two different behaviours at the same time of the training process which can affect the learning algorithm. The second strategy which is instances weight which is based on training the instances based on their weight such as the ones that ensemble learnings algorithms do (Widmer and Kubat, 1996; Klinkenberg and Renz, 1998). However, this method has an issue of finding the optimal weight of the instances which increase the tuning process of learning algorithms. The third strategy, dynamic handling, is performed by firstly detecting the change and then retraining the model based on the change of the datasets (Hulten et al., 2001; Bifet and Gavalda, 2006). The change of the new instances can be captured through the error rate of the test sets or when the performance of the model is decreased (Hulten et al., 2001; Bifet and Gavalda, 2006). However, it can increase the computation cost by adding a detective method and cannot handle the retraining change or seasonal change which is expected on the e-commerce datasets.

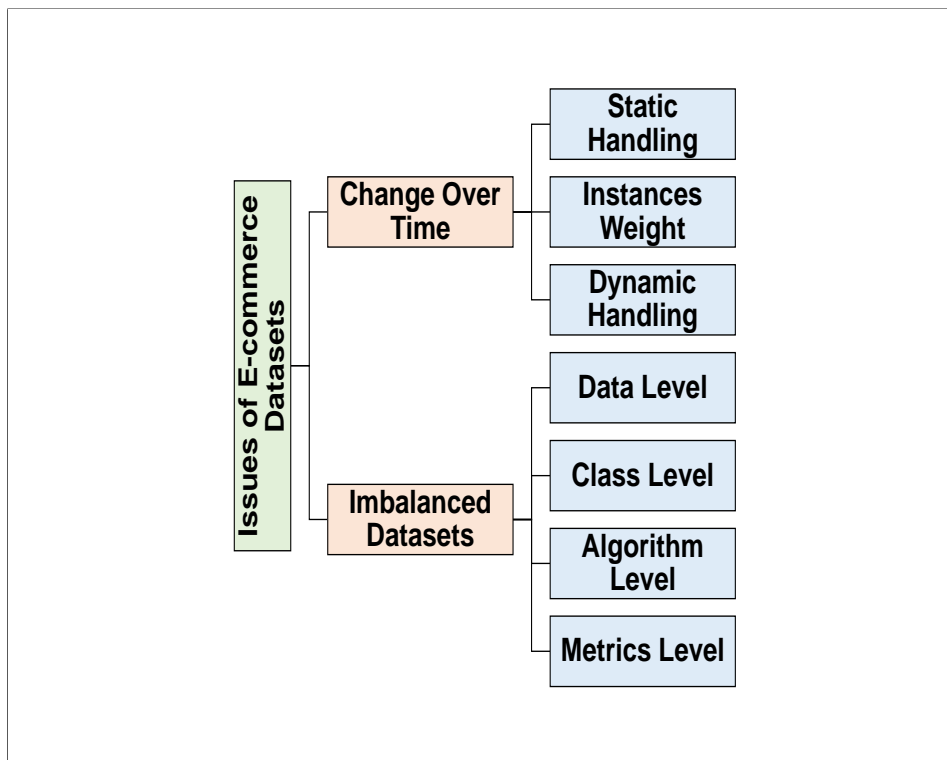


Figure 2.3: Issues of e-commerce datasets

Imbalanced Datasets

The imbalanced dataset issue is the situation where the classes of instances are not balanced. For instance, the number of the sessions that end with purchases in the e-commerce context are very low compared with the number of sessions that end with no purchase. Using classical machine learning algorithms or deep learning algorithms to classify sessions that contain purchase or not can cause bias toward the majority class which in this case is the session that does not contain purchase. The bias towards the majority class causes overfitting issues of the majority class or failure to generalise the model to classify the new instances because the learning algorithm considers the minority class a noise and discards them in the training process and as a result these machine learning algorithms misclassify the instance in the testing process. There are

four methods to deal with imbalance issues: data level, class level, algorithm level, and choosing the appropriate metrics to measure the performance of the model. In the next section we will describe each method, and show which one is more suitable for e-commerce datasets.

Data Level

The data level method for handling the imbalanced issue is through resampling the instances of the training dataset or balancing the number of instances in each class to prevent the learning algorithm from considering minority instances noise. There are two main techniques for resampling the instances: under sampling the majority and over sampling the minority. The goal is to make a balanced training set which reduces the overfitting issue.

Under sampling the training set by reducing the number of the instances from the majority class and using all of the minority instances of the training set for learning. However, the quantity of instances should be enough to decrease the effect of reducing the number of instances. Over sampling the minority instances by duplicating the instances to reduce the issue of discarding them in the training process.

The main techniques of this method are random oversampling or under-sampling, cluster based resampling and Synthetic Minority Over-sampling. Random based oversampling or under sampling is performed by randomly eliminating instances from the majority class or duplicating instances from the minority class ([Estabrooks et al., 2004](#); [Drummond et al., 2003](#)). This technique can suffer from overfitting issues if the oversampling technique is used or under fitting if the under-sampling technique is used ([Drummond et al., 2003](#); [Han et al., 2005](#)).

The cluster-based technique is performed by applying a clustering algorithm to the minority and the majority class instances to identify the cluster in each

class and resample the identified clusters (Chawla et al., 2004). The number of instances in each class may differ based on the clusters in each class. However, this method can suffer from overfitting issues when the cluster is oversampled (Yen and Lee, 2009).

The Synthetic Minority Over-Sampling Technique (SMOTE) or informed Over Sampling is a technique that adds synthetic instances to the training set from a subset of the minority class (Kubat et al., 1997; Chawla et al., 2002; Han et al., 2005). The goal is to reduce overfitting issues which are caused by oversampling techniques (Han et al., 2005). However, this technique does not consider instances from other classes which may cause classes to overlap and as a result increase noise in the dataset (Chawla et al., 2003).

Metrics Level

There are many metrics that can be used to measure the performance of the learning algorithm however, some can be misleading when the dataset is imbalanced such as accuracy of the model or overall error rate (He and Garcia, 2009). The reason behind that is that some metrics treat classes equally and as a result they neglect the performance of the minority class.

The main metrics that can be used to measure the performance of the imbalanced datasets are precision and recall and the area under the curve of the Receiver Operator Curve (AUC of the ROC). AUC of the ROC is basically showing the trade-off between true positive rate (sensitivity) and false positive rate (specificity) in different thresholds (He and Ma, 2013). The goal of this metric is to show the ability of the model to distinguish between classes. The recall is the true positive rate or sensitivity of the model while precision is the positive predictive value or the ratio of the positive instances that are correctly classified. The precision and recall are more appropriate metrics if one class has more priority (He and Ma, 2013). The AUC of the ROC is preferable

when the classes have similar priorities and the precision recall metrics are preferable when we are interested in one of the classes.

Class Level

The idea of class level is to penalize the learning algorithm in the training set based on misclassification error (Elkan, 2001). The learning algorithm will have a higher penalty if it is misclassified instances from the minority class. Many studies have shown that learning from the imbalanced dataset can be done through employing cost sensitive methods (Maloof, 2003; Chawla et al., 2004; Weiss, 2004).

These methods can be applied through cost sensitive learning and have three main classes. The first method is to apply the cost sensitive functions to find the best distribution for training the dataset (Zadrozny et al., 2003). The second cost sensitive method is to apply a cost minimizing technique through a learning step of ensemble learning algorithms where there are many classifiers used to find the optimal classifiers (Domingos, 1999). The cost sensitive method is to include the cost sensitive features or functions with the learning algorithm such as cost sensitive decision trees and cost sensitive neural networks (Kukar et al., 1998; Elkan, 2001).

Using the cost sensitive method required to find what is the optimal cost sensitive function and how to incorporate it with learning algorithms is usually used when the sampling techniques cannot be applied to the dataset because it is harder to implement compared with previous techniques.

Algorithm Level

One of the affective machine learning algorithms for dealing with imbalanced datasets is ensemble learning algorithms (Zhou, 2012; Kaur et al., 2019). The idea behind the ensemble learning algorithms is training many learners achieves

better results than training one learner. Ensemble learning algorithms have been proven to be better than resampling techniques for dealing with imbalanced datasets (Philip and Chan, 1998; Guo and Viktor, 2004; Liu et al., 2008; Kaur et al., 2019).

The main idea of ensemble learning for dealing with imbalanced datasets is training the model on different sets of instances to decrease the bias towards the majority class. The learners can be decision trees, logistic regression, or neural networks (Huang et al., 2000). There are two main types of ensemble learning algorithms that can be used to deal with imbalanced datasets: bagging and boosting. The bagging technique can learn from the imbalanced dataset in parallel by applying many classifiers then voting for the best classifier, while the boosting algorithms learn from the imbalanced dataset sequentially by trying to correctly classify the instances based on their weights (Schapire, 1990; Breiman, 1996).

Bagging

Bagging or bootstrap aggregating is an ensemble learning algorithm that trains many classifiers in parallel, then chooses the best classifier or aggregate of many classifiers for prediction and the instances are chosen randomly for the training set (Breiman, 1996). The main idea behind the bagging algorithm is to decrease bias by randomly training different instances. The best classifier is the one that has low error rate or achieves high accuracy (Breiman, 1996). The instances can be trained more than one time because the instances are chosen randomly and could be chosen by one or more classifiers. The main advantages of this method are: it decreases overfitting issues, achieves better results than single learners and it is not affected by noise (Breiman, 1996). However, bagging requires base learners to have high performance or it will not achieve the desired output (Krogh and Vedelsby, 1995).

Boosting

Boosting is an ensemble learning algorithm that learns in a cascading style and the learners are trained based on the weight of instances (Schapire, 1990). The main idea for boosting algorithms is to choose the instances for the next iteration of the training process based on their weights. The weight of instance is decreased when the instance is correctly classified by the learners and it increases if the learners misclassify the instance (Schapire, 1990). The learner should be weak or just better than a random classifier. The main difference between bagging and boosting is that the former has many learners that learn in a parallel way and the instances are replaced randomly while the boosting has one learner at a time and the instances cannot be replaced (Schapire, 1990; Freund and Schapire, 1995). The AdaBoost algorithm or adaptive boosting algorithm is a popular boosting algorithm that can tackle the issue of imbalanced datasets (He and Ma, 2013). There are many studies that combined the boosting models with other techniques to deal with imbalanced datasets such as SMOTEBoost to improve the SOMTE technique and DataBoost to improve the oversampling technique (Guo and Viktor, 2004). However, this method is affected when there is noise in the data because it will keep training to classify them correctly resulting in an increase in the chance of overfitting issues (Schapire, 1990).

2.2.2 Recommender Systems and Classifications

Recommender systems have been using classification algorithms for building recommendations such as the k-nearest neighbors classifier which has been heavily used for building a collaborative filtering recommender system. Building a hybrid recommender system by applying a Naive Bayes classifier and collaborative filtering recommender systems to generate recommendations can increase the accuracy of the recommender system (Ghazanfar and Prugel-Bennett, 2010). The classification algorithms can also be employed before or

after the recommendation processes for increasing the quality of recommendation lists. Support vector machines multiclass classifier is used to classify items based on their categories then feed information to a collaborative filtering recommender system to increase personalization in the recommendation lists and decrease issues of sparsity (Nicolas, 2015). The Naive Bayes classifier can be used after building the recommendations to filter the recommendation lists (Billsus et al., 2000). Classification can be applied before, through, or after building the recommendations to increase performance of recommender systems

2.3 Dealing with Risks in Recommender Systems

Addressing risk is an essential step in many areas. In different disciplines there are different risks definitions based on the discipline goals. The risk definition emphasises the effect of uncertainty on goals. According to ISO 31000 (2009), risk is the “effect of uncertainty on objectives”. In this section, we illustrate the definitions of risk in recommender systems as well as other related domains. For project management discipline, risk can either be a positive or a negative and the role of risk management is to increase the influence of positive risk and decrease the influence of negative risk (Rose, 2013). For software engineering, software risk is anything that causes loss in development of software and there are two types of software risks, internal and external risks, which are categorised based on the project manager’s ability to control risk (Boehm and DeMarco, 1997; Madachy, 1997). Bouneffouf et al. (2013) define risk in recommender systems as “possibility to disturb or to upset the user which leads to a bad answer for the user”. In this section we introduce risk management and how to assess risk in general and in recommender systems.

2.3.1 Risk Management

Risk management is a crucial step in any system to decrease threats that may negatively affect system goals. Risk management is not specified for specific systems but it is required in most of the systems, especially when there are many stakeholders that can influence system goals (Dali and Lajtha, 2012). Recommender systems are one of those systems that have many stakeholders and each of them plays a role that can influence recommender systems' goals. For instance, the users decision to make a purchase can affect e-commerce recommender systems, which have a goal of increasing sales of online stores, and ignoring user's behaviour that may affect buying decisions of the user is risky and can affect conversion rates of the business.

Risk management has many steps starting with identifying, assessing and ending with controlling of risk (Crouhy et al., 2005). The identification and analysis steps are the processes of finding in which circumstance the system may not achieve its goals or expectations by defining the threats that may negatively affect the goals of the system (Leitch, 2010). For instance, providing poor recommendations may negatively affect users' decisions to buy from online stores. Then, the assessment of risk step takes place by measuring how much the identified risks can affect the system goals (Crouhy et al., 2005). For instance, finding the probability the probability that the user will not buy from the online store based on the likelihood of the risks and their consequences if the users face a poor recommendations. Risk controlling is the process of acting to prevent or reduce influences of risk that may affect the goal of the system (Dali and Lajtha, 2012). For instance, how the system can mitigate poor recommendations or provide interesting recommendations to increase the chance of buying from the online store. Some of the business viewpoints ac-

cept risks and act based on that, particularly when the risk assessment shows that the risk influence is not high, or the mitigation of the risks is very costly. However, some risks need to be reduced or mitigated or otherwise the goals of the systems will not be achieved or hugely affected.

2.3.2 Risk Analysis and Assessment

There are two approaches for risk analysis and assessment: model driven and data driven. Each approach has advantages and disadvantages.

Model Driven Approach

The model driven approach depends on deep understanding of the system goals and the features or factors that influence them and how they are related. The data that is used should be clear and relative or the assessment will not be useful (Lund et al., 2010). It can provide explanations behind the assessment which are very useful for understanding the reason behind the results. It is considered a trial and error approach that is based on scientific modelling, hypothesis development and experiment testing. The main procedures to assess the risk based on the model driven approach are the feature exploration, analysis and assessment and testing processes (Power and Sharda, 2007). There are two types of model driven approaches: qualitative risk analysis and assessment, and quantitative risk analysis and assessment. Qualitative risk analysis and assessment uses the scale for measuring impact and the likelihood of risk factors to find the level of risk. This kind of risk analysis and assessment is easier to implant compare with quantitative risk analysis and assessment. Quantitative risk analysis and assessment is a numerical evaluation of the risks influences on the goals of the system.

Qualitative Risk Analysis and Assessment

Qualitative risk analysis does not require an expert to perform it and it is cost

effective (Curtis and Carey, 2012). The output of qualitative risk analysis and assessment could be used to plan for risk countermeasures as the output is more descriptive than the actual cost of the risk and the output of the analysis stage could be used as a foundation for quantitative risk analysis and assessment. It is performed frequently to analyse new risks that may appear (Rose, 2013). However, the estimation of risk is not as accurate as quantitative risk analysis and assessment. The goals that are affected by the risk should be considered and can be used to prioritize risk and plan for countermeasures (Curtis and Carey, 2012). The impact and likelihood can be categorical or numerical values as well as the risk.

The categorical values are defined from low to high or vice versa (Curtis and Carey, 2012). The numerical values are used to measure the risk level based on the numerical value of impact and likelihood or probability. One of the commonly known matrices is the Likelihood and Impact Matrix which is used for qualitative risk analysis and assessment (Rose, 2013). It is based on two components of risk: likelihood of occurrence and the impact on goals if it happens. This matrix is two-dimensional for mapping the occurrence likelihood or probability of the risk and their influence on the system goals (Rose, 2013). The output of the matrix is referred to as risk level or the degree of risk and is calculated by multiplying the two dimensions of the matrix. The qualitative risk analysis methods are based on the likelihood and impact of risk events.

Quantitative Risk Analysis and Assessment

Quantitative risk analysis and assessment is more accurate analysis than the qualitative risk analysis and assessment. However, it requires huge resources and time to be measured and evaluated accurately compared with qualitative risk analysis and assessment. This analysis is usually used for high priority risks. In some cases, it is not possible to perform this kind of analysis and assessment due to the lack of data. It is a repeatable process till the output

of the assessment reaches the desirable state. The most common methods based on the quantitative risk analysis and assessment are Sensitivity Analysis, Monte Carlo Simulations and Expected Monetary Value (Rose, 2013). Sensitivity Analysis is a technique used to evaluate the variables which have the highest influence on risk (Council et al., 2005). They help assess the risks with the highest influence and how variations in the goals and uncertainties are correlated and the impact of each risk factor on the system goals (Rose, 2013). Generally, it is only the high priority risks that are included in this kind of analysis. This analysis requires huge effort and resources to be implemented and usually it is performed after the Likelihood and Impact Matrix to identify the highest risks (Iloiu et al., 2009).

Expected Monetary Value is similar to the Likelihood and Impact Matrix because it involves multiplying the likelihood and impact of risks. The likelihood of a risk is identified and the impact is assigned a monetary value. The risk could be positive or negative and it assigns positive values for a positive risk (opportunity) and negative values for a negative risk (threat), and the decision trees are commonly used to measure Expected Monetary Value (Rose, 2013). Monte Carlo Simulations are used for quantifying risk (Rose, 2013). In risk management, the inputs are estimates of system goals and the model outputs are the likelihood of each goal due to the uncertainties (Rose, 2013). These outputs can then be used to identify countermeasures or a plan for controlling the risk.

Qualitative risk analysis is just for estimation purposes that depend mainly on the quality of the data that is used to build the analysis (Curtis and Carey, 2012). In some cases, risks that are quantitatively different may obtain similar risk levels. It is basically a subjective analysis and the output of the analysis may have different interpretations. The matrix shows the risk level for each risk factor but it doesn't provide the overall risk and it is also a repeatable

process (Curtis and Carey, 2012). Quantitative risk analysis is more accurate. Quantitative risk analysis is not as cost-effective as qualitative risk analysis. This analysis is basically dependent on the data that is used as input and any insufficient data will result in meaningless output (Curtis and Carey, 2012). One of the model driven risk assessments in the software engineering field is the Constructive Cost Model (COCOMO), a mathematical model used in software engineering to measure effort needed and assesses risks and efforts of a project by assigning numerical values to each factor (Madachy, 1997).

Data Driven Approach

The Data Driven approach is based on data and their patterns. One of the first risk assessment models based on the data driven approach uses neural networks algorithms and traditional statistical methods to improve credit risk models (Altman et al., 1994). It is more cost effective than the model driven approach (Kagermann et al., 2013). Deep understanding of the features that are used to assess risk is not essential and the assessment could be built without identifying the features, such as assessing risk based on neural networks or clustering. The main procedures to assess the risk based on the data driven approach is the training and testing process (Niesen et al., 2016). However, it requires data that represents all the cases that may happen. The interpretation of the results is not possible in some cases. Complexity or limited understanding of the output are an important issues in these black box techniques.

2.3.3 Risks in Recommender Systems

In recommender systems, there are many threats that can be considered as risks and as shown in Figure 2.4. Some of these threats are based on privacy perspectives, provider's view point, unfairness, unresponsiveness, pure person-

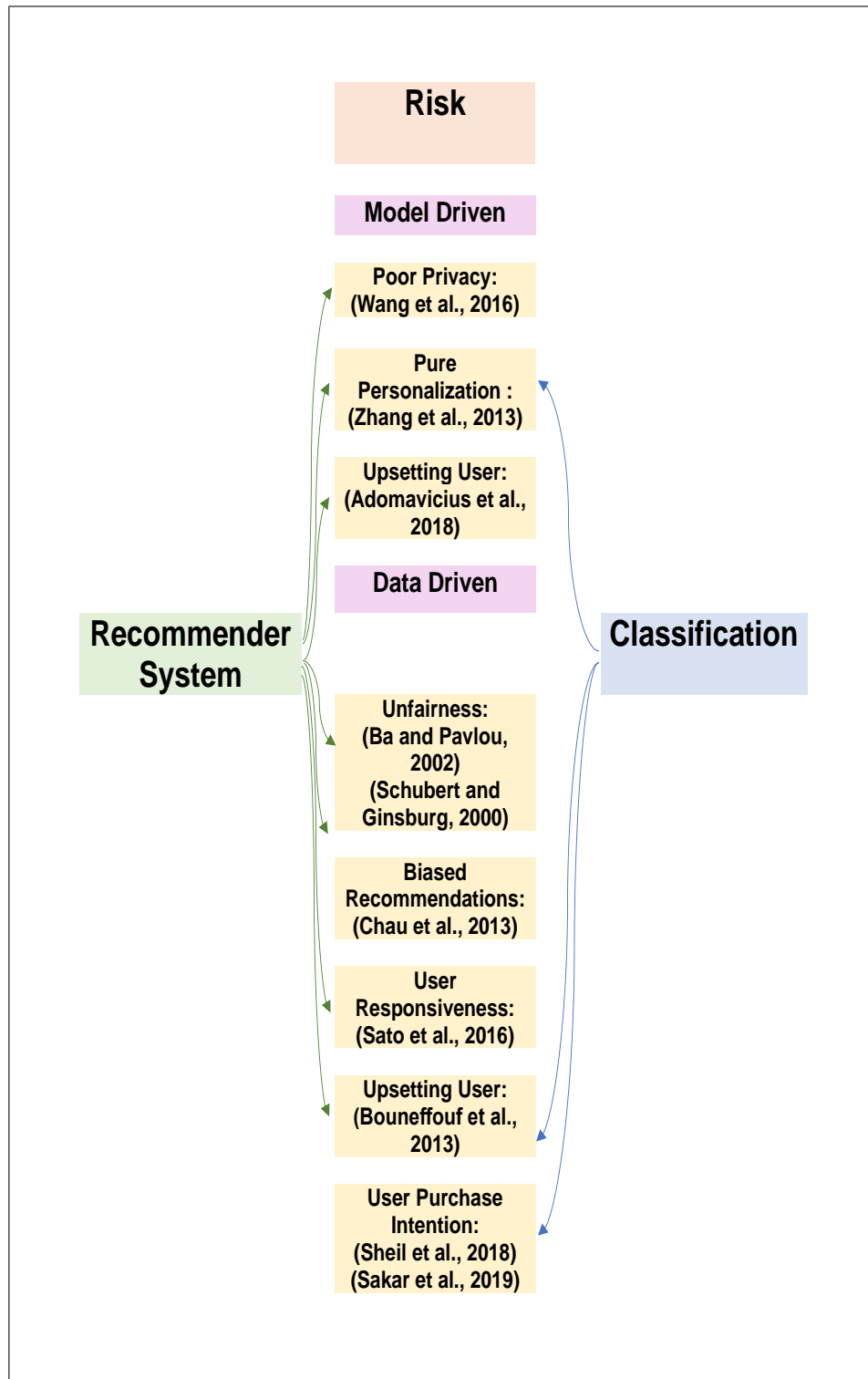


Figure 2.4: Recommender systems and risks

alization, and finally upsetting users by providing poor recommendations. Extracting user preferences by malicious users due to poor privacy is a risk (Wang et al., 2016). The system builds a user-to-user matrix to find the trust

between users and to prevent attackers from obtaining other users' preferences. The model adopts the data driven approach for assessing risk and it is based on a matrix factorization technique. However, the system built as part of that project has a scalability issue. Scalability is considered an essential factor in e-commerce recommender systems (Karimova, 2016).

Unfairness in community based recommender systems, where buyers and sellers communities are using the same platform of e-commerce, can have a negative influence on consumers because it affects trust (Schubert and Ginsburg, 2000; Ba and Pavlou, 2002; Chen et al., 2009). Unfairness risk assessment is required for building community based recommender systems, and they adopt the model driven approach by using categorical risk assessment (Schubert and Ginsburg, 2000). They proposed a model that adopts the model driven approach for assessing risk of unfairness by analysing the features of items such as the price based on regression analysis (Ba and Pavlou, 2002). Furthermore, providing biased recommendations to the user may be considered a major issue that affects trust and as a result, affects their decision to buy (Chau et al., 2013). The study proposed a model that adopts the model driven approach to study the influence of providing recommendations from a particular product vender on users trust (Chau et al., 2013).

Individual responsiveness could be considered a risk. Many researchers claimed that individual responsiveness is independent of the user (Shani et al., 2005; Jiang et al., 2015). However, there is a risk of ignoring the impact of recommendations, because not all consumers will accept recommendations in a similar way. Measuring responsiveness from the consumer can improve recommender systems (Sato et al., 2016). They proposed an algorithm that adopts the data driven approach, and shows the importance of including the risk of users not responding and shows the effect on recommendation accuracy (Sato et al., 2016).

Upsetting users by providing poor or inaccurate recommendations could be considered a risk. The risk of providing inaccurate recommendations may influence negatively consumers' decisions to buy (Adomavicius et al., 2018). They proposed a model that adopts the model driven approach, and shows the influence of inaccurate recommendations on buying decisions based on the passion regression model (Adomavicius et al., 2018).

Bouneffouf et al. (2013), developed a risk aware recommender system based on a data driven approach that recommended documents based on the user's contextual information and Click Through Rate (CTR), and they found that it increases the performance of recommender systems. Their system faced weaknesses such as the cold start problem of a new user, the navigation process to obtain a user's interest before providing a recommendation list, and the agenda form that needs to be filled for social modeling.

Pure personalization could be considered a risk because it may not be effective in all cases, some users prefer popularity based recommender systems which recommend items that are popular regardless of their similarities to users' preferences, instead of personalized recommender systems because usually sparse data is used to predicate personalized items (Zhang et al., 2013). They proposed a model that adopts the model driven approach, and the goal is to reduce the risk of user dissatisfaction by switching between popularity based recommender systems and providing relevant items based on users profiles (Zhang et al., 2013).

Personalization of recommendations could raise some issues such as bias towards subsets of popular items or lack of diversity in recommendation lists. The studies that solve the issues that may appear due to personalization are summarised in Table 2.1. However, the level of diversity that the recommender system can present in recommendation lists is ignored. Assessing the

user interaction is important to reduce irritating users, especially when the over diversity could affect user decisions negatively (Kapoor et al., 2015). The goal of assessing the risk of user interaction is to find the level of diversity the recommender system can present in recommendation lists. To the best of our knowledge there is no study that predicts the level of diversity before presenting recommendation lists.

Focusing on recommending the most relevant items to the user is the major goal of current recommender systems. However, users nowadays need more than that from recommender systems. They require recommendations to be more suitable for their situations or non-risky situations, and a more targeted recommendation list in risky situations (where the user is busy or does not have that much time). The need to understand the user situation and provide recommendations based on that situation may be considered an important factor to obtain benefits from recommendations and ensure positive influence on the user at the same time which increases user satisfaction. Many studies focus on the importance of predicting the user's purchase intention or assessing the risk of the user not making a purchase for enhancing recommendations, services and personalization by understanding how users browse for specific items or how they search for related items, that conclude that different recommendations need to be based on a user's intention (Morwitz and Schmittlein, 1992; Lo et al., 2016; Cheng et al., 2017). There are some studies which aim to classify sessions based on user's purchase intention. Sheil et al. (2018), propose a model to categorize users as clickers and buyers based only on a user's click-stream, this model uses a recurrent neural network RNN. The second model is the real-time prediction of online shoppers predicting a user's intention to purchase based on the user's clickstream and session information (Sakar et al., 2019). This model fed six features into the multilayer perceptron MLP: the month, number of pages visited by the user, region, operating sys-

tem, amount of time spent on the session and whether it is a special day or not, to classify sessions. However, normally e-commerce datasets are imbalanced because they depend on users behaviours, and these kinds of neural networks are biased towards the majority class. To the best of our knowledge, there is no research related to building recommendations based on user intention to make a purchase.

Table 2.1: Personalization Risk Countermeasures

Research	Approach	Goal	Methodology
(Davidson et al., 2010)	Model driven	Increase diversity in recommendation list	Adding some rules to the recommender system which is removing the videos that are too similar
(Li and Murata, 2012)	Data driven	Increase diversity in recommendation list	Hybrid recommender systems based on users and items clusters where the clusters that are similar to other clusters are deleted
(Zhang et al., 2013)	Model driven	Increase user satisfaction	Switching between popularity based recommender system and personalized based recommender system
(Said et al., 2013)	Data driven	Increase diversity in recommendation list	K-Furthest Neighbour Collaborative Filtering
(Lee and Lee, 2014)	Model driven	Decrease bias toward the popularity of items	Identified some users as experts and sent their recommendations to other users
(Zhang et al., 2018)	Data driven	Decrease bias toward the popularity of items	Applying a multi-armed bandits method by setting the upper confidence bound high when the restaurant is new which gives it more chance to be recommended
(Mansoury et al., 2020)	Data driven	Increase diversity in recommendation list	Applying post-processing to identify items that have low visibility, then apply bipartite graph between items and users to generate recommendation lists
(Boratto et al., 2021)	Data driven	Decrease bias toward the popularity of items	Balancing the training samples between the popular items and unobserved items.

2.4 Conclusion

This chapter presents the background and relevant research areas of e-commerce recommender systems. Furthermore, it sheds light on the gaps that the reviewed studies have. The first gap is how to assess risk in e-commerce recommender systems based on a model driven approach and a data driven approach. We choose to assess the risk of a user not making purchases because the main goal of the e-commerce recommender system is to increase sales. This gap has been addressed in Chapters 3 and 4. The second gap is how to reflect the e-commerce purchasing pattern to predict the level of diversity. The level of diversity can be predicted based on the users' behaviours and their interaction with recommender systems. All the reviewed studies increase diversity without considering the context of the user. This gap has been addressed in Chapter 5. The third gap is how to build a recommender system framework to present personalized and diverse recommendations based on the risk assessment. We choose to assess the risk of a user not making purchases because many studies show the importance of this risk, however; no study employs it to provide recommendations. This gap has been addressed in Chapter 6.

3

Risk Assessment based on a Model Driven Approach

The aim of the proposed model in this chapter is to assess the risk of a user not purchasing before providing a recommendation to an e-commerce user. The proposed model applies a model driven approach to classify sessions as risky or non-risky, which indicates the likelihood of the session to contain a purchase. The risk calculation is based on two behaviours and seven features.

3.1 Introduction

Several research projects emphasize the importance of predicting the user's purchase intention or assessing the risk of the user not making a purchase for enhancing recommendations, services and personalization by understanding how users browse for specific items or how they search for related items, which conclude that different recommendations are required to be based on a user's purchase intention (Morwitz and Schmittlein, 1992; Lo et al., 2016; Cheng et al., 2017).

The goal of the proposed model is to assess the risk of users not purchasing before providing recommendations. The proposed model identifies the features that can be used to assess the risk of the user not purchasing, then measures them using a model driven approach to calculate risk level.

In recommender systems, there are many threats that could be considered a risk because they may irritate users and negatively affect users decisions to make a purchase. The risk can be categorised based on the risk source into two categories which are user behaviour and population behaviour. User behaviour includes the implicit feedback from the user such as the time of the session and the number of clicks. There is a risk of unawareness of contextual information and tracking the number of clicks in the session, and awareness of these factors can increase the performance of recommender systems (Bouneffouf et al., 2013; Lo et al., 2016). Moreover, decreasing the number of items that users are interested in and if the user is a new visitor means that the likelihood to perform a purchase on the session is low (Pu et al., 2011). Lack of trust has an inverse relationship with user satisfaction and can lead the user to not perform a purchase (Pathak et al., 2010). There are other factors that can affect the risk of the user not purchasing and they can be assessed based on population implicit data such as an increase in the number of users who leave the system without performing a purchase (Armstrong et al., 2000).

Furthermore, decreasing the number of users who use the system increases the risk because the likelihood of performing a purchase may decrease (Bag et al., 2019).

In this chapter, we focus on identifying and assessing the risks that influence a user's intention to purchase based on user and population behaviour. We apply a model driven approach to assess the risk of a user not purchasing. This model driven approach is based on rules which are explicitly represented. The main advantage of using this approach is interpretation, the ability to explain the reason behind prediction. Furthermore, a model driven approach is not affected by imbalanced dataset issues because it is rule based.

Assessing risk in recommender systems is important to obtain the maximum benefit of the recommender systems. It is suggested that risk assessment can be part of the cost estimation process (Menzies and Sinsel, 2000). One of the effective cost estimation tools is the Constructive Cost Model (COCOMO), a mathematical model used in software engineering to measure effort needed and assess risks of a project (Madachy, 1997). Intermediate COCOMO includes subjective assessment of product, hardware, personnel and project perspectives to assess an effort adjustment factor. This model can be used to estimate software effort and risk assessment (Menzies and Sinsel, 2000). Furthermore, there is a relationship between risk and cost which is cost estimation. Cost estimation is a combination of actual cost and risk associated with that cost (Kansala, 1997). Although risk estimation in recommender systems is different from cost measurement in software engineering, we note that both domains required a notion of risk. The COCOMO model is used to find the cost and risk of a software project and we adapt the method to find the risk of a user not purchasing. The COCOMO model is a popular risk assessment model that transforms the qualitative scale into quantitative assessment (Merlo-Schett et al., 2003; Manalif et al., 2014). Qualitative risk assessment depends

on subjective assessment and the results may have different interpretations (Curtis and Carey, 2012). Adapting the COCOMO model can reduce the issue of the subjective assessment by transforming the subjective assessment of risk features to a quantitative value. The reason for choosing the COCOMO model is that it assigns numerical values to features to measure the risk. As a result, the novelty of the proposed model is that it is built based on a model driven approach and derived from an intermediate COCOMO model to assess the risk of a user not purchasing based on implicit data of the user's session to predict whether the session may contain purchase (non- risky session) or may not contain purchase (risky session).

This chapter is organized as follows. Section 3.2 describes the contributions of the chapter. Section 3.3 introduces the proposed model. We first identifies features that can be used to assess the risk of a user not purchasing, then we propose a model based on a model driven approach to predict and classify sessions. Section 3.4 presents the experimental results of the proposed method on two real world data sets. Section 3.5 concludes the chapter.

3.2 Contributions

This chapter addresses contribution C1 as described in Section 1.4, which is proposing a model to assess the risk of a user not purchasing, based on two behaviours and identifies seven features that can be used for risk assessment. The methodology that we used is described below:

- Firstly, we propose a model that extracts the features from e-commerce datasets and performs the experiments on two real world and publicly available datasets.
- Secondly, we adopt a model driven approach which is the Constructive

Cost Model to assess the risk of a user not purchasing.

- To show the effectiveness of the proposed model, we compare it against two state of the art models that are used to predict and classify sessions based on the users' purchase intention.

3.3 Proposed Model

There are many threats that influence the purchase intention and can be used to assess the risk of a user not purchasing. The proposed model identifies risks based on two behaviours: user behaviour and population behaviour. Each behaviour has features that can be used to estimate the risk of a user not purchasing. The proposed model will assign a numeric value to seven risk features to assess the risk of a user not purchasing, then classify sessions as risky or non-risky. For example, if most users are doing purchases on weekends, then the sessions on weekends will have a higher chance of containing a purchase, so, weekends are considered less risky than weekdays because the chance of session including a purchase is higher. The model obtains these features values from the intermediate COCOMO model to assess risk level ([Madachy, 1997](#)).

The user behaviour focuses on users themselves, including the four features: Contextual Information, User Interest, Responsiveness Degree, and Trust Degree. The Contextual Information feature assesses whether the user is busy (weekdays) bearing a lower chance of accepting recommendations, or not busy (weekends) bearing a higher chance of accepting recommendations. A higher value represents a busier user. The User Interest feature assesses the risk based on the number of items that the user is interested in at the current session and previous sessions. For example, if the number of items that the user is interested in is high, bearing a higher chance of accepting recommendations or (low risk), few items (high risk) bearing a low chance of accepting

recommendations. The Responsiveness Degree feature assesses risk based on the time spent viewing items, more time to view items (low risk) bearing a higher chance of accepting recommendations, short time to view items (high risk) bearing a lower chance of accepting a recommendations. The Trust Degree feature assesses the number of clicks in a session, more clicks (low risk) bearing a higher chance of accepting recommendations, fewer clicks (high risk) bearing a lower chance of accepting recommendations. Figure 3.1, shows how to assign values to features based on their influence on risk. Some of the features have a positive influence on risk and others have a negative influence on risk. For instance, low bounce rate exhibits a low risk while a high conversion rate exhibits a low risk. The proposed model, Risk Assessment based on a model driven approach (RA-MD), will formulate the risk from the user behaviour as follows:

$$\mathbf{UR} = CI \times UI \times RD \times TD \quad (3.1)$$

where CI is Contextual Information, UI is User Interest, RD is Responsiveness Degree and TD is Trust Degree.

The population behaviour focuses on other users and their influence on the session, and it contains three features: Bounce Rate, Population Traffic and Conversion Rate. The Bounce Rate feature assesses risk based on the rate of users who visit the page then leave the system during that session. A high bounce rate means high risk, bearing a low chance that the session will contain a purchase, or a low bounce rate means low risk bearing a higher chance that the session will contain a purchase. The Population Traffic feature captures the number of users who are interested in the same items that the current user is interested in, high Population Traffic (low risk) bearing a high chance that the

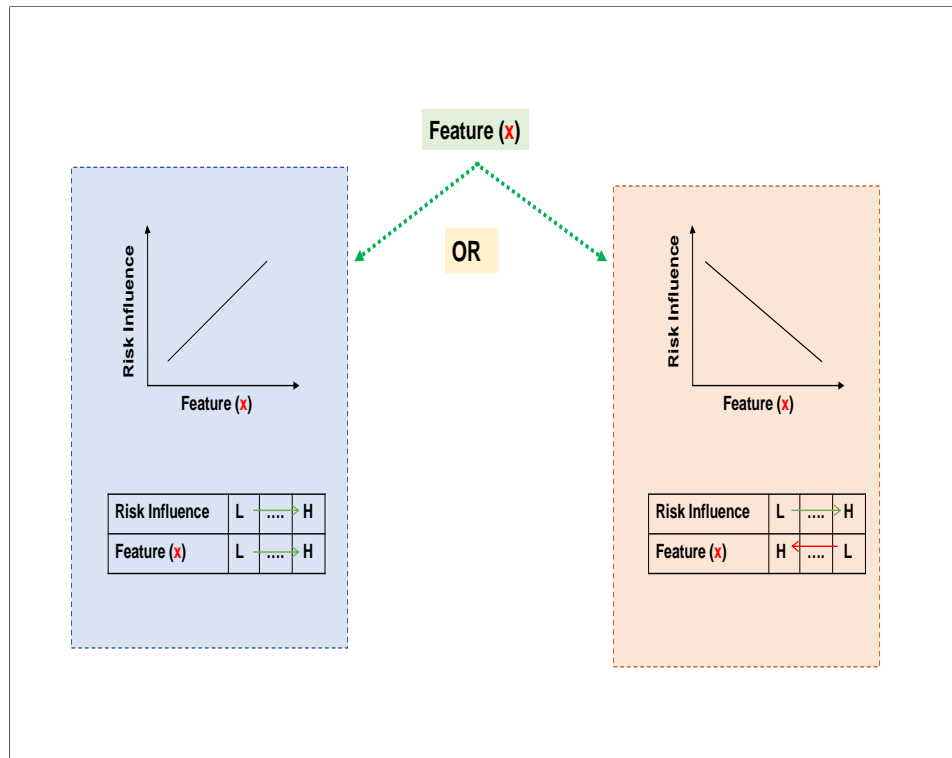


Figure 3.1: Assigning values of features

session will contain a purchase, or low Population Traffic (high risk) bearing a lower chance that the session will contain a purchase. A higher value of risk represents a low Population Traffic. The Conversion Rate feature assesses risk based on the rate of purchases divided by the number of total sessions, high Conversion Rate (low risk) bearing a high chance that the session will contain a purchase, or low Conversion Rate (high risk) bearing a lower chance that the session will contain a purchase. The system will formulate the risk of the population behaviour as follows:

$$\mathbf{PR} = BR \times PT \times CR \quad (3.2)$$

where BR is Bounce Rate, PT is Population Traffic and CR is Conversation Rate.

Table 3.1: Risk features value of the Proposed Model

User					
Risk Influence	Very Low	Low	Nominal	High	Very High
Contextual Information	0.75	0.88	1	1.4	
User Interest	1.29	1.13	1	0.82	
Responsiveness Degree	1.21	1.11	1	0.9	
Trust Degree	1.21	1.11	1	0.9	
Population					
Risk Influence	Very Low	Low	Nominal	High	Very High
Bounce Rate		0.87	1	1.15	1.3
Population Traffic	1.29	1.13	1	0.91	0.82
Conversion Rate	1.42	1.17	1	0.86	0.7

Each of the seven features has a rating on a five-point scale that ranges from “very low” to “very high” (where the value is based on their influence on the risk). All the values in Table 3.1 are derived from the COCOMO model (Madachy, 1997). For example: Conversion Rate scales: 1.42 (Very Low), 1.17 (Low), 1 (Nominal), 0.86 (High), 0.7 (Very High). Thus, if the Conversion Rate is very low then the multiplying feature will be 1.42 and will result in a higher risk level. This is the reason for applying multiplication between features. If the feature value is higher than one, the risk level will increase and if the feature value is less than one this means that the risk level will decrease. The values of features and their influence on the risk of the user not purchasing were validated through experimentation.

Table 3.2: Coefficient and the Exponent of the Proposed Model

User classes	a	b
Returning user	2.8	1.2
New user	3	1.12
Other	3.2	1.05

The (RA-MD) model will choose the coefficient a_i and the exponent b_i depending on the user class. The user from a returning class bears a high chance that the session will contain a purchase. For each class i , the values of a_i and b_i are different based on the effect of the class of risk, the values are derived from the COCOMO model (Madachy, 1997) and given in Table 3.2. The system will formulate the two behaviours as follows:

$$\mathbf{RL} = UR \times PR \quad (3.3)$$

$$\mathbf{R} = a_i \left(\frac{1}{1 + p_j} \right)^{b_i} \times \log(R) \quad (3.4)$$

where RL is the risk level, R is the risk value, p is the number of purchases of user j , and the coefficient a and the exponent b depend on user classes i . Equation 3.4, is derived experimentally, where a log transformation has been used to normalize the range of the R values. One of the commonly used method to find the optimal threshold for classification is the Youden index method (Kumar and Indrayan, 2011). This method defines the the optimal threshold as the threshold maximizing the Youden function which is the difference between true positive rate and false positive rate over all possible threshold values. The proposed model will find the optimal threshold to classify sessions using the Youden Index of the receiver operating characteristic curve of the validation set, as follows:

$$\theta = SN + SP - 1 \quad (3.5)$$

where θ is the threshold, SN is the sensitivity of the validation set of the receiver operating characteristic curve and SP is the specificity of the validation

set of the receiver operating characteristic curve.

Based on the literature review, there are many features that can be used to assess the risk of the user not purchasing. However, we choose the features that can be extracted from e-commerce data. Their effect on the risk of the user not purchasing has been proven experimentally.

The proposed model will classify a session as a risky session or a non-risky session based on the threshold derived by Equation 3.5. A risky session indicates that the user acceptance of recommendations is very low, and the session will have less chance of containing a purchase. A non-risky session indicates that user acceptance of recommendations is high, and the session will have a high chance of containing purchases.

3.4 Experiments and Results

In this section we conduct experiments to show features' influence of the proposed model (RA-MD) on the risk, compare the performance of the proposed model with state-of-the-art models and evaluate the effectiveness of the proposed model on different datasets. For the chosen datasets, 70% of the data was used for analysis and validation and 30% for testing. The validation phase aimed to find out in which situations of the recommender system, the user performs a purchase.

3.4.1 Experimental setup

The datasets that are used in the experiments are the Retail Rocket recommender system dataset (Ret, 2017) and the Online Shoppers' Purchasing Intention dataset (Sakar et al., 2019), and they are publicly available. To build an effective prediction model at least three clicks are required (Ding

et al., 2015). So, a pre-processing step was done and only the sessions that had at least three clicks were included. The Retail Rocket recommender system dataset contains the clickstream data, representing interactions that were collected over a period of 4.5 months. In total and after the pre-processing step there are 91,870 sessions. The Online Shoppers' Purchasing Intention Dataset contains the clickstream data, representing interactions that were collected over a one-year period and contains 10,320 sessions and each session belongs to a new user (Sakar et al., 2019). The main difference between the datasets is that the first dataset provides information about previous sessions of the user while the second dataset does not provide any information about users' past sessions. All experiments were run on an Intel Core i5-6300 2.4 GHz Windows 10 Pro system with 16 GB of RAM. Source code of RA-MD is available for download ¹. Table 3.3 contains statistics of the datasets.

Table 3.3: Statistics of Datasets

Dataset	sessions	Risky Sessions	Non-Risky Sessions
Retail Rocket recommender system	91,870	91.5%	8.5%
Online Shoppers' Purchasing Intention	10,320	82%	18%

The datasets are imbalanced so the accuracy of the models cannot be chosen as an evaluation metric to avoid a biased result toward the majority class; therefore, other metrics are used to evaluate the models: True Positive Rate (TPR), True Negative Rate (TNR), Precision, Recall, F1 score and AUC of the ROC curve. AUC of the ROC curve is used to evaluate models' capabilities of distinguishing between different classes.

There is research which aims to classify sessions based on user's purchase intention. There is a model to categorize users as clickers and buyers based only on a user's clickstream, this model uses a recurrent neural network RNN

¹Available at <https://github.com/DanahAG/RA-MD>

(Sheil et al., 2018). This model fed features into the recurrent neural networks (RNN): item id, timestamp, category of item, price, and quantity of the item to classify sessions. The second model is the real-time prediction of online shoppers which predicts a user's intention to purchase based on the user's clickstream and session information (Sakar et al., 2019). This model feeds six factors to the multilayer perceptron (MLP) to classify session: the month, number of pages visited by the user, region, operating system, amount of time spent on the session, and whether it is a special day or not. However, normally e-commerce datasets are imbalanced. The RNN and MLP neural networks are biased towards the majority class.

3.4.2 Experimental results

Four experiments were performed on each dataset. Firstly, we performed a sensitivity analysis on each feature and showed the impact on classifying session of a risky session and a non-risky session. Secondly, we compared the AUC of the ROC area of the proposed model with other state-of-the-art models to indicate the ability of the proposed model to distinguish between the risky sessions, sessions that do not contain purchases and the non-risky sessions, sessions that contain purchases. Thirdly, we compared the precision, recall, and the F1 score of the proposed model with other state-of-the-art models to measure the performance of the proposed model. Lastly, we showed the effect of user and population behaviours on the performance of the proposed model. A sensitivity analysis was performed for each feature to find their influence on the classifying sessions based on the deterministic method (Borgonovo et al., 2017). Sensitivity analysis is the study the relationships between feature values and the proposed model output. It shows how varying feature values can influence the output of the proposed model. The proposed model output classifies a session to a risky session and a non risky session. The goal is to

determine how sensitive the output is around a specific feature's values. For each feature, we found the probability the session is risky (does not contain a purchase) and the probability the session is not risky (contains a purchase) regarding their subjective assessment in Table 3.1. The first experiment was done to perform sensitivity analysis on each feature and show their impact on classification on the Retail Rocket recommender system dataset (Ret, 2017), where the information about previous sessions of the user is provided. The first experiment, Figure 3.2, shows some of the features have a positive influence such as bounce rate and some of them have a negative influence such as user interest and this can explain the reason why some of the features values are increasing and others are decreasing in Table 3.1. Figure 3.2, shows that the output of the proposed model on Retail Rocket Dataset is sensitive to the Responsiveness degree feature and the Trust degree feature. However, the Responsiveness feature has more influence on the output compared with the Trust degree. For the Responsiveness degree, around 80% of sessions will be risky if this feature is very low and 15% of sessions will be risky if it is high. For the Trust degree, around 95% of sessions will be risky if this feature is very low and 75% of sessions will be risky if it is high. It shows that the magnitude of influence is different between features, however, all the features have an influence on the risk of the user not purchasing. A sensitivity analysis was performed on the Online Shoppers' Purchasing Intention Dataset, where each session is for new users. Figure 3.3, shows the sensitivity analysis of the features of the proposed model and their influence on the risk of the user not purchasing. Furthermore, it shows that the influence magnitude is different between features. Figure 3.3, shows that the output of the proposed model on the online shoppers' dataset is very sensitive to the User interest feature and around 90% of sessions will be risky if the user interest is very low. Furthermore, it shows that the output of the proposed model on the online

shoppers' dataset is not very sensitive to the Responsiveness degree feature for different risk levels. Comparing this dataset with the previous dataset shows that the impact magnitude of the feature on the risk is dependent on the dataset's characteristic. For instance, the responsiveness degree has a higher impact on the risk on the Retail Rocket Dataset than the online shoppers dataset.

The second experiment compares the AUC of the ROC curve of the proposed model with other state-of-the-art models to indicate the ability of the models to distinguish between risky sessions and non-risky sessions. The experiment compares the prediction of our model against the predictions of the state-of-the-art models. We do this because the state-of-the-art models have different features. We are unable to do a fair comparison of the models in both datasets because these models are predicting the output based on different features and the datasets do not have all the required features of these models. The Retail rocket recommender system dataset does not have information about the region and operating system of the user and the Online shoppers' purchasing intention dataset does not have the price of the item and category of item. Figure 3.4, shows that the proposed model outperforms the model that uses a recurrent neural network RNN (Sheil et al., 2018) in terms of AUC of the ROC curve by 0.05 on the Retail Rocket Dataset, and the AUC of the ROC curve is equal to 0.89. The results show that the proposed model can distinguish between the sessions containing purchases and the sessions that do not contain purchases more accurately than the model that uses the recurrent neural network RNN. The results of the second experiment on the Online Shoppers' Purchasing Intention Dataset is show in Figure 3.4 and shows that the proposed model outperforms the model that uses the multilayer perceptron MLP (Sakar et al., 2019), in terms of AUC of the ROC curve by 0.01, and the AUC of the ROC curve is equal to 0.79. The proposed model can distinguish between the sessions

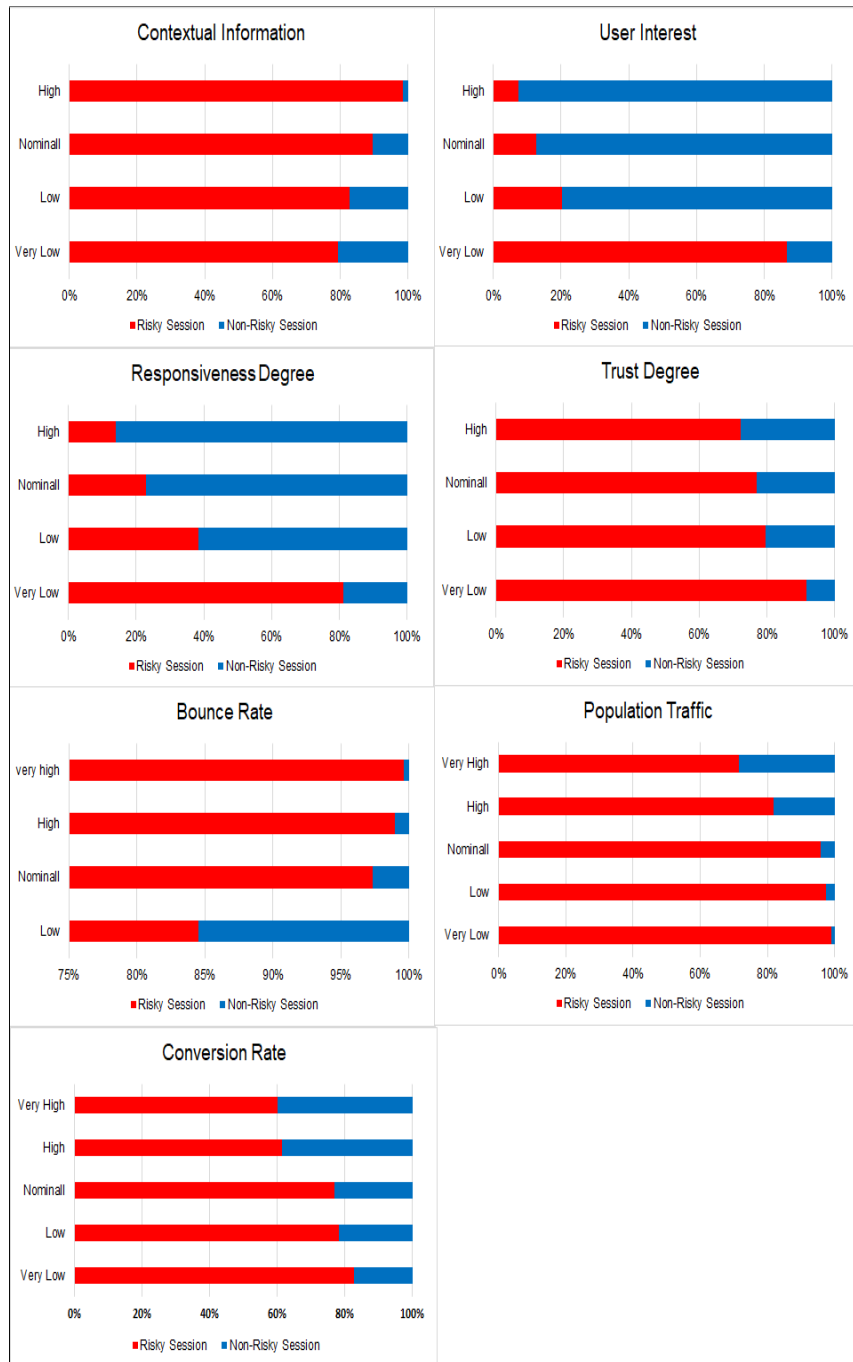


Figure 3.2: Sensitivity analysis of the features of the proposed model (Retail Rocket Dataset)

containing purchases and the sessions that do not contain purchases, even if there is no previous information about the user or if the user is a new user.

The third experiment was done to show the precision, recall and F1 score of the proposed model (RA-MD) compared with the model that uses the recurrent

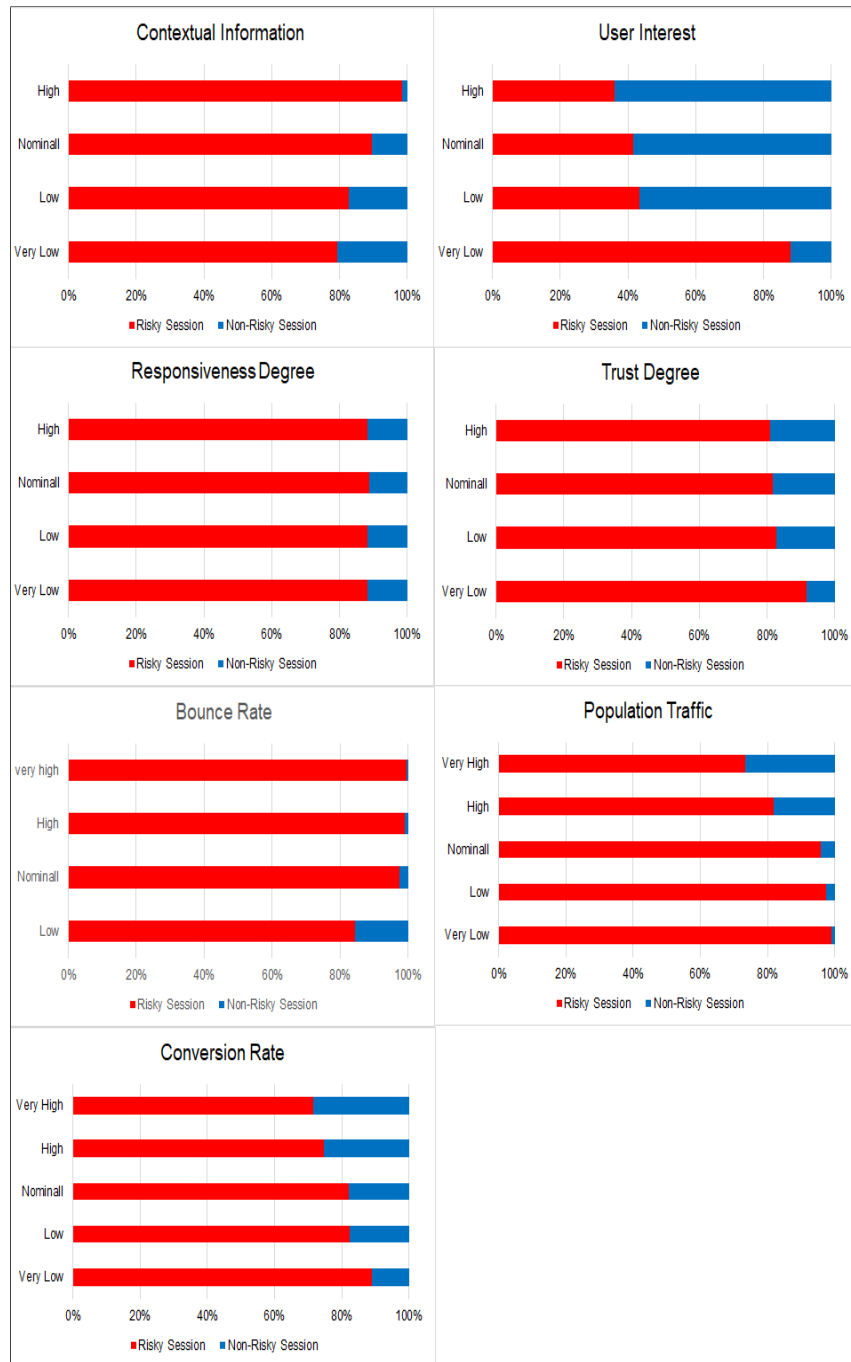


Figure 3.3: Sensitivity analysis of the features of the proposed model (Online Shoppers Dataset)

neural network RNN. Figure 3.5, shows that the proposed model outperforms the model that uses the recurrent neural network RNN in terms of Precision and F1 score by 8% and 4%, respectively. However, the recall value of the proposed model is equivalent to the recall value of the model that uses the

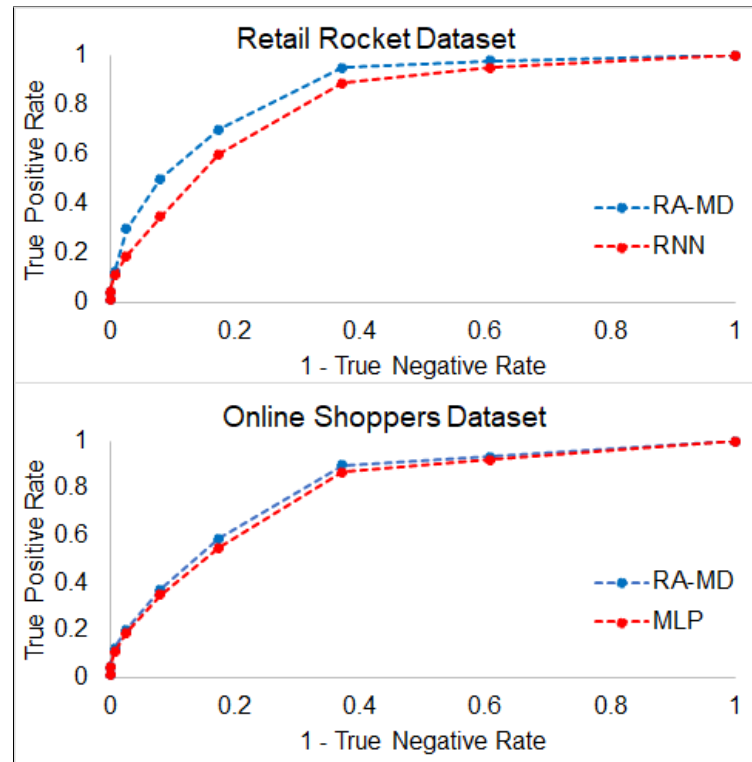


Figure 3.4: The ROC Curve of proposed model (RA-MD), MLP, and RNN

recurrent neural network RNN, and it is equal to 85%.

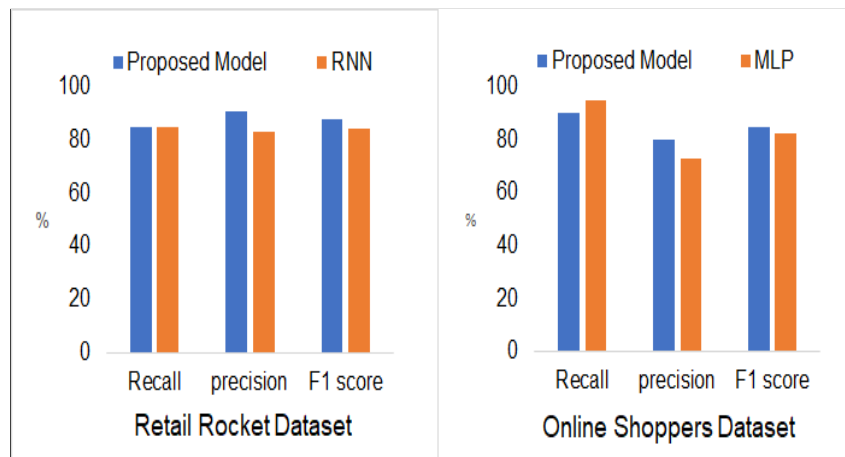


Figure 3.5: Evaluation of the proposed model(RA-MD), MLP, and RNN

The third experiment on the Online Shoppers' Purchasing Intention Dataset is shown in Figure 3.5, and shows that the proposed model outperforms the model that uses the multilayer perceptron MLP in terms of precision and F1

score by 7% and 2.5%, respectively, but underperforms the former model in terms of recall by 5%. However, it is expected to have lower precision because the proposed model has higher recall, and precision and recall have an inverse relationship. The F1 score of the proposed model outperforms the model that uses the multilayer perceptron MLP, and the F1 score is a more accurate measure of the model performance, especially if the dataset is imbalanced.

The final experiment was to show the importance of each behaviour of the proposed model (RA-MD) and how they influence the True Positive Rate (TPR), True Negative Rate (TNR) and the AUC of the ROC curve of the proposed model. Figure 3.6, shows that ignoring one of the behaviours can affect model performance, especially the TPR which can be negatively influenced and decreases by more than 40%, on the retail rocket dataset and by more than 50% on the online shoppers dataset, which means that the ability of the model to identify risky sessions is very low. As a result, the population and user behaviours are important for the proposed model to achieve high performance.

3.5 Conclusion

In this chapter, we proposed a model, Risk Assessment based on a model driven approach (RA-MD), to assess the risk of a user not purchasing based on two behaviours and identified seven features that can influence the user purchase intention. The proposed model assessed the risk of a user not purchasing by using a concept of the model driven approach which is the Constructive Cost Model, to predict and classify sessions as risky and non-risky sessions. Experiments show the effectiveness of the proposed model in assessing the risk of a user not purchasing before providing a recommendation, and the highest AUC of the ROC curve is 0.89. The result of the experiment shows that

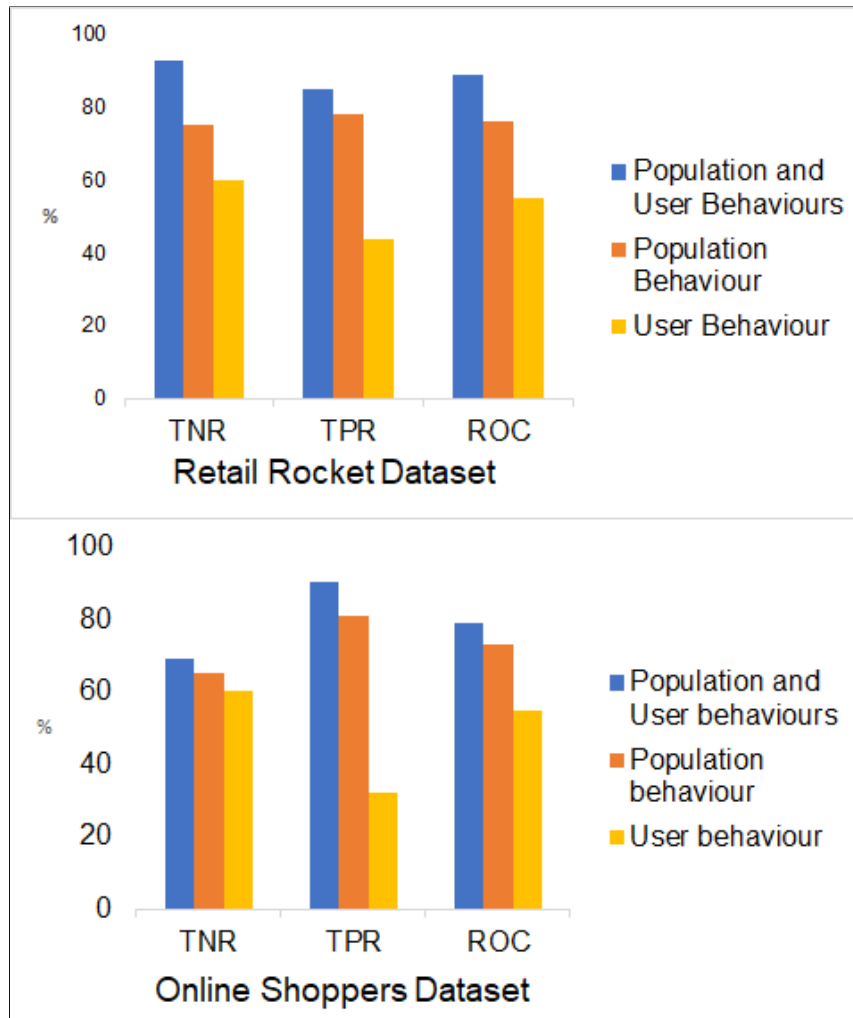


Figure 3.6: User and population behaviours influence on the proposed model(RA-MD)

the proposed model can be used to assess the risk of a user not purchasing, for new users, and the lowest AUC of the ROC curve is 0.79. Furthermore, the experiment shows that the proposed model outperforms two state-of-the-art models that are used to predict and classify sessions based on the user's purchase intention in regard to correctly distinguishing between risky and non-risky sessions. The limitation of this model is that it does not consider the change in users' behaviours over time and this can affect the performance of the model. In the next chapter we are going to propose a model that can tackle the issue of users' behaviours changing over time.

4

Risk Assessment based on a Data Driven Approach

The aim of the proposed model in this chapter is to predict the risk of a user not purchasing before provision of a recommendation to an e-commerce user. The proposed model applies a data driven approach to classify sessions as risky or non-risky and can handle issues of users' behaviours changing over time.

4.1 Introduction

In the previous chapter we proposed a model to assess the risk of a user not purchasing by using the concept of a model driven approach which is the Constructive Cost Model, to predict and classify sessions. Experiments show the effectiveness of the proposed model in assessing the risks of a user not purchasing before provision of a recommendation. However, the proposed model does not consider the changes of users' behaviours over time and this can affect the performance of the model. The e-commerce data is mainly affected by users and their behaviour over time, which means that change can happen to the data. The model driven approach cannot handle the change of users' behaviours over time because it is rule based. Changing the rules consumes resources and time because it needs to re-analysis data and build new rules. For this reason, the data driven approach could be considered more appropriate to handle this kind of change.

In this chapter we are going to assess the risk of a user not making a purchase and tackle the issue of change of users' behaviours based on a data driven approach. The data driven approach is built based on data patterns or machine learning algorithms. The main goal of the proposed model is to classify sessions based on the user's purchase intention. There are three main types of the machine learning approach for supervised classification: Classical machine learning, deep learning, and ensemble learning. The classical machine learning algorithm uses a single base learner algorithm that is trained by training sets, and then tests the model on unseen data. Ensemble learning uses multiple base learner algorithms to obtain higher accuracy than a single base learner algorithm. Deep machine learning or deep learning is a subset of machine learning algorithms that use multiple layers of neural networks as a learner to predict unseen data. Based on the literature all have their advantages and disadvantages. In summary, the classical approach can provide interpretation but

it cannot handle imbalanced issues or change of behaviours over time (He and Garcia, 2009). An imbalanced issue is when the classes are not balanced. For instances, the number of sessions that contain purchases is much lower than the number of sessions that do not contain purchases. The classical machine learning algorithms cannot handle imbalanced issues by itself. It is required to apply a change of data or algorithm to be capable to learn from imbalanced data and which may result in overlapping between classes (Kaur et al., 2019). For the deep learning approach, it can be more accurate than other machine learning algorithms but cannot handle change of behaviours over time and imbalanced issues (Nicolas, 2015). Applying a pre-processing step such as resampling techniques can improve deep learning algorithms' prediction on an imbalanced dataset but it may not reduce the error rate of the minority class (Kaur et al., 2019). For the ensemble learning approach, it can be interpreted (Friedman et al., 2000), but it is not as easy to interpret as some of the classical machine learning algorithms because it is based on many base learners. Furthermore, some of the ensemble learning can handle imbalanced issues such as boosting algorithms (Galar et al., 2011; Kaur et al., 2019).

Ensemble learning uses many base learners to predict unseen data. The base learners of the ensemble learning algorithm are learning in parallel stages and some of them are learning sequentially. The ensemble learning algorithms have three different approaches: bagging, boosting and stacking. The bagging algorithm is basically training base learners in parallel on different instances of training sets. The instances are chosen randomly from the training data sets, then a vote is taken to choose the more suitable base learner (Breiman, 1996). The boosting algorithm applies the same concept as the bagging algorithm but it learns sequentially and instead of choosing the instances from the training sets randomly they choose them based on their difficulty to classify, so those which are mis-classified will have more priority to be chosen on the next iter-

ation of the training process (Friedman et al., 2000). The stacking algorithm uses the different base learners as input then aggregates the output of the base learners to make a new learner based on those base learners (Zhou, 2012). From the previous description you can see that the boosting algorithm is the one that focuses on the instances of training sets and this can make them more powerful on handling the imbalanced data set. The goal of using the boosting algorithm is to correctly classify the instances that are difficult to classify.

One of the boosting learning algorithms that can be interpreted is the Adaptive Boosting algorithm (Friedman et al., 2000). This algorithm can handle imbalance issues of data sets (Sun et al., 2006). For imbalance issues, the Adaptive Boosting Algorithm can tackle imbalance datasets issues through the training process by adding more weight to the instances that are mis-classified after each iteration, and this gives them a higher priority to be chosen on the next iteration of the model training. The Adaptive Boosting uses the uniform distribution for the weight of instances which means that all instances initialize with equal weight and have similar priority to be chosen when the model starts the training process. However, e-commerce datasets have different characteristics than other datasets and it is suggested that recent data is more important than historical data (Whittington, 2013; Zhao and Cen, 2013). Investigating two real world publicly available e-commerce datasets (Ret, 2017; Sakar et al., 2019) shows that the behaviours may change over time and the analysis is illustrated in Figure 4.1.

To include the importance of the sessions, we proposed an algorithm that assigns higher weight to recent sessions which means that the recent sessions will have higher priority to be chosen by the base learner for training than older sessions. The proposed model uses the Bernoulli distribution for the sessions weights to add higher weight to recent sessions and lower weight to older ones. The goal of using this distribution is to let the model choose the recent sessions

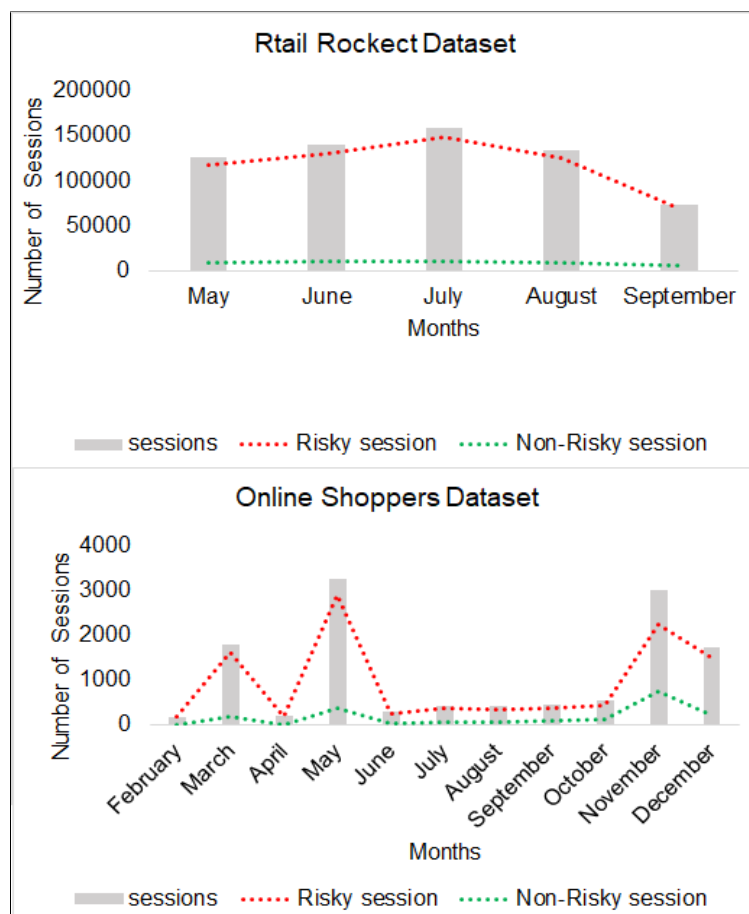


Figure 4.1: The change of behaviours over time for e-commerce datasets

on the early iterations of the training and make sure that they are classified correctly. We propose to use the Bernoulli distribution for sessions weights because the Bernoulli distribution has a linear time complexity, which means it will not influence the model complexity compared with other distributions that have exponential complexity.

This chapter is organized as follows. Section 4.2 describes the contributions of the chapter. Section 4.3 introduces the proposed model. We first introduce the problem of predicting the risk of users not purchasing based on historical data, then we propose a model that can handle the issue by adding higher weight to the recent instances of the training data set based on the Adaptive Boosting algorithm. Section 4.4 presents the experimental results of the proposed method on two real world data sets. Section 4.5 concludes the chapter.

4.2 Contributions

The contributions of this chapter are described below:

- Proposes an algorithm that predicts user purchase intention using a data driven approach.
- Compares the proposed algorithm with other machine learning algorithms to show the effectiveness of the model to handle e-commerce data.
- Evaluates the performance of the proposed model and shows that it has reached similar accuracy compared with the adaptive boosting algorithm but with a low number of iterations, which means with lower computational complexity.
- Shows that the proposed model can handle imbalanced datasets as well as data that are changing over time.

4.3 Proposed Model

One of the ensemble learning algorithms that learns sequentially is the Adaptive Boosting algorithm. The Adaptive Boosting algorithm uses the uniform distribution of the weight of sessions which means that all sessions initialize with equal weight and have similar priority to be chosen when the model starts the training process (Friedman et al., 2000). However, e-commerce datasets have different characteristics than other datasets and it is suggested that recent sessions are more important than historical sessions, or non-recent sessions (Whittington, 2013; Zhao and Cen, 2013). The proposed model, Risk Aware based on Data Driven approach (RA-DD), increases the importance of recent sessions to adapt the changes of users behaviours. We propose using the Bernoulli distribution for increasing the weight of the sessions. Applying

the Bernoulli distribution increases the chance of choosing recent sessions on the early iterations of training.

The proposed model, Risk Aware Data Driven algorithm (RA-DD), initializes the weight of the sessions, and assigns weight p to recent sessions and weight q to non-recent sessions. The sessions with weight p will have higher priority to be chosen for training iterations than sessions with weight q . A threshold is applied to categorise sessions to a recent or a non-recent session. The threshold depends on the session's timestamp and the training set size. The timestamp is used to sort the sessions in the training set in ascending order. For instance, when the threshold is 0.5 this means that 50% of the last half of the training set is going to have a higher weight than other sessions. Equation 4.1 has been used to initialize the weight of sessions based on Bernoulli distribution:

$$\mathbf{D}(\mathbf{i}) = \begin{cases} p & i = \text{recent session}, 0.5 < p \leq 1 \\ q = 1 - p & i = \text{non-recent session}, 0 \leq q < 0.5 \end{cases} \quad (4.1)$$

The weight of each classifier based on Adaptive boosting is calculated as follows:

$$\alpha_t = \frac{1}{2} \ln\left(\frac{1 - \epsilon_t}{\epsilon_t}\right) \quad (4.2)$$

where ϵ is the rate of the number of mis-classified sessions divided by the training set size, and t is the classifier.

The sessions weight updates based on Adaptive boosting , and is given by:

$$\mathbf{D}_{t+1}(\mathbf{i}) = \frac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t} \quad (4.3)$$

where D is weight of sessions i , when training classifier t . y_i is the correct output of session i and h_t is the predicted output by classifier t , and Z is the sum of all weights by the classifier t . The pseudocode of RA-DD is shown in Algorithm 1.

Algorithm 1: Risk Assessment based on Data Driven (RA-DD)

1. **Input:** Data set $D = \{(x_1, y_1), \dots, (x_m, y_m)\}$; Decision stump S ; Threshold θ ; Recent session weight p ; Non-Recent session weight q ; Number of Iterations T .

2. For $i = 1$ to m :

(a) If $D(x_i) \geq \theta$:

$$D_1(i) = p$$

Else

$$D_1(i) = q$$

3. For $t = 1$ to T :

(a) Fit a Decision stump s_t to the training data using weights D_i .

(b) Compute error of s_t

$$\varepsilon = \Pr_{\mathbf{i} \sim \mathbf{D}_t} [s_t(x_i) \neq y_i]$$

(c) Compute weight of s_t

$$\alpha_t = \frac{1}{2} \ln\left(\frac{1 - \varepsilon_t}{\varepsilon_t}\right)$$

(d) For $i = 1$ to m :

i. Update weight of session

$$\mathbf{D}_{t+1}(\mathbf{i}) = \frac{D_t(i) \exp(-\alpha_t y_i s_t(x_i))}{Z_t}$$

4. **Output:** $S(x) = [\sum_{t=1}^T \alpha_t s_t(x)]$.

In the previous chapter, we identified seven features that can be used to assess the risk of a user not purchasing and can be extracted from the e-commerce dataset. These features are categorised into user and population behaviours, with some of them relating to the user while other factors are

influenced by other users. The Risk Assessment based on a Model Driven Approach (RA-MD) assigns a value to the features based on Table 3.1, but the RA-DD features are either numeric or categorical, and they are described in Table 4.1, and the values will be based on datasets. The reason is each data set has different characteristics and as a result different values are required for different datasets.

Table 4.1: Risk features types of RA-DD

User	
Feature	Type
Contextual Information	Categorical
User Interest	Numerical
Responsiveness Degree	Numerical
Trust Degree	Numerical
Population	
Feature	Type
Bounce Rate	Numerical
Population Traffic	Numerical
Conversion Rate	Numerical

The user behaviour focuses on users themselves and contains four features: Contextual Information, User Interest, Responsiveness Degree, and Trust Degree. Contextual Information captures when the user is performing a purchase, such as on a weekend or weekday. User Interest captures how many items the user is interested in. For example, if the user is interested in many items the likelihood of performing a purchase may increase. The Responsiveness Degree feature captures the time spent viewing the items. Trust Degree indicates whether the user is responding to the system or not. This feature can be assessed by the number of clicks in a session.

Population behaviour focuses on the influence of other users and contains three features: Bounce Rate, Population Traffic, and Conversion Rate. The Bounce Rate feature captures sessions of users who visit the page then leave the system

during that session without any purchases and is calculated as follows:

$$\mathbf{Bounce\ Rate} = \frac{\text{Number of sessions that do not contain a purchase}}{\text{Total number of sessions}} \quad (4.4)$$

The Population Traffic feature captures the number of users who are interested in the same items that the current user is interested in. The Conversion Rate feature indicates sessions of users who perform a purchase divided by the total number of sessions and is calculated as follows:

$$\mathbf{Conversion\ Rate} = \frac{\text{Number of sessions that contain a purchase}}{\text{Total number of sessions}} \quad (4.5)$$

The features of user and population behaviours are fed into the proposed model, Risk Aware based on a Data Driven approach (RA-DD), which learns from a sequence of the training iterations. The proposed algorithm applies the same concept that the Adaptive Boosting learning algorithm used, which is iterating the base classifier and each session that is mis-classified will receive a higher weight, giving it more priority to be chosen on the next training, then a vote takes place to choose the best classifier, this can handle the imbalance issue. Furthermore, the proposed algorithm initializes the session's weight based on the Bernoulli Distribution and increases the weight of recent sessions. The recent sessions have more priority to be chosen for the training process to handle the change of behaviours over time.

4.4 Experiments and Results

In this section we conduct experiments to show sensitivity analysis of sessions' weights and threshold of recent sessions sets, then compare the performance of the proposed model with state-of-the-art models on different datasets. For the

chosen datasets, we used 70% of the dataset for training and 30% for testing the proposed model.

4.4.1 Experimental setup

Experimental setup for this chapter is similar to the previous chapter. Source code of RA-DD is available for download ².

4.4.2 Experimental result

Two experiments were performed on each dataset. Firstly, we performed sensitivity analysis of the sessions' weights and threshold of recent sessions subset on the F1 score of the proposed model, RA-DD. Secondly, we compared the AUC of the ROC curve area of the proposed model with other state-of-the-art models to indicate the ability of the proposed model to distinguish between the risky sessions, sessions that do not contain purchases, and the non-risky sessions, sessions containing purchases, and we tested the statistical significance of the RA-DD and other models which are a recurrent neural network RNN (Sheil et al., 2018), a multilayer perceptron MLP (Sakar et al., 2019) and the proposed model of the previous chapter, RA-MD.

The first experiment was done to perform sensitivity analysis of sessions' weights and threshold of recent sessions' subset on the F1 score of the proposed RA-DD model on each dataset. Figure 4.2 shows that when the threshold is 0.50, it means that 50% of the training set is considered as recent sessions, and increasing the weight of recent sessions p to 0.90 can cause overfitting on both datasets. This is because the model keeps training the recent sessions datasets and the difference between the weight of non-recent sessions q and recent sessions p is 0.80, when the training process starts. Decreasing the weight of p to 0.60 when the training process starts, results in a similar F1 score as

²Available at <https://github.com/DanahAG/RA-DD>

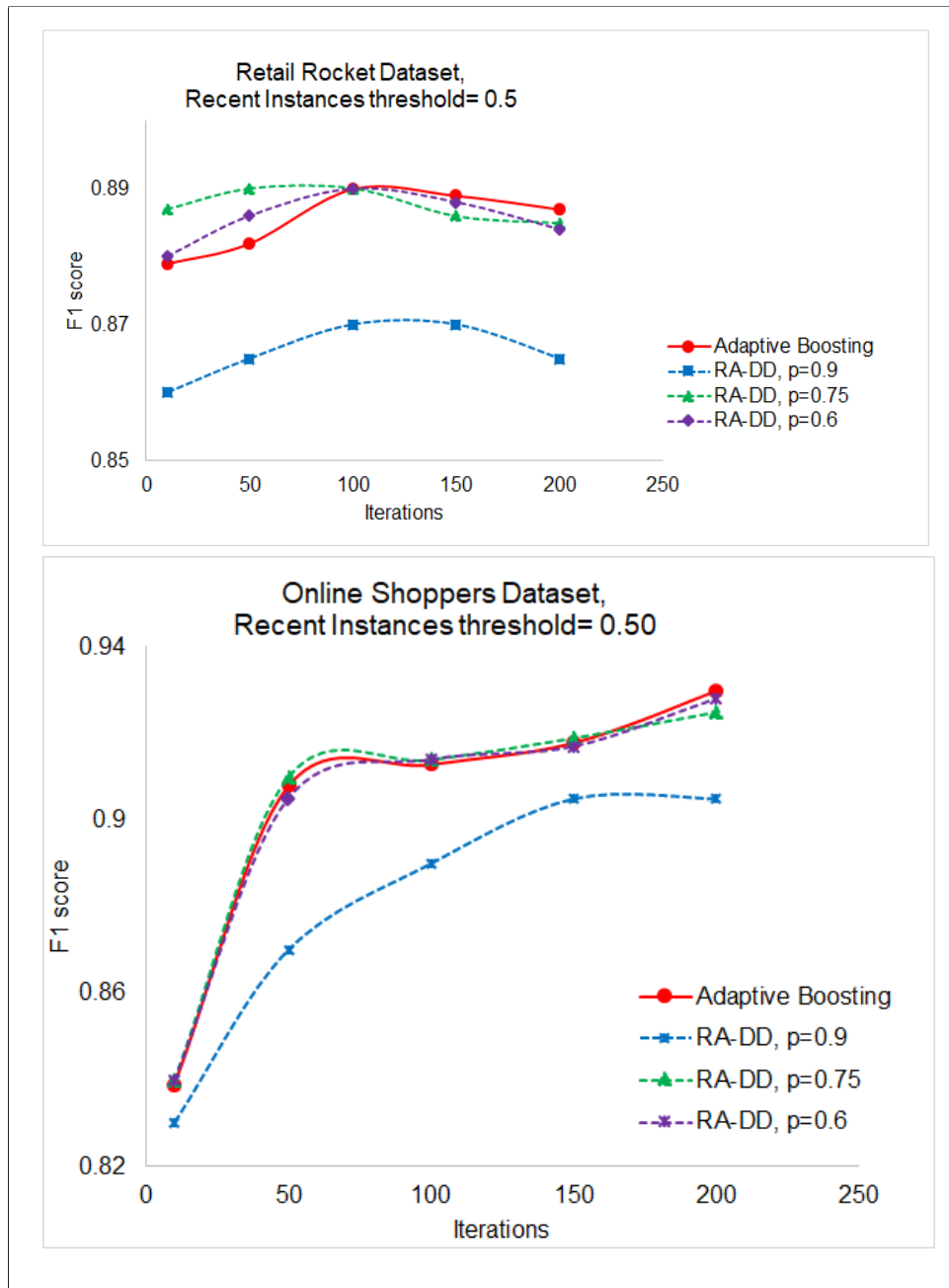


Figure 4.2: Influence of recent sessions weight in F1 score, Note: the y-axis for each dataset has a different range.

Adaptive Boosting because the difference between p and q is equal to 0.10. So the non-recent sessions' weights have a slight influence on the training process. The optimal weight of p is 0.75 for both datasets and the highest F1 score is when p is 0.75 and the threshold of the recent sessions set is 0.50 of the retail rocket dataset and the online shoppers' dataset is 0.89 and 0.93, respectively.

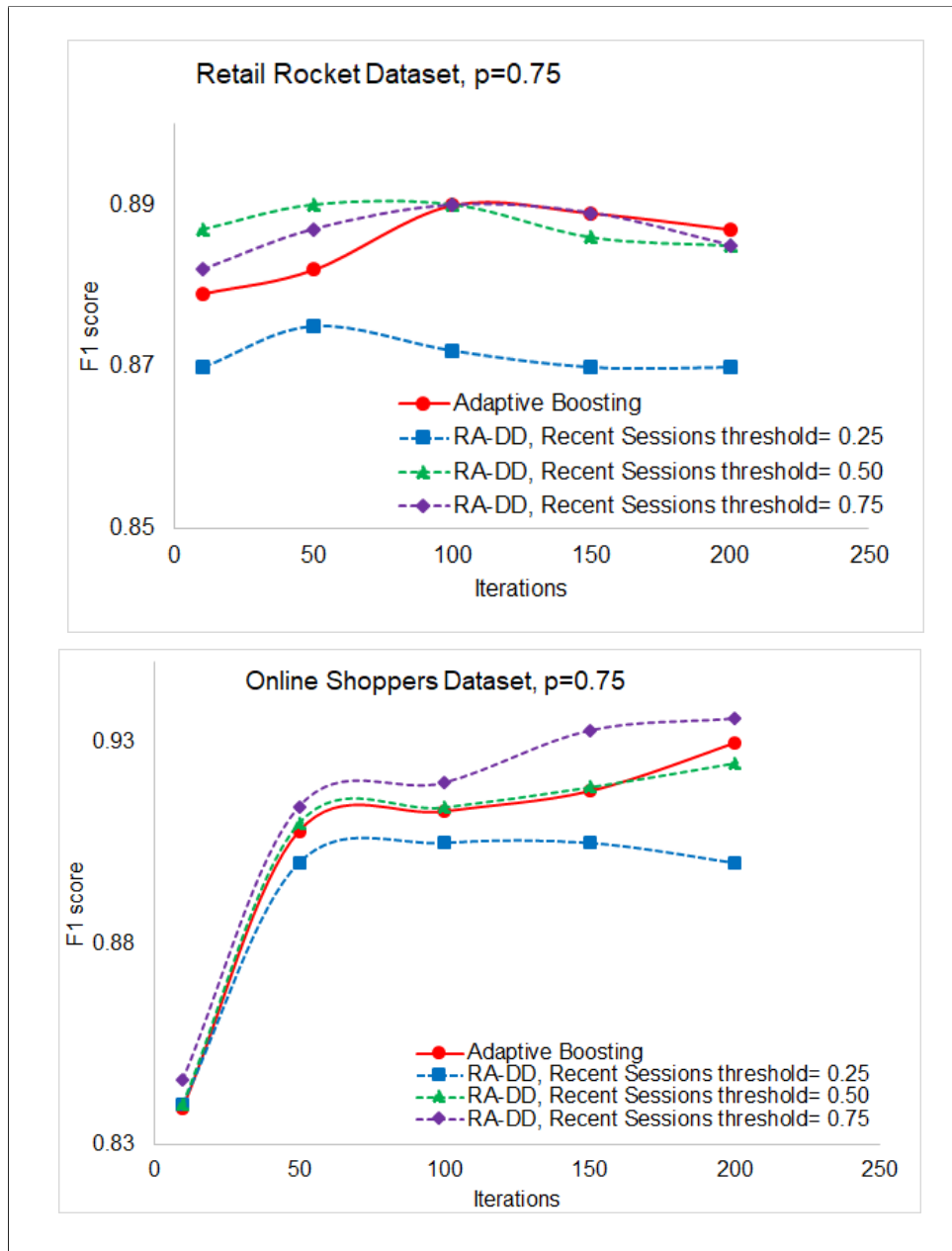


Figure 4.3: Influence of recent sessions threshold in F1 score, Note: the y-axis for each dataset has a different range.

Figure 4.3 shows that when the weight of the recent sessions p is 0.75, the optimal threshold of the recent sessions set of training set is 0.50 and the F1 score is 0.89 of the retail rocket dataset. For the online shoppers' dataset, the F1 score is 0.94 when the threshold of the recent sessions set of training set is 0.75 and the weight of the recent sessions p is 0.75. The optimal threshold is based on the change of behaviours of the dataset. However, in both datasets,

when the threshold of the recent sessions set of training set is 0.25, the proposed model obtains the minimum F1 score and it is equal to 0.87 and 0.91 on the retail rocket dataset and online shoppers dataset, respectively. The reason behind this is that the model focuses on the recent sessions set which is 25% of the training dataset and this means lower learning compared to those which have a higher threshold of the recent sessions sets.

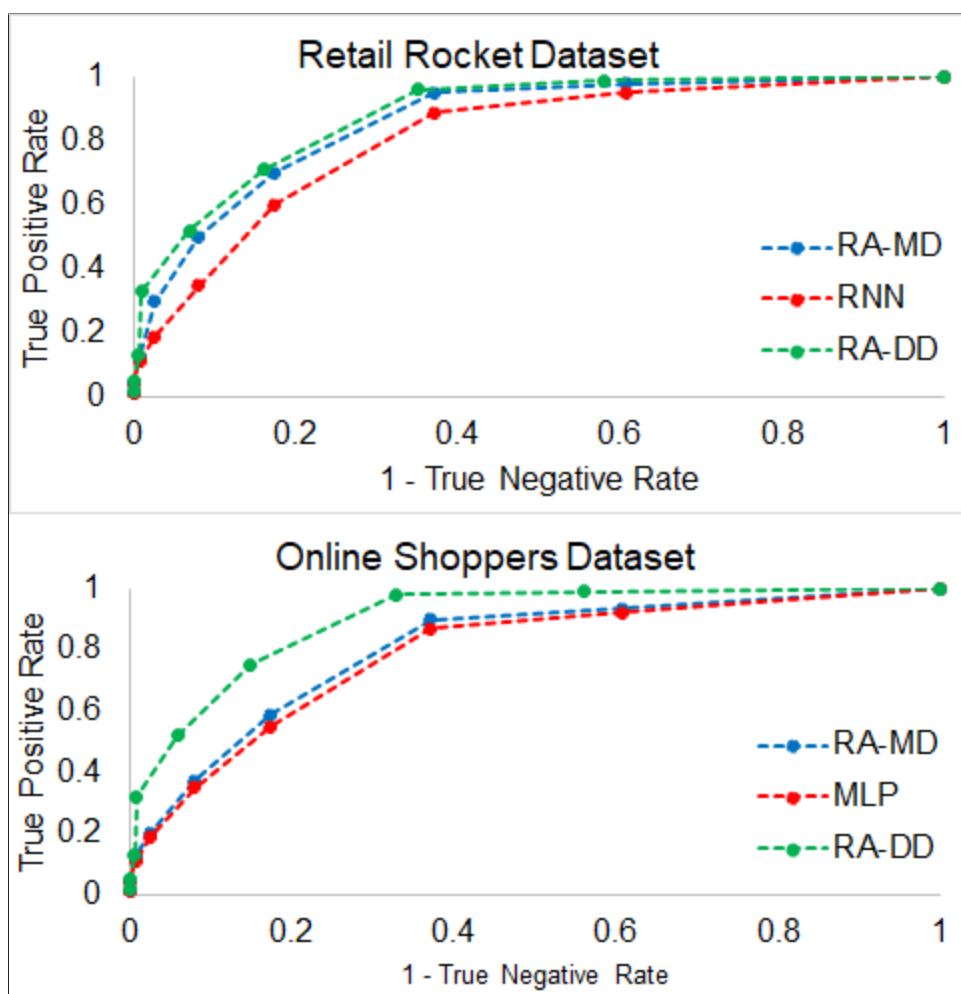


Figure 4.4: The ROC curve of the models

The second experiment was done to compare the AUC of the ROC curve of the proposed model, RA-DD, with other state-of-the-art models and Risk Assessment based on a Model Driven (RA-MD) to indicate the ability of the models to distinguish between risky sessions and non-risky sessions. Figure

4.4 shows that the RA-DD outperforms the model that uses a Recurrent Neural Network RNN (Sheil et al., 2018) and RA-MD in terms of the AUC of the ROC curve by 0.06 and 0.01, respectively, on the Retail Rocket Dataset, and the AUC of the ROC curve is equal to 0.90. The results show that the proposed model can distinguish between the sessions containing purchases and the sessions that do not contain purchases more accurately than other models. The results of the second experiment on the Online Shoppers' Purchasing Intention dataset are shown in Figure 4.4 which shows that the RA-DD outperforms the model that uses the Multilayer Perceptron MLP (Sakar et al., 2019) and RA-MD, in terms of the AUC of the ROC curve by 0.13 and 0.12, respectively, and the AUC of the ROC curve is equal to 0.915. To test the statistical significance of the RA-DD and other models, we applied the Friedman test according to the methodology recommended by (Vázquez et al., 2001; Pizarro et al., 2002), for testing models over a single dataset. The RA-DD is not statistically significant according to the Friedman test over all models on the Retail Rocket dataset (Friedman p-value = 0.0754). For the Online Shoppers' Purchasing Intention dataset, RA-DD is statistically significant according to the Friedman test over all models (Friedman p-value = 0.0498). The proposed model can distinguish between the sessions containing purchases and the sessions that do not contain purchases, even if there is no previous information about the user or if the user is a new user.

4.5 Conclusion

In this chapter, we proposed a Risk Assessment based on a Data Driven approach (RA-DD), to assess the risk of a user not purchasing and tackled the issue of behaviours changing over time. The proposed model assessed the risk

of a user not purchasing by using the concept of a data driven approach, the Adaptive Boosting algorithm, to predict and classify sessions to risky and non-risky sessions. Experiments show the effectiveness of the proposed model in assessing the risk of a user not purchasing before provision of a recommendation in two different datasets, and the highest AUC of the ROC curve is 0.915. The algorithm obtains a higher F1 score than with lower numbers of iterations compared to the Adaptive Boosting algorithm. Furthermore, the experiment shows that the proposed model outperforms two state-of-the-art models that are used to predict and classify sessions based on the user's intention to make purchases and the Risk Assessment based on a Model Driven (RA-MD) regarding to correctly distinguishing between risky and non-risky sessions. The limitation of this model is that it requires tuning more parameters than the Adaptive Boosting Algorithm to find the optimal weighting of the sessions and the threshold of the recent sessions set. In the next chapter we are going to propose a model that can be used to assess the risk of a user not interacting with the system to find how many categories that the user may explore during the session and tackle the issue of users' behaviours changing over time.

5

Risk Assessment of User Interaction based on a Data Driven Approach

In this chapter we propose a model that assesses the risk of a user not interacting by predicting how many different categories of items the system can include in a recommendation list before providing recommendations to the user. The proposed model is based on a data driven approach and will assess the risk of a user not interacting with the system based on users' behaviours.

5.1 Introduction

In the previous chapters we proposed two models that can be used to assess the risk of a user not purchasing before provision of a recommendation based on the user's behaviours. The goal is to understand their situation by predicting their purchase intention before providing a recommendation. In this chapter we are going to propose a model that assesses the risk of a user not interacting with the recommender system by predicting how many different categories the user will explore before provision of recommendations. The goal is to reduce irritating the user by proposing many different items to the user when they are not interacting with system and increase the chance of attracting a user when they are interacting with system.

There are many benefits of recommending different items to the user or increasing diversity in the recommendation list. Increasing diversity in recommendation lists can be used to increase the chance of making purchases (Ziegler et al., 2005; Knijnenburg et al., 2012). Furthermore, it can be used to reduce issues of recommendation lists overfitting, or depending only on past preferences of users to build recommendation lists (Ziegler et al., 2005; Willemsen et al., 2011). Moreover, it can be used to decrease items distribution long tail issues, or reducing issues of recommending a subset of items from an items catalog to different users because they have more interactions than other items (Tam and Ho, 2003). However, other research has shown that there is a risk of providing too many items in the recommendation list because it can affect user's decisions negatively (Kapoor et al., 2015).

It is important to understand the user's behaviour before providing diverse items in the recommendation list because different users accept different levels of diversity in recommendation lists. The goal of the proposed model is to suggest the number of different categories of items the recommender system will present to the user. We do this by considering the user's context and find

how likely it is that the user will explore the recommendation list. This can be done based on the prediction of how many different categories of items the user wants to see. Providing different items in the recommendation list or increasing the diversity of the recommended items can be measured by how different the categories of items are. The categories of items can be used to define diversity because categories are used to group things or items of a similar type. The category of items can be used to explicitly group items based on their features and has a general interpretation among users. Moreover, the category of items can be used to represent different suggestions for users, especially in the e-commerce context because the categories of items are well accepted for e-commerce categorization and are already available in most online stores. It can be assumed that the user will realize the diversity of the recommendation list if the categories are diversified among the recommended items.

The proposed model, Risk Assessment of User not Interacting with the system based on a Data Driven approach (RAI-DD), predicts how many different categories the user will explore during the session before providing the recommendation list based on the user's behaviour. The RAI-DD model will be based on Stagewise Additive Modeling using a Multi-class Exponential loss function (Zhu et al., 2006). The model tackles the issue of imbalanced datasets by increasing the weight of misclassified sessions in the training sets. Furthermore, e-commerce datasets have different characteristics than other datasets because the most recent sessions have more relevance than older sessions (Whittington, 2013; Zhao and Cen, 2013). Investigating two real world publicly available e-commerce datasets (Ret, 2017; Chen et al., 2012), show the behaviours may change over time and the analysis is shown in Figure 5.1. The proposed model will increase the priority of recent sessions by increasing their weights so the recent sessions will have more chance of being selected.

This chapter is organized as follows. Section 5.2 describes the contribu-

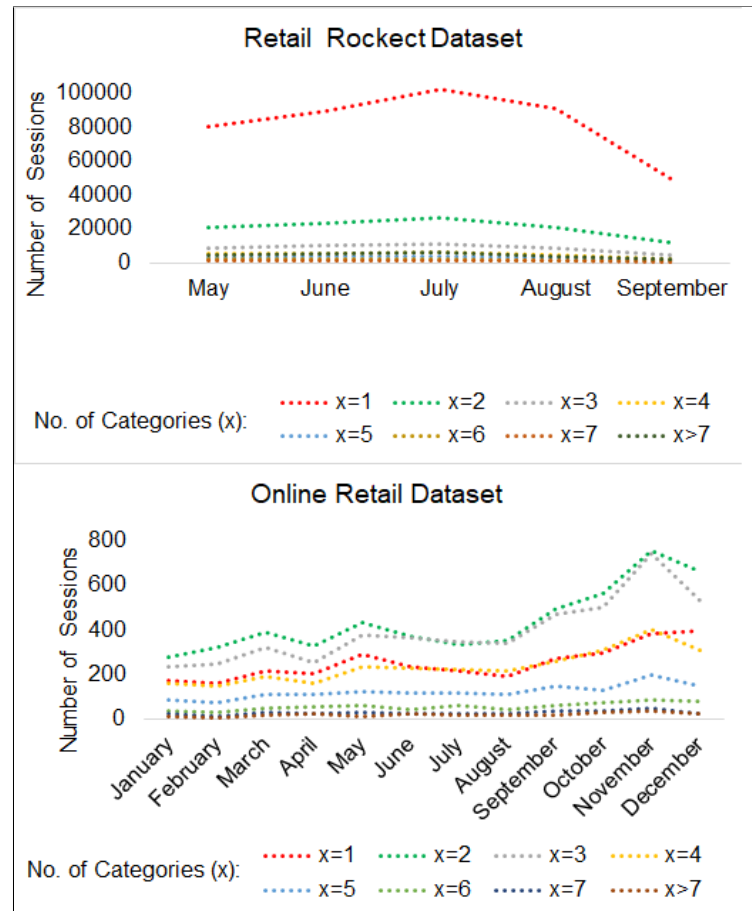


Figure 5.1: The change of behaviours over time for e-commerce datasets

tions of the chapter. Section 5.3, introduces the proposed model. We first introduce the problem of predicting the risk of users not interacting with the system, then we propose a model that can handle the issue by adding a higher weight to the recent sessions of training datasets, based on Stagewise Additive Modeling using a Multi-class Exponential loss function algorithm. Section 5.4 presents experimental results of the proposed method on two real world data sets. Section 5.5 concludes the chapter.

5.2 Contributions

The contributions of this chapter are described below:

- Proposes a model which assesses a risk of the user not interacting with

the system based on a data driven approach and predicts how many different categories the recommender system can provide to users.

- Proposes a model that can increase exploration of items or reduce the risk of irritating users with many recommended items when the users' behaviours show that they may not interact with the recommendations that include diverse items.
- Shows the effectiveness of the proposed model by comparing it with other recommender system approaches.
- Proposes and evaluates a model that predicts how many different categories the recommender system can provide in the recommendation list for new users.

5.3 Proposed Model

The proposed model predicts how many different categories the recommender system can include in the recommendation list based on a user's interaction with the system. The proposed model, Risk Assessment of User not Interacting with system (RAI-DD), is based on a data driven approach and applies an ensemble learning algorithm, the Stagewise Additive Modeling using a Multi-class Exponential loss function (Zhu et al., 2006).

The Stagewise Additive Modeling using a Multi-class Exponential loss function is one of the ensemble learning algorithms that learns sequentially. This algorithm is similar to the Adaptive Boosting algorithm which assigns a higher weight of misclassified instances that increases the chance of being chosen for the next iteration in the training process. The main difference between the Adaptive Boosting algorithm and Stagewise Additive Modeling using a Multi-class Exponential loss function is that the adaptive boosting algorithm requires

the base learner to be a better than random classifier and the error rate should be less than 0.5, while the Stagewise Additive Modeling using a Multi-class Exponential loss function does not require the base learner error rate to be more than 0.5. The Stagewise Additive Modeling using a Multi-class Exponential loss function has lower computational cost than the Adaptive Boosting when there are more than two classes because the Adaptive Boosting deals with multi-classes as one vs other classes and if there is K classes the computational cost will increase K times (Zhu et al., 2006). The Stagewise Additive Modeling using a Multi-class Exponential loss function algorithm can handle imbalanced issues of the data set because it is similar to the Adaptive Boosting algorithm which can handle this issue (Sun et al., 2006). For the imbalance issue the Stagewise Additive Modeling using a Multi-class Exponential loss function can tackle the imbalanced datasets issue through a training process by adding more weight to the instance that is mis-classified after each iteration, and this increases their priority to be chosen on the next iteration of the model training.

The Stagewise Additive Modeling using a Multi-class Exponential loss function uses the uniform distribution for the weight of instances which means that all instances initialize with equal weight and have similar priority to be chosen when the model starts the training process. However, the recent e-commerce datasets are more important than historical data (Whittington, 2013; Zhao and Cen, 2013).

The proposed model, Risk Assessment of User not Interacting with system based on a Data Driven approach (RAI-DD), increases the priority of recent sessions by increasing their weights. As described in Chapter 4, we propose to use the Bernoulli distribution for sessions weights because the complexity of this distribution is linear. The goal of using this distribution is to let the model choose the recent sessions on the early iteration of the training and make sure

that they are predicted correctly. The proposed model initializes the weight of the training set sessions, and assigns a weight p to recent sessions and weight q to non-recent sessions. The weight of recent sessions should be higher than the weight of non-recent sessions and the sessions with p weight will have a higher priority of being chosen for training iterations. Each session will be categorised as a recent or non-recent session based on a threshold. The threshold is based on the timestamp of sessions and the size of the training set. The timestamp is used to organise the training set from the oldest session to the newest session. For instance, when the threshold is 0.5 this means that 50% of the last half of the training set is considered as a recent sessions set. Equation 5.1 was used to initialize the weight of instances based on the Bernoulli distribution:

$$\mathbf{D}(\mathbf{i}) = \begin{cases} p & i = \text{recent session}, 0.5 < p \leq 1 \\ q = 1 - p & i = \text{non-recent session}, 0 \leq q < 0.5 \end{cases} \quad (5.1)$$

To calculate the weight of each classifier based on Stagewise Additive Modeling (Zhu et al., 2006):

$$\alpha_t = \log\left(\frac{1 - \epsilon_t}{\epsilon_t}\right) + \log(k - 1) \quad (5.2)$$

where ϵ is the rate of the number of mis-classified sessions divided by the training set size, t is the classifier, and k is the number of classes.

Then the sessions weight update based on Stagewise Additive Modeling (Zhu et al., 2006), and is given by:

$$\mathbf{D}_{t+1}(\mathbf{i}) = \frac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t} \quad (5.3)$$

where D is the weight of sessions i , when training classifier t . y_i is the correct output of session i and h_t is the predicate output by classifier t , and Z is the sum of all weights by the classifier t . The pseudocode of RAI-DD is shown in Algorithm 2.

Algorithm 2: Risk Assessment of User Interaction (RAI-DD)

1. **Input:** Data set $D = \{(x_1, y_1), \dots, (x_m, y_m)\}$; Decision stump S ; Threshold θ ; Recent session weight p ; Non-Recent session weight q ; Number of Iterations T .

2. For $i = 1$ to m :

(a) If $D(x_i) \geq \theta$:

$$D_1(i) = p$$

Else

$$D_1(i) = q$$

3. For $t = 1$ to T :

(a) Fit a Decision stump s_t to the training data using weights D_i .

(b) Compute error of s_t

$$\varepsilon = \Pr_{\mathbf{i} \sim \mathbf{D}_t} [s_t(x_i) \neq y_i]$$

(c) Compute weight of s_t

$$\alpha_t = \log\left(\frac{1 - \varepsilon_t}{\varepsilon_t}\right) + \log(k - 1)$$

(d) For $i = 1$ to m :

i. Update weight of session

$$\mathbf{D}_{t+1}(\mathbf{i}) = \frac{D_t(i) \exp(-\alpha_t y_i s_t(x_i))}{Z_t}$$

4. **Output:** $S(x) = [\sum_{t=1}^T \alpha_t s_t(x)]$.

The features that are fed into the proposed model are based on the users' behaviours and how the users are interacting with the system. To find how the a user is interacting with the system there will be some features that are related to the current session and others related to previous session of

the user. The contextual information features of the current session are the date and time of the session. The current information of the session is the number of clicks in the current session and the duration of the current session. The features that are based on user behaviour in the previous session are the rate of items that the user explored in the previous sessions, the number of previous sessions that have been performed by the user, and whether the user performs any purchases and the kind of user, such as if they are a new user or already a user. Table 5.1 shows the features and their type. The features that are related to the current session and other features that are related to the previous sessions of the user will be fed into the RAI-DD model to predict how many different categories we can provide to the user before providing a recommendation list.

Table 5.1: Risk features types of RAI-DD

Features that are related to the current session	
Feature	Type
Time of session	Numerical
Date of session	Numerical
Number of clicks	Numerical
Duration of the current session	Numerical
Features that are related to the previous sessions	
Feature	Type
Rate of Items	Numerical
Number of previous sessions	Numerical
Purchases	Binary
kind of the user	Categorical

5.4 Experiments and Results

In this section we conduct experiments to show a sensitivity analysis of sessions weight and thresholds of recent sessions, compare the performance of the proposed model with recommender system approaches and show the ability of

the model to predict how many categories the new user may explore during the session on different datasets. For the chosen datasets, we have used 70% of the dataset for training and 30% for testing the proposed model.

5.4.1 Experimental setup

The datasets that are used in the experiments are the Retail Rocket recommender system dataset (Ret, 2017) and the Online Retail dataset (Chen et al., 2012), which are publicly available. To build an effective prediction model at least three clicks are required in the session (Ding et al., 2015) so, a pre-processing step has been done and only the sessions that have at least three clicks are included. The Retail Rocket recommender system dataset contains the clicks data, representing interactions that were collected over a period of 4.5 months. In total and after the pre-processing step there were 91,870 sessions. The Online Retail dataset represents interactions that were collected over a one-year period and contains 18,535 sessions (Chen et al., 2012). The main difference between the datasets is that the Retail Rocket dataset provides information about the clicks and purchases that occur in sessions of the users while the Online Retail dataset provides information about purchases that occur in sessions of the users. The hardware and software environment that have been used for the experimental part in Chapter 3, have also been used in these experiments. Source code of RAI-DD is available for download ³. Table 5.2 contains statistics of the datasets.

The datasets are imbalanced so to avoid bias toward the majority class, the

Table 5.2: Statistics of Datasets

Dataset	sessions	categories	Period
Retail Rocket	91,870	8	May to September
Online Retail	18,535	8	January to December

³Available at https://github.com/DanahAG/RAI_DD

accuracy of the models cannot be chosen as an evaluation metric. Therefore, Precision, Recall, F1 score, True negative rate (TNR) and the AUC of the (ROC) curve used for evaluation.

To show the effectiveness of the proposed model (RAI-DD), we compared RAI-DD with a collaborative filtering recommender system based on purchases of items and a popularity based recommender system. Cosine similarity has been used to calculate the item-to-item similarity and the top ten recommended items. The base learners are a decision stump, tree with two leaves, because the datasets are imbalanced and this kind of tree does not consider the minority classes features as noise.

5.4.2 Experimental result

Three experiments were performed on each dataset. Firstly, we performed sensitivity analysis of the session's weight and threshold of recent sessions subset on the F1 score of the proposed model. Secondly, we compared RAI-DD with a collaborative filtering recommender system based on purchases of items and a popularity based recommender system. Finally, we showed the TPR, TNR and the AUC of the ROC curve of the proposed model for new users and compared it with other users.

The first experiment is to perform sensitivity analysis of sessions weight and threshold of recent sessions subset on the F1 score of the proposed model RAI-DD and Stagewise Additive Modelling using a Multi-class Exponential loss function Algorithm (SAMME) on each dataset. Figure 5.2, shows that when the threshold is 0.5, (where 50% of the training set is considered as recent sessions), the highest F1 scores are 0.77 and 0.74 for online retail data set and retail rocket data set, respectively, when p is 0.9, and they are the lowest F1 score compared with other p values. The reason behind this is that the proposed model keeps training on a subset of training set which causes over-

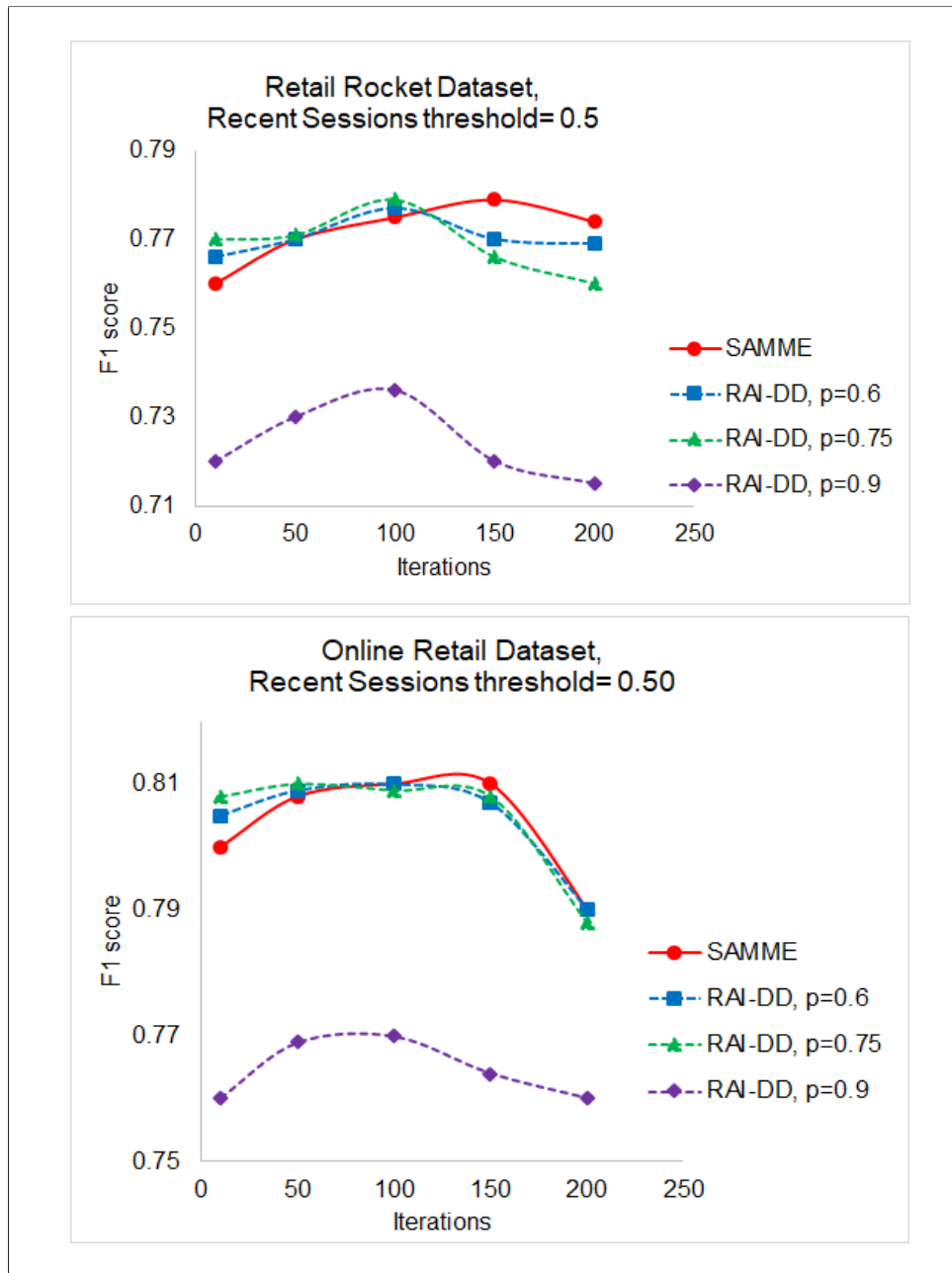


Figure 5.2: Influence of recent sessions weight in F1 score, Note: the y-axis for each dataset has a different range.

fitting. The optimal weight of p is 0.75 for both datasets and the highest F1 score when p is 0.75 and the threshold of the recent sessions set is 0.50 of the retail rocket dataset and the online retail dataset is 0.78 and 0.81, respectively. The (SAMME) requires 150 and 100 iterations to obtain the highest F1 score of the retail rocket dataset and the online retail dataset, respectively, while RAI-DD requires 100 and 50 iterations to obtain the highest F1 score of the

retail rocket dataset and the online retail dataset, respectively. The SAMME assigns similar weight to all sessions but RAI-DD assigns higher weights to recent sessions. Figure 5.3, shows that when the weight of the recent sessions p

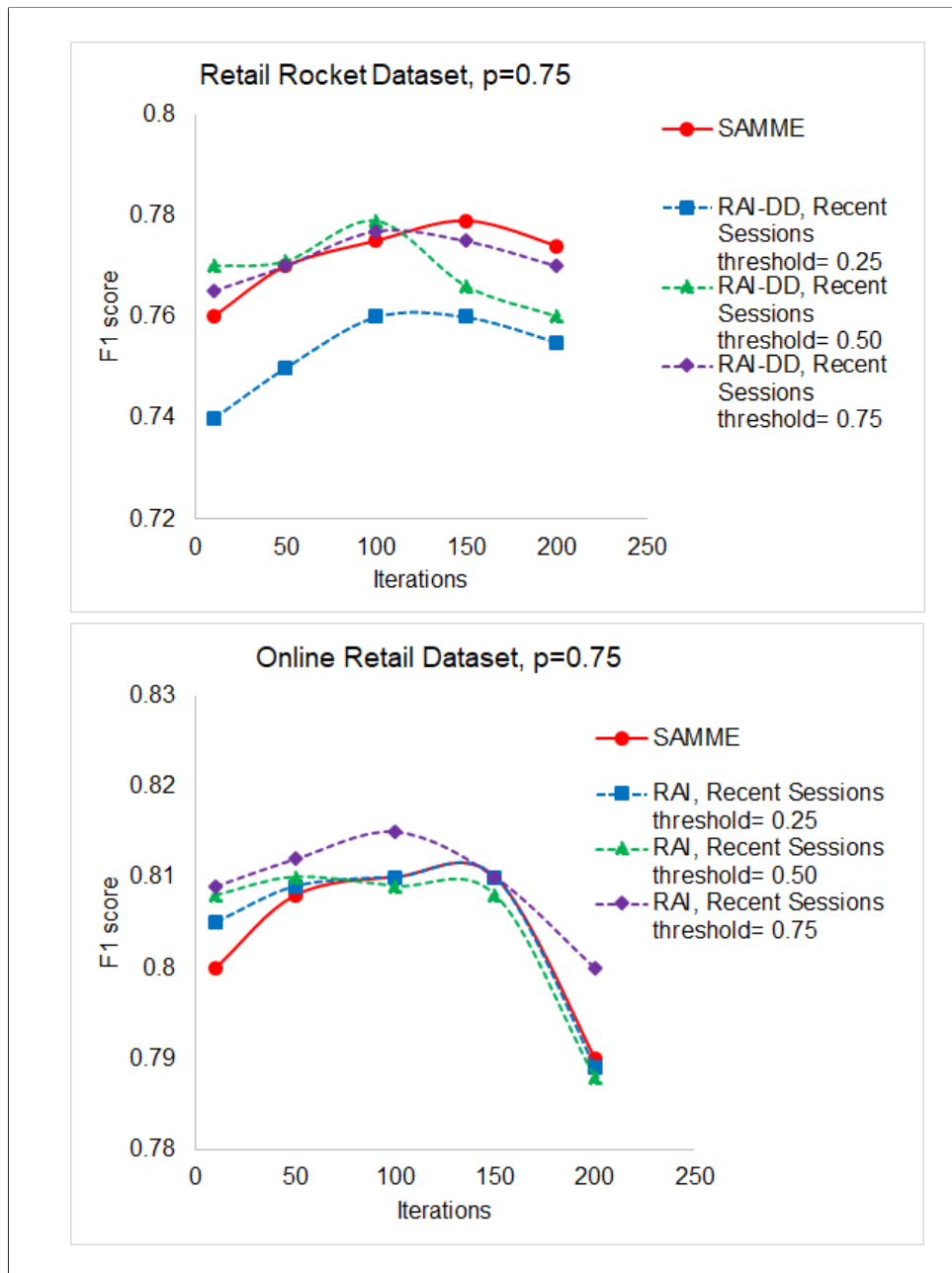


Figure 5.3: Influence of recent sessions threshold in F1 score, Note: the y-axis for each dataset has a different range.

is 0.75 and the number of iterations is 100, the optimal threshold of the recent sessions set of training set is 0.5 and the F1 score is 0.78 of the retail rocket dataset. For the online retail dataset, the F1 score is 0.82 when the threshold

of the recent sessions set of training set is 0.75 and the weight of the recent sessions p is 0.75. However, in both datasets when the threshold of the recent sessions set of training set is 0.25 and the number of iterations is 100, the proposed model obtains the minimum F1 score and is equal to 0.76 and 0.81 on the retail rocket dataset and online retail dataset, respectively. The reason behind this is that the model focuses on the recent sessions set which is 25% of the training dataset and this means that the model depends on recent sessions for prediction which causes overfitting or failing to generalize on unseen data. Finding the optimal threshold is based on the changes of behaviours in the datasets.

In the second experiment, we show the precision, recall of the proposed model, item based collaborative filtering recommender system and popularity-based recommender system, for two datasets, retail rocket dataset and online retail dataset shown in Figure 5.4. For the item based collaborative filtering recommender system, the proposed model outperforms the former recommender system with regard to precision by 66.2% and 60% for the retail rocket dataset and the online retail dataset, respectively, and outperforms the former recommender system with regard to recall by 63.2% and 41.6% for the retail rocket dataset and the online retail dataset, respectively. The reason behind this is that the collaborative filtering approach provides a recommendation list without considering user behaviour and more than 50% of the recommendation list provided based on this approach contains four to five categories of items but more than 80% of the sessions have fewer than three categories in the retail rocket dataset and more than 70% of the sessions have fewer than three categories of online retail dataset. For the popularity based recommender system, the proposed model outperforms the former approach with regard to precision by 74.2 % and 81.5 % for the retail rocket dataset and the online retail dataset, respectively, and outperforms the former recommender system with regard to

recall by 73.3 % and 75.7 % for the retail rocket dataset and the online retail dataset, respectively. The reason behind this is that the popularity based recommender system provides recommendation lists without considering user behaviour and almost all the recommendation lists provided based on this approach contain eight categories of items but fewer than 20% of the sessions have more than six categories for the retail rocket dataset and fewer than 30% of the sessions have more than six categories for the online retail dataset. For the proposed model, evaluating precision and recall are important because we want the model to get as many sessions that are correctly predicted as possible.

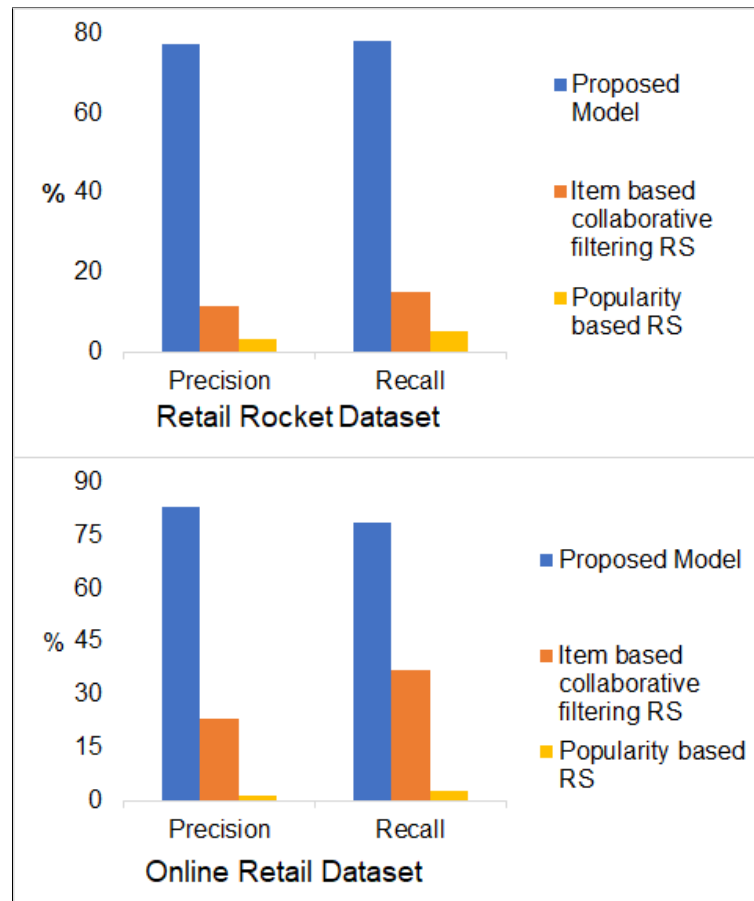


Figure 5.4: Comparison of the proposed model and other recommender systems approaches

In the third experiment, we show the TNR, TPR and the AUC of the ROC curve of the proposed model for new users sets and past users sets for two

datasets, the retail rocket dataset and the online retail dataset in Figure 5.5. For the new users set, the TPR of the proposed model is lower than the past users set by 1.1% and 7.3% for the retail rocket dataset and the online retail dataset, respectively, and the TNR of the proposed model is lower for new users than the past users set by 0.5% and 4.6%, for the retail rocket dataset and the online retail dataset, respectively. The AUC of the ROC curve can show the effectiveness of the model to distinguish between classes, especially when there is an imbalance issue on the datasets. The AUC of the ROC curve of the new users set is lower than the past users set by 0.5% and 1.9% for the retail rocket dataset and the online retail dataset, respectively. The experiment result shows that the proposed model can effectively predict the number of categories that we can provide to new users.

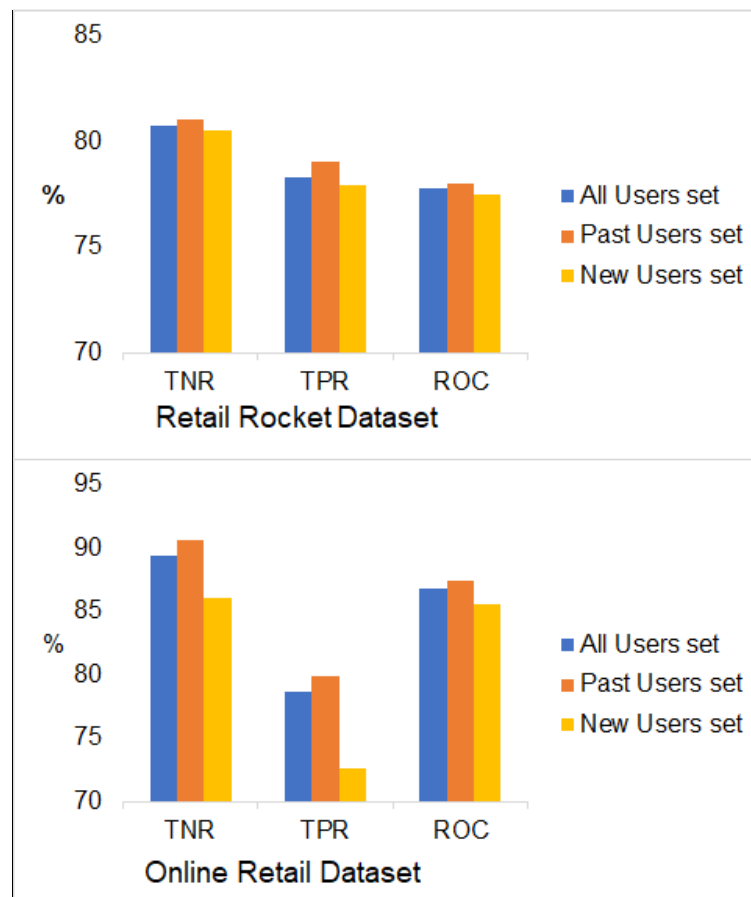


Figure 5.5: Comparison of the proposed model on different datasets

5.5 Conclusion

In this chapter, we proposed a model, Risk Assessment of a user not Interacting with the system, to assess the risk of a user not interacting and tackle the issue of users' behaviours changing over time. The proposed model assesses the risk by using concepts from Stagewise Additive Modelling using a Multi-class Exponential loss function Algorithm, to predict how many different categories the recommender system can provide to users in the recommendation list. Experiments show the effectiveness of the proposed model in assessing the risk of a user not interacting before provision of a recommendation in two different datasets, and the highest F1 score is 0.81. The algorithm obtains higher F1 scores with a lower number of iterations compared to the Stagewise Additive Modelling using a Multi-class Exponential loss function algorithm. Furthermore, we carried out experiments against a collaborative filtering system that used cosine similarity to measure the similarity between items, and a popularity based recommender system. The experiments on the datasets show that the proposed model outperforms the two recommender systems with regard to predicting the number of categories the recommender system should present in the recommendation lists. The limitations of this model are that it may increase the complexity of the recommender system and requires tuning more parameters than the Stagewise Additive Modelling using a Multi-class Exponential loss function algorithm to find the optimal weighting of the sessions and the threshold of the recent sessions set. In the next chapter we are going to propose a risk aware recommender system that can be used to provide a recommendation list based on assessing the risk of the user not purchasing and to reduce risk of recommendation list overfitting by finding how many

categories that the user may explore during the session.

6

Personalized Risk Aware Recommender System

The aim of the proposed framework in this chapter is to build a recommender system that considers the risk of a user not purchasing before provision of recommendations to the user. The goal of the framework is to reduce the chance of irritating users with a diversity of items when the user's purchase intention is low by providing personalized recommendations and to increase the diversity in the recommendation list when the user's purchase intention is high. The level of diversity depends on the risk assessment of user interaction, because different users may prefer different levels of diversity.

6.1 Introduction

Recommender systems in the e-commerce context are important from a user viewpoint and a business viewpoint. The benefit of building these systems in the e-commerce domain from the users viewpoint is to save the user time and effort and from the business viewpoint is to increase sales (Ricci et al., 2011; Lee and Hosanagar, 2016). The recommender systems in the e-commerce domain have focused on providing recommendations based on a user's preferences, regardless of the user's current situation. However, this situation may prevent users from accepting the recommendations and it may irritate them instead of helping them to find what they are looking for. Providing poor or inaccurate recommendations could be considered a risk because it upsets users and as a result negatively influences a consumer's decision to purchase (Adomavicius et al., 2018).

The goal of the proposed framework is to provide a personalized recommendation when the risk of user not purchasing is high and to increase diversity in the recommendation list when the risk of the user not making a purchase is low. One of the roles of embedded marketing is to find out how stores should organise products to customers (Wheelwright and Clark, 1992). The proposed model will inject diverse items from different categories based on the end cap zone that is used in item placement in the retail stores. This end cap strategy has an influence in how customers interact and can increase sales (Caruso et al., 2018).

Diversity generally is how different the items are in the recommendation lists (Bradley and Smyth, 2001; Hurley and Zhang, 2011). Diversity is important in the recommender systems for many reasons. Coverage of items or providing non-obviousness recommendations have a positive influence on user satisfaction and the performance of recommender systems (Herlocker et al., 2004). Recommending a wide range of items is generally a good approach to enhance

the chances that the user is pleased by at least some of the recommended items and it may enhance business (Fleder and Hosanagar, 2009).

Evaluating recommender systems based on accuracy only will have several limitations. The diversity and novelty can be used for evaluating the quality of a recommendation list (Adomavicius and Tuzhilin, 2005; Herlocker et al., 2004; McNee et al., 2006). Novelty in recommendations is important when there are a few items which are extremely popular and the rest of them are much less known (Anderson, 2006). The novelty can be assessed by the long tail distribution of items (Anderson, 2006; Brynjolfsson et al., 2010).

There are two kinds of diversity: individual diversity and aggregate diversity (Adomavicius and Kwon, 2011). Individual diversity is how different items are in a recommendation list for a user. Aggregate diversity is the total number of different items a recommendation algorithm can provide to users. Aggregate diversity is an important factor for business and users because increasing the coverage of items across users can increase sales as well as user satisfaction (Vargas and Castells, 2014). Aggregate diversity considers recommendation diversity across all users (Adomavicius and Kwon, 2011). Aggregate diversity can be measured by the number of items in the test set the model recommends in the recommendation lists divided by the number of all items in the training set which is the coverage of items (Robillard and Walker, 2014; Shlomo et al., 2018).

Providing diverse items in the recommendation list or increasing the diversity of the recommended items can be measured by how different the categories of items are, because categories are used to group things or items of a similar type. The category of items can be used to represent different suggestions for users, especially in the e-commerce context because the categories of items are well accepted for e-commerce categorization and are already available in most online stores. The user may realize the item diversity if the categories of items

are diversified among the recommended items. From the previous chapters, we assessed the risk of a user not making purchases based on user behaviour and population behaviour, then we assessed the risk of a user not interacting with the system before provision of recommendations to predict how many different categories we can provide to the user in the recommendation lists. The goal of risk assessment is to decrease the chances of irritating the user when the session is risky or the user's purchase intention is low and increase diversity in the recommendation list which can be used to increase the chance of making a purchase when the session is not risky or the user intention to make purchase is high.

This chapter is organized as follows. Section 6.2 describes the contributions of the chapter. Section 6.3 introduces the proposed model. We first introduce the problem of the current recommender systems, then we propose a model that can handle the issue, by including the risk of the user not purchasing and the risk of recommending a narrow range of items. Section 6.4 presents the experimental results of the proposed method on two real world data sets. Section 6.5 concludes the chapter.

6.2 Contributions

The contributions of this chapter are described below:

- Proposes a recommender system that considers user behaviour by assessing the risk of the user not purchasing and does not rely on user preferences only.
- Proposes a recommender system which increases the diversity of the recommendation list based on the risk of a user not interacting with the system.

- Shows the effectiveness of the proposed model by comparing it with other recommender system approaches and shows how the proposed recommender system can decrease long tail issues of items distribution.
- Proposes a recommender system that increases coverage of items in the recommendation lists.

6.3 Proposed Model

The goal of the e-commerce recommender system is to provide recommendations to users to save their effort and time and, as a result increase their satisfaction which persuades them to make purchases and increase sales (Ricci et al., 2011; Lee and Hosanagar, 2016). The e-commerce recommender system usually provides recommendation lists based on the preferences of users (Terveen and Hill, 2001). There are many approaches of recommender systems for providing recommendation lists based on user preferences, the main approaches being collaborative filtering approach, the content based approach and the hybrid approach (Ricci et al., 2011). The collaborative filtering approach recommends items based on other users who prefer similar items to provide personalized recommendations (Terveen and Hill, 2001; Ricci et al., 2011). However, there is a risk of providing recommendations based only on the past preferences of users because these preferences may change overtime. Relying on these preferences can cause overfitting issues of the recommendation list or keep recommending similar items to the user because it makes the recommender system decrease the coverage of items by recommending a subset of items and as a result increases the items distribution long tail issues. Furthermore, there is a risk of irritating the user by providing many recommended items when the user is not in a situation to accept many recommendations and make purchases and as a result it will affect their decision to make a purchase

negatively (Adomavicius et al., 2018).

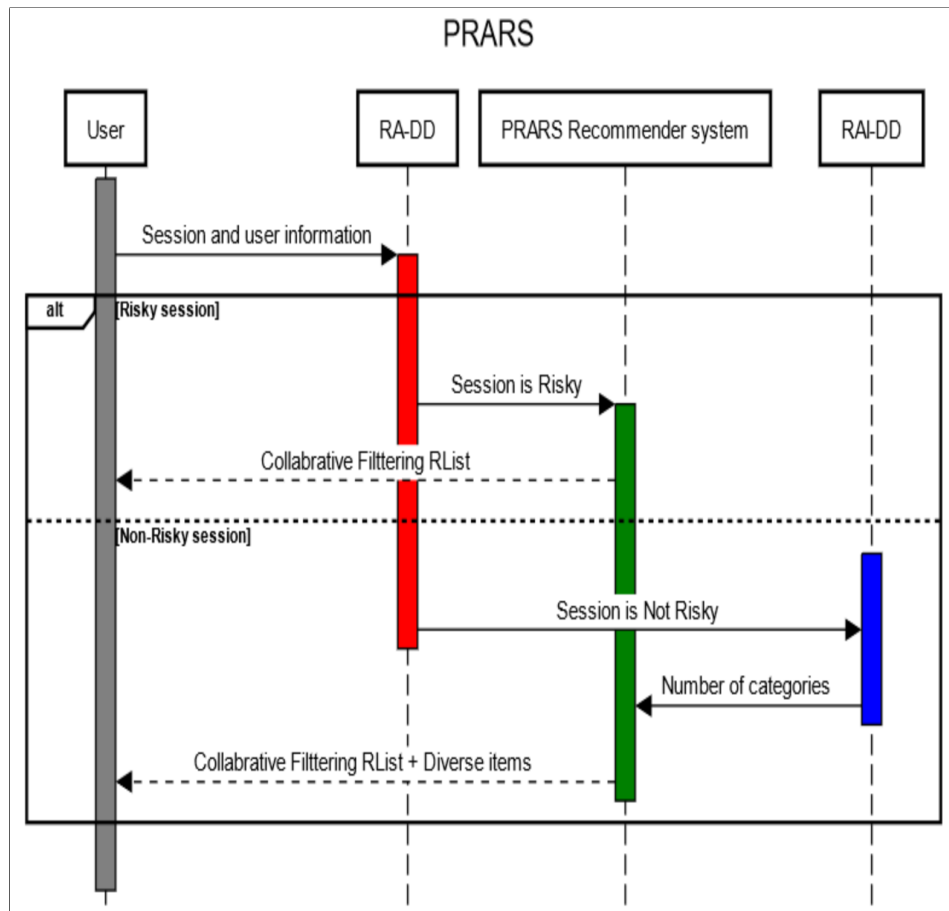


Figure 6.1: Personalized Risk Aware Recommender System sequence diagram

The proposed framework, Personalized Risk Aware Recommender System (PRARS), depends on two risk assessment models: risk assessment of user not purchasing and risk assessment of recommending a narrow range of items. The goal of the former risk assessment model is to find if the session will contain purchase. The aim of the proposed framework is to increase diversity in the recommendation list when the session is predicted to contain a purchase. The reason for increasing diversity for those sessions is to reduce bias towards popular items and increase the chance of making a purchase. The risk assessment of recommending a narrow range of items is applied to suggest the number of different categories of items the framework will present to the user. The goal of this assessment is to find the level of diversity that the user wants to see.

The session that is predicted as high risk will receive only personalized items, to reduce irritating the user with diverse items. The session that is predicted as low risk will receive diverse items recommendations. Furthermore, personalized items will be recommended to make sure that the user is receiving recommendations based on their preferences similar to those how are in a high-risk session.

The recommender system framework will inject diverse items from different categories in the recommendation list which will increase diversity in the recommendation list and as a result will decrease the overfitting issue of users preferences or bias towards popular items. The proposed framework builds the diverse items list based on the distribution of items and include items from the long tail distribution of items. For each category, the proposed framework will find the items that have minimum interactions based on the Split-Apply-Combine Strategy (Wickham et al., 2011) and the pseudocode of Diverse items list is shown in Algorithm 3.

Algorithm 3: Diverse items List (DI)

1. **Input:** item i ; Purchase of item i_p ; Category of item $i_{category}$; List of all items I
 2. For each $i \in I$:
 - (a) $(i_{p.count}) = \text{Count}(i_p)$
 3. Split $(I, i_{category})$
 4. For each $i_{category} \in I$:
 - (a) $\text{Min}(i_{p.count}) = \text{Get-Minimum}(i_{category}, i_{p.count})$
 5. $\text{DI} = \text{Combine}(i_{category}, \text{Min}(i_{p.count}))$
 6. **Output:** DI
-

The goal of the proposed framework is to increase the coverage of items and decrease the long tail issue. Figure 6.1, shows the sequence diagram of the per-

sonalized risk aware recommender system framework, PRARS. The proposed framework will build personalized recommendations based on the collaborative filtering approach. There are two kinds of collaborative filtering approaches: user-to-user collaborative filtering and item-to-item collaborative filtering. The proposed framework is based on item-to-item collaborative filtering to reduce the dimensionality of the user-to-user collaborative filtering approach, because it does not require any changes in matrix dimension when there is a new user which is supposed to be more often than new items.

The PRARS will be based on user's purchases which means that if items i and j have been purchased by different users this means that item i is similar to item j and the user who prefers items i will most likely prefer item j . To find similarities between items, we first build a user-item matrix, then an item-item similarity matrix. The user-item matrix is to find whether a user makes purchases of those items or not where rows represent user IDs and columns represent item IDs. Then, based on this information, we start to build the item-item similarity matrix which is used to find the relation between items. To find similarities between items i and j , firstly we find all users who have bought both items. Then, we measure the similarity between i and j , which have been bought by users a and b by using the Cosine similarity. The Cosine similarity is more suitable for the item based collaborative filtering (Jannach et al., 2011). We build two item-vectors, p_i for item i and p_j for item j , in the user space of (a, b) and find the cosine angle between these vectors, by using the Cosine similarity metric (Sarwar et al., 2001), and it is calculated as follows:

$$\mathbf{w}_{ij} = \text{sim}(i, j) \quad (6.1)$$

$$= \cos(p_i \cdot p_j) \quad (6.2)$$

$$= \frac{p_i \cdot p_j}{\|p_i\| \cdot \|p_j\|} \quad (6.3)$$

If the cosine is equal to one this means that the angle between vectors is equal to zero and the items are totally similar, and if the cosine is equal to zero this means that the angle between vectors is equal to 90 degrees and the items are not similar. For each item, the score is going to be calculated based on similar items that the user has already bought. To calculate the aggregation score for user u and item i (Sarwar et al., 2001):

$$\mathbf{s}(\mathbf{u}, \mathbf{i}) = \frac{\sum_{j \in I_u} W_{ij} \cdot p_{uj}}{\sum_{j \in I_u} |W_{ij}|} \quad (6.4)$$

where W_{ij} is the similarity between the item i and item j . Because p is equal to zero or one, all the scores will be zero or one. Then it will not be useful to distinguish between relevancy of items. So, the aggregation score becomes:

$$\mathbf{s}(\mathbf{u}, \mathbf{i}) = \sum_{j \in I_u} W_{ij} \quad (6.5)$$

The proposed framework, personalized risk aware recommender system (PRARS), provides recommendations based on assessing the risk of the user not purchasing and assessing the risk of the user not interacting with system. The pseudocode of PRARS is shown in Algorithm 4.

6.4 Experiments and Results

In this section we conduct experiments comparing the proposed framework with other recommender system approaches which are the item-to-item collaborative filtering recommender system and the popularity based recommender system. The evaluation is based on the accuracy through precision and recall, diversity through coverage of items, novelty through long tail distribution of items, and complexity.

Algorithm 4: Personalized Risk Aware Recommender System (PRARS)

1. **Input:** User session $US = \{(x_1, y_1, z_1), \dots, (x_m, y_m, z_m)\}$; Number of items to be recommended N ; item i ; List of all items I ; List of diverse items DI ;
 2. For $u = 1$ to m :
 - (a) $y_x \leftarrow RA - DD(u)$
 - (b) For each $i \in I$
 - i. Compute $s(u, i)$
 - ii. Descending.sort($s(u, i), I$)
 - (c) If $y_u = \text{risky}$:
 - i. $RL = \text{top}(N, I)$
 - (d) Else
 - i. $z_u \leftarrow RAI - DD(u)$
 - ii. $RL = \text{top}(N - z_u, I) + \text{random}(z_u, DI)$
 3. **Output:** Recommendation list RL
-

6.4.1 Experimental setup

The datasets that are used in the experiments are the Retail Rocket recommender system dataset (Ret, 2017) and the Online Retail dataset (Chen et al., 2012), which are publicly available. To build an effective prediction model at least three clicks are required (Ding et al., 2015) so, a pre-processing step was been done and only the sessions that had at least three clicks are included. The Retail Rocket recommender system dataset contains 91,870 sessions. The Online Retail dataset contains 18,535 sessions (Chen et al., 2012). The hardware and software environment that have been used for the experiments in Chapter 3, have also been used in these experiments. Source code of PRARS is available for download ⁴. Table 6.1, contains statistics of the datasets.

There are some studies that argue that the accuracy of the recommender system model is not enough and they suggest metrics that are used to evaluate

⁴Available at <https://github.com/DanahAG/PRARS>

Table 6.1: Statistics of Datasets

Dataset	No. of sessions	No. of items	No. of users
Retail Rocket	91,870	243	26609
Online Retail	18,535	395	4373

recommender systems based on the diversity and the novelty of recommendation list (Adomavicius and Tuzhilin, 2005; Herlocker et al., 2004; McNee et al., 2006). Diversity can be assessed based on the coverage of items (Robillard and Walker, 2014; Shlomo et al., 2018). Novelty can be assessed based on the long tail distribution of items (Anderson, 2006; Brynjolfsson et al., 2010). The evaluation of the proposed framework is going to be based on the precision, or the ratio of recommended items that are relevant to the user, recall is the percentage of items that a user buys that are recommended, and the coverage of the items is the ratio of items in training sets the framework can recommend in the recommendation lists. To show the effectiveness of the proposed framework (PRARS), we compared it with the item-to-item collaborative filtering recommender system and the popularity based recommender system.

$$\mathbf{Recall} = \frac{\text{Number of recommended items that are relevant}}{\text{Total number of relevant items}} \quad (6.6)$$

$$\mathbf{Precision} = \frac{\text{Number of recommended items that are relevant}}{\text{Number of recommended items}} \quad (6.7)$$

$$\mathbf{Coverage} = \frac{\text{Number of recommended items}}{\text{Number of items in the training set}} \quad (6.8)$$

6.4.2 Experimental result

To measure the accuracy of the model we are using precision and recall of the proposed framework PRARS, and compare the results with the item-to-item collaborative filtering recommender system and the popularity based recommender system. Figure 6.2, shows that the proposed model has less precision than the item to item collaborative filtering recommender system when we increase the top N recommendations. The precision ratio of the proposed model underperforms the item to item collaborative filtering recommender system by 0.91% of the Retail Rocket dataset and 0.01% of the Online Retail dataset at the top ten recommendations. The reason behind this is that the proposed framework is adding more diverse items to the end of the recommendation list when the session is not risky, or the user's purchase intention is high. It is expected that diversity can affect the accuracy of the model negatively (Adomavicius and Kwon, 2008). The precision of the proposed model outperforms the popularity based recommender system by 2.95% of the Retail Rocket dataset and 0.34% of the Online Retail dataset at the top ten recommendations. The popularity based recommender system has the lowest precision compared with the proposed framework and the item-to-item collaborative filtering recommender system. The reason behind these results is that the popularity based recommender system does not measure similarity between items, it just recommends the most popular items that are shown on the training set. The item to item collaborative filtering recommender system has the highest precision compared with other recommender systems which are the proposed framework and popularity based recommender systems. The reason behind this is that the item-to-item collaborative filtering recommender system is building the recommendation list based on similarities between items that the users bought in the training sets which increases the chance of predicting the items that the user will purchase.

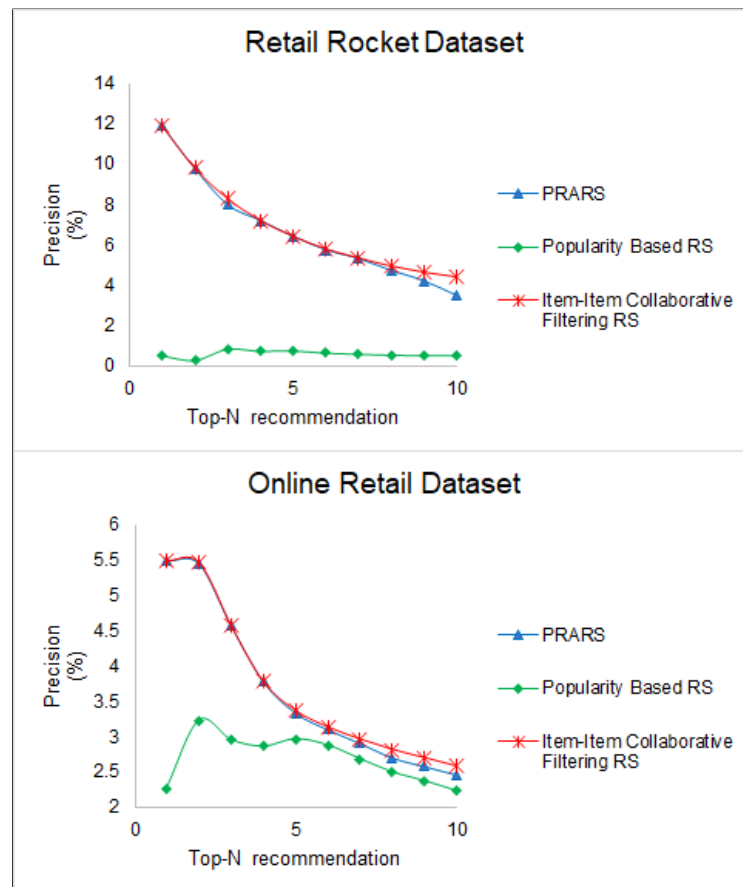


Figure 6.2: Precision of PRARS, Item-to-Item Collaborative Filtering and Popularity based recommender systems

Figure 6.3 shows the recall of the proposed framework compared with other recommender system approaches. The recall of the proposed framework underperforms the item-to-item collaborative filtering recommender system by 1% of the Retail Rocket dataset and 0.54% of the Online Retail dataset at the top ten recommendations. The proposed model has less recall than the item to item collaborative filtering recommender system when we increase the top N recommendations. The reason behind this is the proposed framework is increasing the diversity of the items at the end of the recommendation list when the risk of the user not purchasing is low or the user's purchase intention is high. The popularity based recommender system has the lowest recall compared with the proposed framework and the item-to-item collaborative fil-

tering recommender system. The reason behind these results is the popularity based recommender system provides recommendation lists based on the popularity of items in the training set without calculating the similarities between items. The item-to-item collaborative filtering recommender system has the highest recall compared with other recommender systems which are the proposed framework and the popularity based recommender system because the item-to-item collaborative filtering recommender system is building the recommendation list based on similarities between items that the users bought in the training sets which positively influences the accuracy of the recommender system.

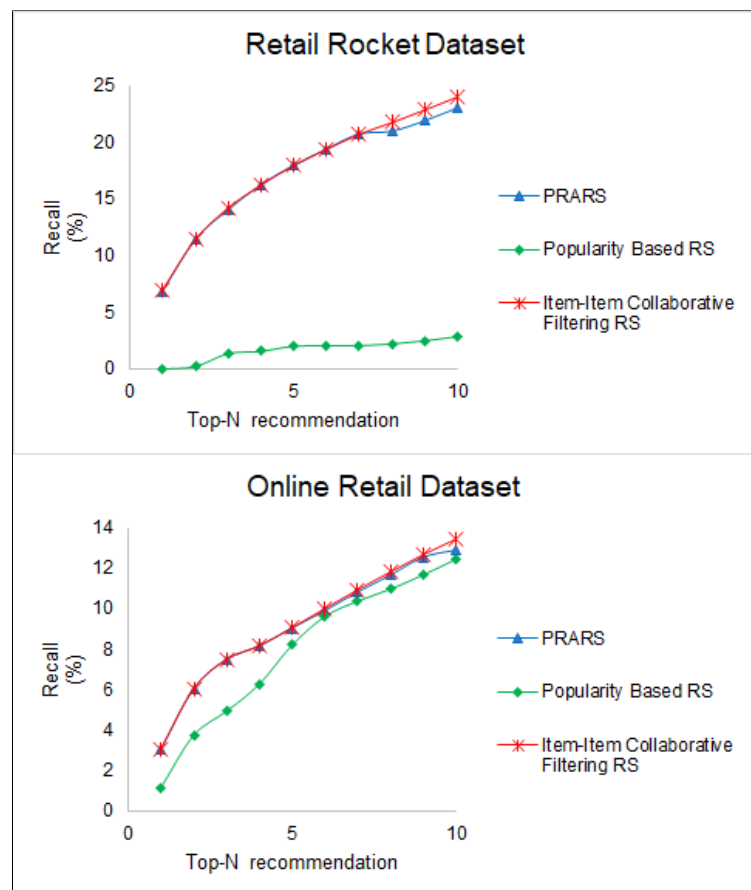


Figure 6.3: Recall of PRARS, Item-to-Item Collaborative Filtering and Popularity based recommender systems

The second experiment is to evaluate the aggregate diversity of the framework.

The aggregate diversity of the model can be measured based on the coverage of the items (Robillard and Walker, 2014; Shlomo et al., 2018). The coverage of the items is used to show how many items from the training set the recommender system can include in the recommendation list. Figure 6.4 shows a comparison of the proposed framework PRARS and two recommender system approaches which are the item-to-item collaborative filtering recommender system and the popularity based recommender system. The proposed framework outperforms the item-to-item collaborative filtering recommender system and the popularity based recommender system by 8% and 84% in the Retail Rocket dataset, respectively, when the top N recommendations is equal to 10 and the coverage of the proposed framework is 89.9%. For the Online Retail dataset the proposed framework outperforms the item to item collaborative filtering recommender system and the popularity based recommender system by 10% and 63% respectively, when the top N recommendations is equal to 10 and the percentage of coverage of proposed framework is 70.2 %. The proposed framework can include more items from the training set in both datasets which means that the proposed framework can increase the aggregate diversity in the recommendation lists when the users sessions are not risky or the user purchase intention is high.

The third experiment is to evaluate the novelty of the proposed framework and other recommender system approaches. Novelty in recommendation is a major factor when there are a few items which are extremely popular, and the rest of them are not recommended to the user (Anderson, 2006). The novelty can be assessed by the long tail distribution of items (Anderson, 2006; Brynjolfsson et al., 2010). Increasing the number of the items in the head of items distribution means more items are becoming more popular and included in the recommendation list. To find how many items are in the head of items distri-

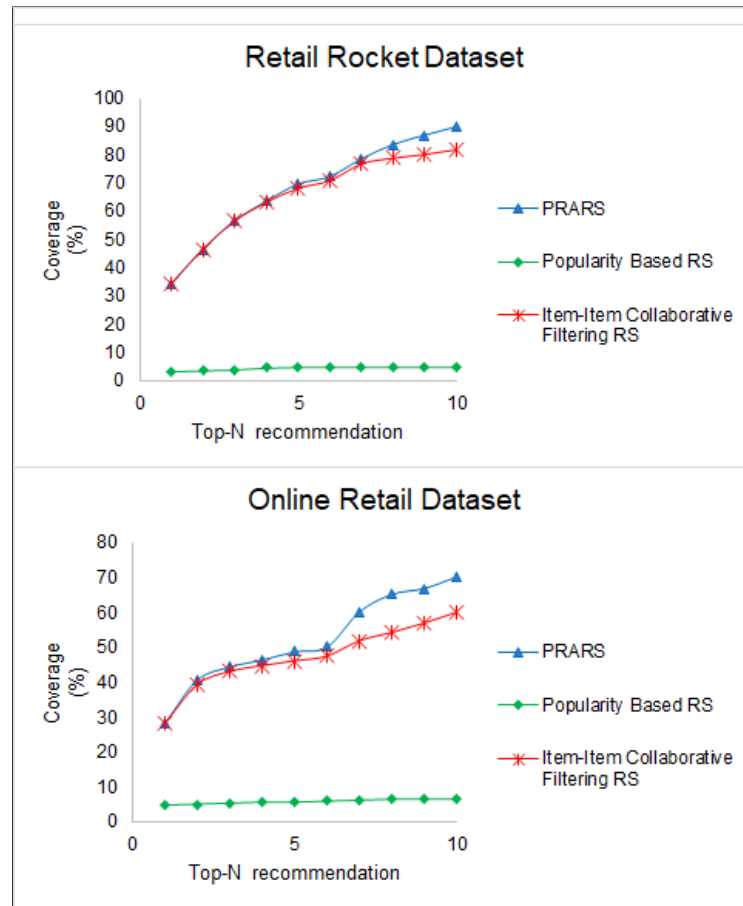


Figure 6.4: Coverage of PRARS, Item-to-Item Collaborative Filtering and Popularity based recommender systems

bution, the cut off is selected based on the areas of distribution, both the head (popular items) and the long tail (un-popular items) areas are equal. Figure 6.5 shows the long tail of items distribution of the proposed framework PRARS and the two recommender system approaches. The proposed framework head includes 11 items while the item-to-item collaborative filtering recommender system includes 10 items and the popularity based recommender system includes 6 items for the Retail Rocket dataset. For the Online Retail dataset, the proposed framework head includes 16 items while the item-to-item collaborative filtering recommender system includes 14 items and the popularity based recommender system includes 7 items. The proposed framework outperforms other approaches in the novelty because it includes more items in the head of the items distribution. Including more items in the head of the items distribu-

tion of the proposed framework depends on the risk assessment of the user not purchasing and the risk assessment of the user not interacting with the system. Lower risk means items from different categories can be recommended and as a result can increase the number of items in the head of the items distribution.

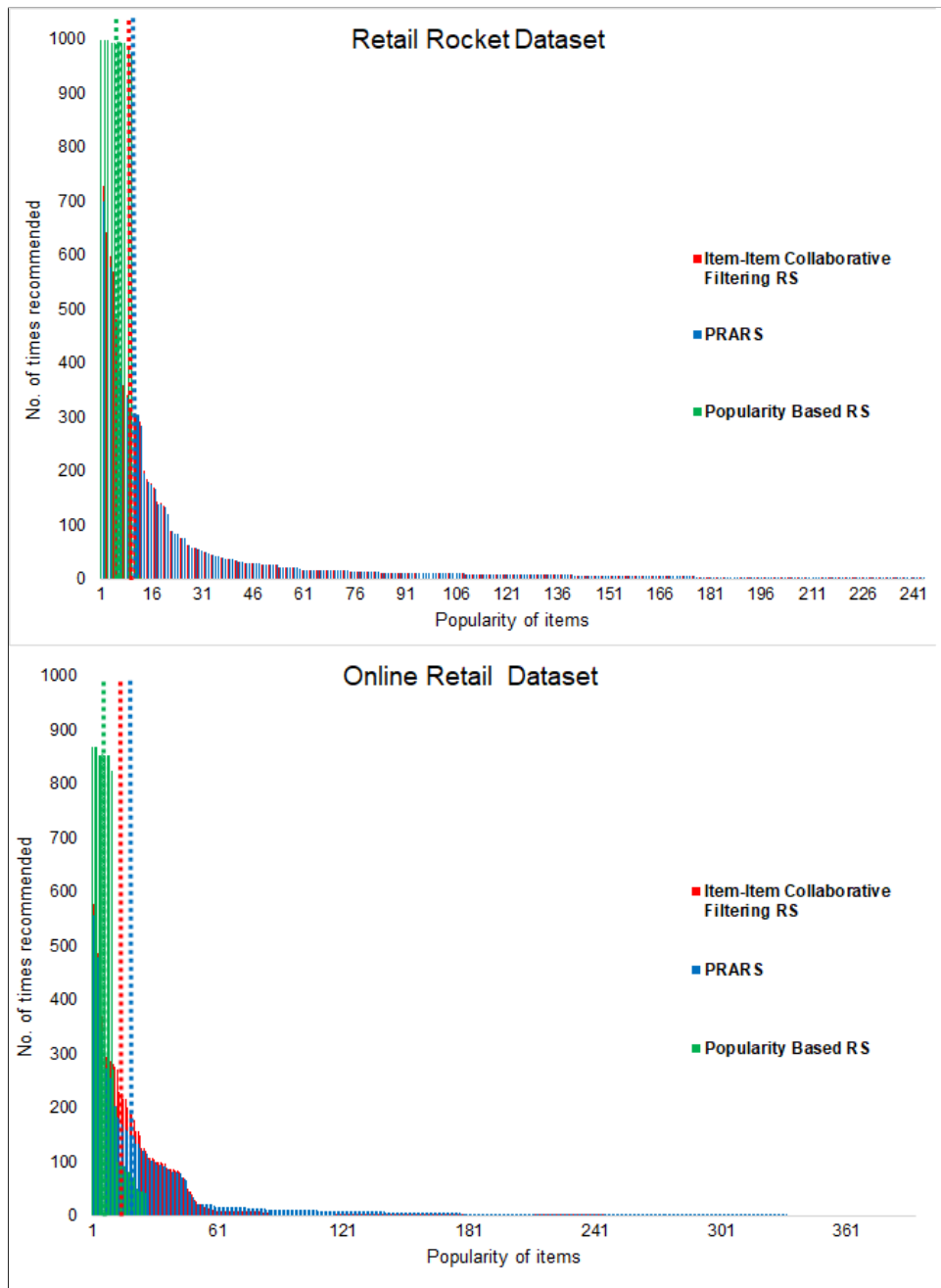


Figure 6.5: long tail of items distribution of the proposed framework (PRARS), the Item-to-Item Collaborative Filtering recommender system and the Popularity based recommender system

The fourth experiment is to compare the complexity of the proposed framework with other recommender system approaches. Figure 6.6 shows the comparison of the proposed framework PRARS and the two recommender system approaches. The proposed framework requires more than one millisecond to build a recommendation list for 100 sessions and more than 16.26 milliseconds to build a recommendation list for 1000 sessions of the Retail Rocket dataset. The proposed framework requires more than 0.78 milliseconds to build recommendation lists for 100 sessions and more than 13 milliseconds to build recommendation lists for 1000 sessions of the Online Retail dataset. The item-to-item collaborative filtering recommender system outperforms the proposed framework by 0.16 milliseconds to build recommendation lists for 100 sessions and by 3.4 milliseconds to build recommendation lists for 1000 sessions for the Online Retail dataset. For the Online Retail dataset, the item-to-item collaborative filtering recommender system outperforms the proposed framework by 0.04 milliseconds to build recommendation lists for 100 sessions and by 3.3 milliseconds to build recommendation lists for 1000 sessions. The reason behind this is that the proposed framework performs a risk assessment of the user not purchasing and a risk assessment of the user not interacting with the system. The complexity of risk assessment of the user not purchasing model and the complexity of the user not interacting with the system model are based on the Adaptive Boosting. The Adaptive Boosting complexity is $O(\log(n) \times f \times m)$ where n is the number of sessions, f is the number of features and m is the number of iterations. The item-to-item collaborative filtering recommender system requires a similarity matrix to provide recommendations while the proposed framework requires a similarity matrix as well as risk assessment models. The popularity based recommender system outperforms the item-to-item collaborative filtering recommender system and the proposed framework required one millisecond to build recommendation lists for 100 sessions and 10 millise-

onds to build recommendation lists for 1000 sessions for the Retail Rocket dataset and for the Online Retail dataset set it requires 0.70 milliseconds to build recommendation lists for 100 sessions and more than 7.35 milliseconds to build recommendation lists for 1000 sessions. The reason behind this is that the popularity based recommender system does not perform a similarity matrix to provide recommendation lists to users which means less processing time compared with other recommender system approaches which perform a similarity matrix to provide recommendation lists.

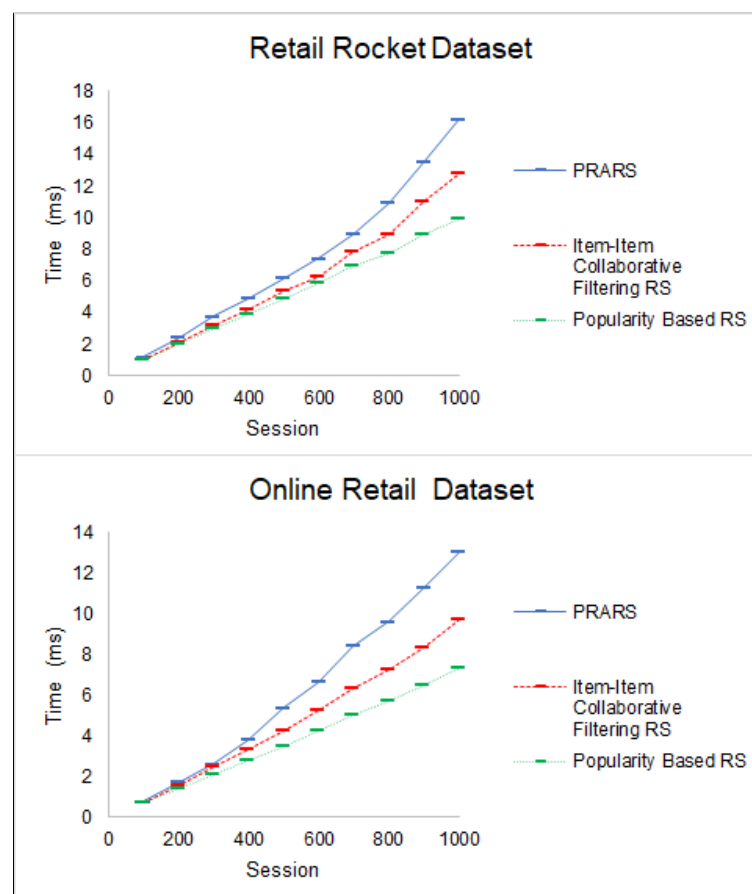


Figure 6.6: Time to provide recommendation lists of the proposed model, the Item-to-Item Collaborative Filtering recommender system and the Popularity based recommender system

The proposed framework, PRARS, outperforms the item to item collaborative filtering recommender system and the popularity based recommender system

regarding diversity and novelty. However, the proposed framework underperforms the item-to-item collaborative filtering recommender system regarding precision, recall and processing time. Furthermore, the proposed framework underperforms the popularity based recommender system regarding processing time only.

6.5 Conclusion

In this chapter, we proposed a recommender system framework, Personalized Risk Aware Recommender System, to increase aggregate diversity in recommendation lists based on the risk of the user not purchasing and the risk of recommending a narrow range of items. The goal was to increase aggregate diversity when the user has high purchase intention and decrease irritating the user by applying the item-to-item collaborative filtering approach when the risk of the user not purchasing is high. The number of diverse items that the proposed framework can include in the recommendation list will depend on the risk of the user not interacting with the system because different users accept different levels of diversity in their recommendation lists. Experiments have shown the effectiveness of the proposed framework in increasing the aggregate diversity of the recommendation list compared with the item to item collaborative filtering recommender system and the popularity based recommender system. The proposed framework has the highest aggregate diversity and outperforms the item-to-item collaborative filtering recommender system coverage by 8% and the popularity based recommender system by 84% in the Retail Rocket dataset and the coverage of proposed model is 89.9%. For the Online Retail dataset the proposed model outperforms the item-to-item collaborative filtering recommender system by 10% and the popularity based recommender system by 63% and the coverage of the proposed framework is 70.2%. Further-

more, the experiment shows that the proposed framework can increase novelty in recommendation lists which is evaluated by the long tail distribution of items and the proposed framework outperforms the item-to-item collaborative filtering recommender system and the popularity based recommender system. However, the proposed framework has some limitations. The first limitation is higher complexity or processing time compared with other recommender systems and the reason behind that is that the proposed framework assesses the risk of the user not purchasing and the risk of recommending a narrow range of items before providing recommendations to the user which is a result of increased complexity. The second limitation is that the proposed framework has lower precision and recall rates compared with the item-to-item collaborative filtering recommender system. In the next chapter we will conclude the thesis and discuss future directions.

7

Conclusions

In this thesis we proposed a framework for a recommender system that comprises risk assessment models to consider risks before providing recommendation lists. These risks are assessed to consider users' behaviours that may influence the recommender system goal: saving the user time and effort. The proposed framework of the recommender systems is awareness of two types of risk: the risk of the user not purchasing and the risk of recommending a narrow range of items. We performed two approaches of risk assessment: a model driven approach and a data driven approach. The proposed framework can influence diversity in recommendation lists positively and decreases issues of the long tail distribution of items.

7.1 Contributions

The thesis contributions are as follows:

- **RA-MD**, a risk assessment model, which assesses the risk of a user not purchasing based on user behaviour and population behaviour. The proposed model assessed the risk of a user not purchasing by using a concept of model driven approach which is the Constructive Cost Model and classifies sessions to a risky session, a session that is predicted to not contain a purchase, and a non-risky session, a session that is predicted to contain a purchase.
- **RA-DD**, an algorithm that assesses the risk of a user not making a purchase by using a concept of data driven approach which is the Adaptive Boosting algorithm, to predict and classify sessions to a risky and non-risky session. This algorithm assesses the risk of a user not purchasing and adapts to the change of users' behaviours.
- **RAI-DD**, an algorithm that assesses the risk of a user not interacting by using a concept of data driven approach which is the Stagewise Additive Modelling using a Multi-class Exponential loss function Algorithm, to predict how many different categories of items the users might explore during a session. The goal is to find the optimal level of diversity.
- **PRARS**, we introduced a recommender system framework, Personalized Risk Aware Recommender System, a recommender system framework that increases aggregate diversity in the recommendation list based on the risk of the user not purchasing and the risk of recommending a narrow range of items. The goal is to increase aggregate diversity when the risk of the user not purchasing, a non risky session, is low and decreasing irritating the user by providing personalized recommendations

by applying an item-to-item collaborative filtering approach when the risk of the user not purchasing is high. The number of diverse items that the proposed model can include in the recommendation list will depend on the risk of a user not interacting with the system, which predicts how many different categories of items the user will explore during a session.

7.2 Limitations

This thesis has proposed a recommender system framework that considers two types of risks before providing recommendations. We introduced algorithms to assess risks and provided recommendation lists. However, there are some limitations that can be noted for future research:

- The proposed algorithms in Chapters 4 and 5 for assessing risks require tuning more parameters based on the change of the datasets. Applying an optimization algorithm to obtain the optimal values for the parameter could be useful taking into account the complexity of these algorithms and their influence on the recommender system scalability.
- The proposed recommender system in Chapter 6 increases the diversity of recommendation lists by providing items from different categories which are from the long tail distribution of items. However, the proposed recommender system does not perform a similarity metric to find how diverse these items are for a specific user.

7.3 Future Directions

In the thesis we proposed a framework for a risk aware recommender system that provides personalized and diverse recommendations to users and considers the risks that may affect recommender systems' goals: the user's purchase

intention and the user interaction with the system. Future directions based on this work can be categorised as research related to recommender systems mechanisms, risks that can affect recommender systems and diversity in recommender systems.

- **Recommender Systems**

We performed the recommender system framework that provides personalization based on implicit data and we applied our risk assessments on the collaborative filtering approach. There are other kinds of recommender systems that are based on the sequence of clicks from users which basically find the next item the user will probably like based on the previous clicks through the Markov Chain Model and other models for next event prediction. It would be interesting to find the relationship between risk awareness and its influence on the sequence of clicks and what are the possible methods to include risk assessments through these kinds of recommendations.

- **Risk**

We studied the risks that can influence the goals of the recommender system: low purchase intention and low interaction. There is another direction that is not related to the user centred application. Including more risks that may influence the goals of the recommender system from different stakeholders' viewpoints could be useful. Applying an optimization of risks to find the most critical risks that are required to be considered may increase the quality of the recommender systems. Furthermore, awareness of the ethics issue in purchasing implicit data that is gathered from users to understand their behaviours are debatable issues that can affect communities in general.

- **Diversity**

The lack of item diversity in a recommender system is considered a new challenge for e-commerce recommender systems. Diversity can be assessed through coverage of items and can affect the accuracy of the recommender system. However, there are no metrics that can be used to measure accuracy and diversity as a multi objective metric that considers balancing accuracy and diversity. Furthermore, in this thesis we considered the items from the long tail distribution of items which do not have a lot of interaction in the training sets as diverse items. Studying the items' features and their similarities with other items for specific users may improve the quality of diversity; however, this will definitely increase the complexity of the system. Finding the trade-off between the quality and the complexity of providing items diversity is also an interesting direction.

Bibliography

- (2017). Retailrocket recommender system dataset. <https://www.kaggle.com/retailrocket/ecommerce-dataset>. Accessed: 2018-05-3.
- Abbassi, Z., Amer-Yahia, S., Lakshmanan, L. V., Vassilvitskii, S., and Yu, C. (2009). Getting recommender systems to think outside the box. In *Proceedings of the 3rd ACM Conference on Recommender Systems*, pages 285–288.
- Adomavicius, G., Bockstedt, J. C., Curley, S. P., and Zhang, J. (2018). Effects of online recommendations on consumers’ willingness to pay. *Information Systems Research*, 29(1):84–102.
- Adomavicius, G. and Kwon, Y. (2008). Overcoming accuracy-diversity tradeoff in recommender systems: A variance-based approach. In *Proceedings of Workshop on Information Technologies and Systems*, volume 8. Citeseer.
- Adomavicius, G. and Kwon, Y. (2011). Improving aggregate recommendation diversity using ranking-based techniques. *IEEE Transactions on Knowledge and Data Engineering*, 24(5):896–911.
- Adomavicius, G. and Tuzhilin, A. (2005). 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.

- Altman, E. I., Marco, G., and Varetto, F. (1994). Corporate distress diagnosis: Comparisons using linear discriminant analysis and neural networks (the italian experience). *Journal of Banking & Finance*, 18(3):505–529.
- Anderson, C. (2006). *The long tail: Why the future of business is selling less of more*. Hachette Books.
- Armstrong, J. S., Morwitz, V. G., and Kumar, V. (2000). Sales forecasts for existing consumer products and services: Do purchase intentions contribute to accuracy? *International Journal of Forecasting*, 16(3):383–397.
- Arnold, J. F. and Voss, L. L. (2004). System and method for incorporating concept-based retrieval within boolean search engines. US Patent 6,745,161.
- Ba, S. and Pavlou, P. A. (2002). Evidence of the effect of trust building technology in electronic markets: Price premiums and buyer behavior. *MIS quarterly*, pages 243–268.
- Bag, S., Tiwari, M. K., and Chan, F. T. (2019). Predicting the consumer’s purchase intention of durable goods: An attribute-level analysis. *Journal of Business Research*, 94:408–419.
- Bagchi, S. (2015). Performance and quality assessment of similarity measures in collaborative filtering using mahout. *Procedia Computer Science*, 50:229–234.
- Barragáns-Martínez, A. B., Costa-Montenegro, E., Burguillo, J. C., Rey-López, M., Mikic-Fonte, F. A., and Peleteiro, A. (2010). A hybrid content-based and item-based collaborative filtering approach to recommend tv programs enhanced with singular value decomposition. *Information Sciences*, 180(22):4290–4311.

- Basiri, J., Shakery, A., Moshiri, B., and Hayat, M. Z. (2010). Alleviating the cold-start problem of recommender systems using a new hybrid approach. In *2010 5th International Symposium on Telecommunications*, pages 962–967. IEEE.
- Bedi, P., Vashisth, P., Khurana, P., et al. (2013). Modeling user preferences in a hybrid recommender system using type-2 fuzzy sets. In *2013 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–8. IEEE.
- Bifet, A. and Gavalda, R. (2006). Kalman filters and adaptive windows for learning in data streams. In *International Conference on Discovery Science*, pages 29–40. Springer.
- Billsus, D., Pazzani, M. J., and Chen, J. (2000). A learning agent for wireless news access. In *Proceedings of the 5th International Conference on Intelligent User Interfaces*, pages 33–36.
- Boehm, B. W. and DeMarco, T. (1997). Software risk management. *IEEE Software*, (3):17–19.
- Boratto, L., Fenu, G., and Marras, M. (2021). Connecting user and item perspectives in popularity debiasing for collaborative recommendation. *Information Processing & Management*, 58(1):102387.
- Borgonovo, E. et al. (2017). Sensitivity analysis. *Number*, 251:93–100.
- Bouneffouf, D., Bouzeghoub, A., and Ganarski, A. L. (2013). Risk-aware recommender systems. In *International Conference on Neural Information Processing*, pages 57–65. Springer.
- Bradley, K. and Smyth, B. (2001). Improving recommendation diversity. In *Proceedings of the 12th Irish Conference on Artificial Intelligence and Cognitive Science, Maynooth, Ireland*, pages 85–94.

- Breiman, L. (1996). Bagging predictors. *Machine learning*, 24(2):123–140.
- Brynjolfsson, E., Hu, Y., and Smith, M. D. (2010). Research commentary—long tails vs. superstars: The effect of information technology on product variety and sales concentration patterns. *Information Systems Research*, 21(4):736–747.
- CarlKadie, J. B. D. (1998). Empirical analysis of predictive algorithms for collaborative filtering. *Microsoft Research Microsoft Corporation One Microsoft Way Redmond, WA, 98052*.
- Caruso, W., Corsi, A. M., Bogomolova, S., Cohen, J., Sharp, A., Lockshin, L., and Tan, P. J. (2018). The real estate value of supermarket endcaps: Why location in-store matters. *Journal of Advertising Research*, 58(2):177–188.
- Chau, P. Y., Ho, S. Y., Ho, K. K., and Yao, Y. (2013). Examining the effects of malfunctioning personalized services on online users’ distrust and behaviors. *Decision Support Systems*, 56:180–191.
- Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). Smote: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357.
- Chawla, N. V., Japkowicz, N., and Kotcz, A. (2004). Special issue on learning from imbalanced data sets. *ACM SIGKDD Explorations Newsletter*, 6(1):1–6.
- Chawla, N. V., Lazarevic, A., Hall, L. O., and Bowyer, K. W. (2003). Smoteboost: Improving prediction of the minority class in boosting. In *European Conference on Principles of Data Mining and Knowledge Discovery*, pages 107–119. Springer.

- Chen, D., Sain, S. L., and Guo, K. (2012). Data mining for the online retail industry: A case study of rfm model-based customer segmentation using data mining. *Journal of Database Marketing & Customer Strategy Management*, 19(3):197–208.
- Chen, P.-Y., Chou, Y.-C., and Kauffman, R. J. (2009). Community-based recommender systems: Analyzing business models from a systems operator’s perspective. In *2009 42nd Hawaii International Conference on System Sciences*, pages 1–10. IEEE.
- Cheng, J., Lo, C., and Leskovec, J. (2017). Predicting intent using activity logs: How goal specificity and temporal range affect user behavior. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 593–601.
- Choi, S. H., Jeong, Y.-S., and Jeong, M. K. (2010). A hybrid recommendation method with reduced data for large-scale application. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 40(5):557–566.
- Chughtai, M. W., Selamat, A., Ghani, I., and Jung, J. J. (2014). Retracted: E-learning recommender systems based on goal-based hybrid filtering. *International Journal of Distributed Sensor Networks*, 10(7):912130.
- Council, N. R. et al. (2005). *The owner’s role in project risk management*. National Academies Press.
- Crouhy, M., Galai, D., and Mark, R. (2005). *The Essentials of Risk Management*. McGraw Hill Professional.
- Cummins, R. (2008). The evolution and analysis of term-weighting schemes in information retrieval. *National University of Ireland*.

- Curtis, P. and Carey, M. (2012). Risk assessment in practice. <https://www2.deloitte.com/ng/en/pages/governance-risk-and-compliance/articles/risk-assessment-in-practice.html>. Accessed: 2018-02-20.
- Dali, A. and Lajtha, C. (2012). Iso 31000 risk management—“the gold standard”. *EDPACS*, 45(5):1–8.
- Davidson, J., Liebald, B., Liu, J., Nandy, P., Van Vleet, T., Gargi, U., Gupta, S., He, Y., Lambert, M., Livingston, B., et al. (2010). The youtube video recommendation system. In *Proceedings of the 4th ACM Conference on Recommender Systems*, pages 293–296.
- De Campos, L. M., Fernández-Luna, J. M., Huete, J. F., and Rueda-Morales, M. A. (2010). Combining content-based and collaborative recommendations: A hybrid approach based on bayesian networks. *International Journal of Approximate Reasoning*, 51(7):785–799.
- Deng, Y., Wu, Z., Tang, C., Si, H., Xiong, H., and Chen, Z. (2010). A hybrid movie recommender based on ontology and neural networks. In *2010 IEEE/ACM Int’l Conference on Green Computing and Communications & Int’l Conference on Cyber, Physical and Social Computing*, pages 846–851. IEEE.
- Ding, A. W., Li, S., and Chatterjee, P. (2015). Learning user real-time intent for optimal dynamic web page transformation. *Information Systems Research*, 26(2):339–359.
- Domingos, P. (1999). Metacost: A general framework for making classifiers cost-sensitive. In *ACM KDD Conference*, pages 155–164.
- Drummond, C., Holte, R. C., et al. (2003). C4. 5, class imbalance, and cost sensitivity: Why under-sampling beats over-sampling. In *Workshop on Learn-*

- ing from Imbalanced Datasets II*, volume 11, pages 1–8. Washington DC: Citeseer.
- Elkan, C. (2001). The foundations of cost-sensitive learning. In *International Joint Conference on Artificial Intelligence*, volume 17, pages 973–978. Lawrence Erlbaum Associates Ltd.
- Estabrooks, A., Jo, T., and Japkowicz, N. (2004). A multiple resampling method for learning from imbalanced data sets. *Computational Intelligence*, 20(1):18–36.
- Fleder, D. and Hosanagar, K. (2009). Blockbuster culture’s next rise or fall: The impact of recommender systems on sales diversity. *Management Science*, 55(5):697–712.
- Freund, Y. and Schapire, R. E. (1995). A decision-theoretic generalization of on-line learning and an application to boosting. In *European Conference on Computational Learning Theory*, pages 23–37. Springer.
- Friedman, J., Hastie, T., Tibshirani, R., et al. (2000). Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). *The Annals Of Statistics*, 28(2):337–407.
- Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., and Herrera, F. (2011). A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(4):463–484.
- Gama, J., Medas, P., Castillo, G., and Rodrigues, P. (2004). Learning with drift detection. In *Brazilian Symposium On Artificial Intelligence*, pages 286–295. Springer.

- Garcia-Valdez, M., Alanis, A., and Parra, B. (2010). Fuzzy inference for learning object recommendation. In *International Conference on Fuzzy Systems*, pages 1–6. IEEE.
- Ghazanfar, M. and Prugel-Bennett, A. (2010). Building switching hybrid recommender system using machine learning classifiers and collaborative filtering. *IAENG International Journal of Computer Science*, 37(3).
- Gomez-Uribe, C. A. and Hunt, N. (2015). The netflix recommender system: Algorithms, business value, and innovation. *ACM Transactions on Management Information Systems (TMIS)*, 6(4):1–19.
- Guo, H. and Viktor, H. L. (2004). Learning from imbalanced data sets with boosting and data generation: the databoost-im approach. *ACM SIGKDD Explorations Newsletter*, 6(1):30–39.
- Han, H., Wang, W.-Y., and Mao, B.-H. (2005). Borderline-smote: a new over-sampling method in imbalanced data sets learning. In *International Conference on Intelligent Computing*, pages 878–887. Springer.
- Hao, J., Zhao, T., Li, J., Dong, X. L., Faloutsos, C., Sun, Y., and Wang, W. (2020). P-companion: A principled framework for diversified complementary product recommendation. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 2517–2524.
- He, H. and Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1263–1284.
- He, H. and Ma, Y. (2013). *Imbalanced learning: foundations, algorithms, and applications*. John Wiley & Sons.
- Herlocker, J., Konstan, J. A., and Riedl, J. (2002). An empirical analysis

- of design choices in neighborhood-based collaborative filtering algorithms. *Information Retrieval*, 5(4):287–310.
- Herlocker, J. L., Konstan, J. A., Terveen, L. G., and Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1):5–53.
- Huang, F. J., Zhou, Z., Zhang, H.-J., and Chen, T. (2000). Pose invariant face recognition. In *Proceedings 4th IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580)*, pages 245–250. IEEE.
- Hulten, G., Spencer, L., and Domingos, P. (2001). Mining time-changing data streams. In *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 97–106.
- Hurley, N. and Zhang, M. (2011). Novelty and diversity in top-n recommendation—analysis and evaluation. *ACM Transactions on Internet Technology (TOIT)*, 10(4):1–30.
- Iloiu, M., Csiminga, D., et al. (2009). Project risk evaluation methods—sensitivity analysis. *Annals of the University of Petrosani, Economics*, 9(2):33–38.
- ISO. (2009). *International Standard: Risk Management: Principles and Guidelines. ISO 31000. Principes Et Lignes Directrices*. ISO.
- Jannach, D., Zanker, M., Felfernig, A., and Friedrich, G. (2010). *Recommender systems: an introduction*. Cambridge University Press.
- Jannach, D., Zanker, M., Felfernig, A., and Friedrich, G. (2011). An introduction to recommender systems. *New York: Cambridge*.

- Jiang, Y., Shang, J., Liu, Y., and May, J. (2015). Redesigning promotion strategy for e-commerce competitiveness through pricing and recommendation. *International Journal of Production Economics*, 167:257–270.
- Kagermann, H., Wahlster, W., and Helbig, J. (2013). Securing the future of german manufacturing industry: Recommendations for implementing the strategic initiative industrie 4.0. *Final report of the Industrie*, 4(0).
- Kansala, K. (1997). Integrating risk assessment with cost estimation. *IEEE Software*, 14(3):61–67.
- Kapoor, K., Kumar, V., Terveen, L., Konstan, J. A., and Schrater, P. (2015). ” i like to explore sometimes” adapting to dynamic user novelty preferences. In *Proceedings of the 9th ACM Conference on Recommender Systems*, pages 19–26.
- Karimova, F. (2016). A survey of e-commerce recommender systems. *European Scientific Journal*, 12(34):75–89.
- Kaur, H., Pannu, H. S., and Malhi, A. K. (2019). A systematic review on imbalanced data challenges in machine learning: Applications and solutions. *ACM Computing Surveys (CSUR)*, 52(4):1–36.
- Klinkenberg, R. and Joachims, T. (2000). Detecting concept drift with support vector machines. In *International Conference on Machine Learning*, pages 487–494.
- Klinkenberg, R. and Renz, I. (1998). Adaptive information filtering: Learning in the presence of concept drifts. *Learning for text categorization*, pages 33–40.
- Knijnenburg, B. P., Willemsen, M. C., Gantner, Z., Soncu, H., and Newell,

- C. (2012). Explaining the user experience of recommender systems. *User Modeling and User-Adapted Interaction*, 22(4-5):441–504.
- Konstan, J. A., Miller, B. N., Maltz, D., Herlocker, J. L., Gordon, L. R., and Riedl, J. (1997). Grouplens: applying collaborative filtering to usenet news. *Communications of the ACM*, 40(3):77–87.
- Koren, Y., Bell, R., and Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37.
- Krogh, A. and Vedelsby, J. (1995). Neural network ensembles, cross validation, and active learning. In *Advances In Neural Information Processing Systems*, pages 231–238.
- Kubat, M., Matwin, S., et al. (1997). Addressing the curse of imbalanced training sets: one-sided selection. In *International Conference on Machine Learning*, volume 97, pages 179–186. Citeseer.
- Kukar, M., Kononenko, I., et al. (1998). Cost-sensitive learning with neural networks. In *European Conference on Artificial Intelligence*, volume 98, pages 445–449.
- Kumar, R. and Indrayan, A. (2011). Receiver operating characteristic (roc) curve for medical researchers. *Indian Pediatrics*, 48(4):277–287.
- Kunaver, M., Požrl, T., Pogačnik, M., and Tasič, J. (2007). Optimisation of combined collaborative recommender systems. *AEU-International Journal of Electronics and Communications*, 61(7):433–443.
- Kwon, H., Han, J., and Han, K. (2020). Art (attractive recommendation tailor) how the diversity of product recommendations affects customer purchase preference in fashion industry? In *Proceedings of the 29th ACM In-*

- ternational Conference on Information & Knowledge Management*, pages 2573–2580.
- Lee, C. and Lee, G. G. (2005). Probabilistic information retrieval model for a dependency structured indexing system. *Information Processing & Management*, 41(2):161–175.
- Lee, D. and Hosanagar, K. (2016). When do recommender systems work the best? the moderating effects of product attributes and consumer reviews on recommender performance. In *Proceedings of the 25th International Conference on World Wide Web*, pages 85–97.
- Lee, K. and Lee, K. (2014). Using dynamically promoted experts for music recommendation. *IEEE Transactions on Multimedia*, 16(5):1201–1210.
- Lee, S., Yang, J., and Park, S.-Y. (2004). Discovery of hidden similarity on collaborative filtering to overcome sparsity problem. In *International Conference on Discovery Science*, pages 396–402. Springer.
- Lee, T. Q., Park, Y., and Park, Y.-T. (2007). A similarity measure for collaborative filtering with implicit feedback. In *International Conference on Intelligent Computing*, pages 385–397. Springer.
- Leitch, M. (2010). Iso 31000: 2009—the new international standard on risk management. *Risk Analysis: An International Journal*, 30(6):887–892.
- Li, X. and Murata, T. (2012). Multidimensional clustering based collaborative filtering approach for diversified recommendation. In *2012 7th International Conference on Computer Science & Education (ICCSE)*, pages 905–910. IEEE.
- Linden, G., Smith, B., and York, J. (2003). Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing*, 7(1):76–80.

- Liu, X.-Y., Wu, J., and Zhou, Z.-H. (2008). Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(2):539–550.
- Lo, C., Frankowski, D., and Leskovec, J. (2016). Understanding behaviors that lead to purchasing: A case study of pinterest. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 531–540.
- Lund, M. S., Solhaug, B., and Stølen, K. (2010). *Model-driven risk analysis: the CORAS approach*. Springer Science & Business Media.
- Madachy, R. J. (1997). Heuristic risk assessment using cost factors. *IEEE Software*, 14(3):51–59.
- Maloof, M. A. (2003). Learning when data sets are imbalanced and when costs are unequal and unknown. In *ICML-2003 Workshop on Learning from Imbalanced Data Sets II*, volume 2, pages 2–1.
- Manalif, E., Capretz, L. F., and Ho, D. (2014). Fuzzy rules for risk assessment and contingency estimation within cocomo software project planning model. In *Exploring Innovative and Successful Applications of Soft Computing*, pages 88–111. IGI Global.
- Mansoury, M., Abdollahpouri, H., Pechenizkiy, M., Mobasher, B., and Burke, R. (2020). Fairmatch: A graph-based approach for improving aggregate diversity in recommender systems. In *Proceedings of the 28th ACM Conference on User Modeling, Adaptation and Personalization*, pages 154–162.
- Masoud, M., Abdollahpouri, H., Pechenizkiy, M., Mobasher, B., and Burke, R. (2020). Feedback loop and bias amplification in recommender systems. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 2145–2148.

- McNee, S. M., Riedl, J., and Konstan, J. A. (2006). Being accurate is not enough: how accuracy metrics have hurt recommender systems. In *CHI'06 extended abstracts on Human factors in computing systems*, pages 1097–1101.
- Menzies, T. and Sinsel, E. (2000). Practical large scale what-if queries: Case studies with software risk assessment. In *Proceedings ASE 2000. 15th IEEE International Conference on Automated Software Engineering*, pages 165–173. IEEE.
- Merlo-Schett, N., Glinz, M., and Mukhija, A. (2003). Seminar on software cost estimation. *Department of Computer Science, University of Zurich, Switzerland*.
- Montaner, M., López, B., and de la Rosa, J. L. (2002). Improving case representation and case base maintenance in recommender agents. In *European Conference on Case-Based Reasoning*, pages 234–248. Springer.
- Morwitz, V. G. and Schmittlein, D. (1992). Using segmentation to improve sales forecasts based on purchase intent: Which “intenders” actually buy? *Journal Of Marketing Research*, 29(4):391–405.
- Nguyen, H. and Haddawy, P. (1998). Diva: applying decision theory to collaborative filtering. In *Proc. of the AAAI Workshop on Recommender Systems*.
- Nicolas, P. R. (2015). *Scala for machine learning*. Packt Publishing Ltd.
- Niesen, T., Houy, C., Fettke, P., and Loos, P. (2016). Towards an integrative big data analysis framework for data-driven risk management in industry 4.0. In *2016 49th Hawaii International Conference on System Sciences (HICSS)*, pages 5065–5074. IEEE.

- Nilashi, M., Ibrahim, O. B., Ithnin, N., and Zakaria, R. (2015). A multi-criteria recommendation system using dimensionality reduction and neuro-fuzzy techniques. *Soft Computing*, 19(11):3173–3207.
- Parsons, J., Ralph, P., and Gallagher, K. (2004). Using viewing time to infer user preference in recommender systems. In *AAAI Workshop in Semantic Web Personalization*.
- Pathak, B., Garfinkel, R., Gopal, R. D., Venkatesan, R., and Yin, F. (2010). Empirical analysis of the impact of recommender systems on sales. *Journal of Management Information Systems*, 27(2):159–188.
- Pazzani, M. J. and Billsus, D. (2007). Content-based recommendation systems. In *The adaptive web*, pages 325–341. Springer.
- Petersen, M. K. and Hansen, L. K. (2011). Emotional nodes among lines of lyrics. In *Face and Gesture 2011*, pages 821–826. IEEE.
- Philip, K. and Chan, S. (1998). Toward scalable learning with non-uniform class and cost distributions: A case study in credit card fraud detection. In *Proceeding of the 4th International Conference on Knowledge Discovery and Data Mining*, pages 164–168.
- Pizarro, J., Guerrero, E., and Galindo, P. L. (2002). Multiple comparison procedures applied to model selection. *Neurocomputing*, 48(1-4):155–173.
- Power, D. J. and Sharda, R. (2007). Model-driven decision support systems: Concepts and research directions. *Decision Support Systems*, 43(3):1044–1061.
- Pu, P., Chen, L., and Hu, R. (2011). A user-centric evaluation framework for recommender systems. In *Proceedings of the 5th ACM Conference on Recommender Systems*, pages 157–164.

- Rafter, R. (2010). *Evaluation and conversation in collaborative filtering*. PhD thesis, University College Dublin.
- Rahman, H. and Ramos, I. (2013). *Ethical Data Mining Applications for Socio-Economic Development*. IGI Global.
- Ricci, F., Rokach, L., and Shapira, B. (2011). Introduction to recommender systems handbook. In *Recommender systems handbook*, pages 1–35. Springer.
- Robillard, M. P. and Walker, R. J. (2014). An introduction to recommendation systems in software engineering. In *Recommendation Systems in Software Engineering*, pages 1–11. Springer.
- Rose, K. H. (2013). A guide to the project management body of knowledge (pmbok® guide)—fifth edition. *Project management journal*, 44(3):e1–e1.
- Said, A., Fields, B., Jain, B. J., and Albayrak, S. (2013). User-centric evaluation of a k-furthest neighbor collaborative filtering recommender algorithm. In *Proceedings of the 2013 Conference on Computer Supported Cooperative Work*, pages 1399–1408.
- Sakar, C. O., Polat, S. O., Katircioglu, M., and Kastro, Y. (2019). Real-time prediction of online shoppers’ purchasing intention using multilayer perceptron and lstm recurrent neural networks. *Neural Computing and Applications*, 31(10):6893–6908.
- Sanchez, F., Barrilero, M., Uribe, S., Alvarez, F., Tena, A., and Menendez, J. M. (2012). Social and content hybrid image recommender system for mobile social networks. *Mobile Networks and Applications*, 17(6):782–795.
- Sapsford, R. and Jupp, V. (1996). *Data collection and analysis*. Sage.

- Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international Conference on World Wide Web*, pages 285–295.
- Sato, M., Izumo, H., and Sonoda, T. (2016). Modeling individual users’ responsiveness to maximize recommendation impact. In *Proceedings of the 2016 Conference on User Modeling Adaptation and Personalization*, pages 259–267.
- Schapire, R. E. (1990). The strength of weak learnability. *Machine learning*, 5(2):197–227.
- Schubert, P. and Ginsburg, M. (2000). Virtual communities of transaction: The role of personalization in electronic commerce. *Electronic Markets*, 10(1):45–55.
- Segaran, T. (2007). *Programming collective intelligence: building smart web 2.0 applications*. O’Reilly Media, Inc.
- Shani, G., Heckerman, D., and Brafman, R. I. (2005). An mdp-based recommender system. *Journal of Machine Learning Research*, 6(Sep):1265–1295.
- Shardanand, U. (1994). *Social information filtering for music recommendation*. PhD thesis, Massachusetts Institute of Technology.
- Sharif, M. A. and Raghavan, V. V. (2014). A clustering based scalable hybrid approach for web page recommendation. In *2014 IEEE International Conference on Big Data (Big Data)*, pages 80–87. IEEE.
- Sheil, H., Rana, O., and Reilly, R. (2018). Predicting purchasing intent: Automatic feature learning using recurrent neural networks. In *Proceedings of ACM SIGIR Workshop on eCommerce*.

- Shlomo, B., Ivan, C., and Domonkos, T. (2018). *Collaborative Recommendations: Algorithms, Practical Challenges And Applications*. World Scientific.
- Sun, Y., Kamel, M. S., and Wang, Y. (2006). Boosting for learning multiple classes with imbalanced class distribution. In *6th International Conference on Data Mining (ICDM'06)*, pages 592–602. IEEE.
- Tam, K. Y. and Ho, S. Y. (2003). Web personalization: Is it effective? *IT professional*, 5(5):53–57.
- Terveen, L. and Hill, W. (2001). Beyond recommender systems: Helping people help each other. *HCI in the New Millennium*, 1(2001):487–509.
- Trivedi, A., Rai, P., DuVall, S. L., and Daumé III, H. (2010). Exploiting tag and word correlations for improved webpage clustering. In *Proceedings of the 2nd International Workshop on Search and Mining User-generated Contents*, pages 3–12.
- Van Meteren, R. and Van Someren, M. (2000). Using content-based filtering for recommendation. In *Proceedings of the Machine Learning in the New Information Age: MLnet/ECML2000 Workshop*, volume 30, pages 47–56.
- Vargas, S. and Castells, P. (2014). Improving sales diversity by recommending users to items. In *Proceedings of the 8th ACM Conference on Recommender Systems*, pages 145–152.
- Vázquez, E. G., Escolano, A. Y., Riaño, P. G., and Junquera, J. P. (2001). Repeated measures multiple comparison procedures applied to model selection in neural networks. In *International Work-Conference on Artificial Neural Networks*, pages 88–95. Springer.
- Wang, X., Zhang, J., and Wang, Y. (2016). Trust-aware privacy-preserving

- recommender system. In *Proceedings of the 9th EAI International Conference on Mobile Multimedia Communications*, pages 107–115.
- Watson, C. C. A. I. T. et al. (2016). *Recommender Systems: The Textbook*. Buku Digital.
- Weiss, G. M. (2004). Mining with rarity: a unifying framework. *ACM SIGKDD Explorations Newsletter*, 6(1):7–19.
- Wen, H., Fang, L., and Guan, L. (2012). A hybrid approach for personalized recommendation of news on the web. *Expert Systems with Applications*, 39(5):5806–5814.
- Wheelwright, S. C. and Clark, K. B. (1992). *Revolutionizing product development: quantum leaps in speed, efficiency, and quality*. Simon and Schuster.
- Whittington, O. R. (2013). *Wiley CPA Examination Review, Outlines and Study Guides*. John Wiley & Sons.
- Wickham, H. et al. (2011). The split-apply-combine strategy for data analysis. *Journal of Statistical Software*, 40(1):1–29.
- Widmer, G. and Kubat, M. (1996). Learning in the presence of concept drift and hidden contexts. *Machine learning*, 23(1):69–101.
- Willemsen, M. C., Knijnenburg, B. P., Graus, M. P., Velter-Bremmers, L. C., and Fu, K. (2011). Using latent features diversification to reduce choice difficulty in recommendation lists. *ACM Conference On Recommender Systems*, 11(2011):14–20.
- Yen, S.-J. and Lee, Y.-S. (2009). Cluster-based under-sampling approaches for imbalanced data distributions. *Expert Systems with Applications*, 36(3):5718–5727.

- Zadrozny, B., Langford, J., and Abe, N. (2003). Cost-sensitive learning by cost-proportionate example weighting. In *3rd IEEE International Conference on Data Mining*, pages 435–442. IEEE.
- Zeira, G., Maimon, O., Last, M., and Rokach, L. (2004). Change detection in classification models induced from time series data. In *Data mining in time series databases*, pages 101–125. World Scientific.
- Zhang, W., Wang, J., Chen, B., and Zhao, X. (2013). To personalize or not: a risk management perspective. In *Proceedings of the 7th ACM Conference on Recommender Systems*, pages 229–236.
- Zhang, X. X., Zhang, S., Wang, Y., Gogate, M., Ning, Y., Peng, C., Liu, I., and Lee, C. (2018). Optimizing listing efficiency and efficacy for a delivery coordination system. US Patent App. 15/586,190.
- Zhao, X., Niu, Z., Chen, W., Shi, C., Niu, K., and Liu, D. (2015). A hybrid approach of topic model and matrix factorization based on two-step recommendation framework. *Journal of Intelligent Information Systems*, 44(3):335–353.
- Zhao, Y. and Cen, Y. (2013). *Data mining applications with R*. Academic Press.
- Zhou, Z.-H. (2012). *Ensemble methods: foundations and algorithms*. Chapman and Hall/CRC.
- Zhu, J., Rosset, S., Zou, H., and Hastie, T. (2006). Multi-class adaboost. *Ann Arbor*, 1001:48109.
- Ziegler, C.-N., McNee, S. M., Konstan, J. A., and Lausen, G. (2005). Improving recommendation lists through topic diversification. In *Proceedings of the 14th international Conference on World Wide Web*, pages 22–32.

Co-Authorship Form

This form is to accompany the submission of any PhD that contains published or unpublished co-authored work. **Please include one copy of this form for each co-authored work.** Completed forms should be included in all copies of your thesis submitted for examination and library deposit (including digital deposit), following your thesis Acknowledgements. Co-authored works may be included in a thesis if the candidate has written all or the majority of the text and had their contribution confirmed by all co-authors as not less than 65%.

Please indicate the chapter/section/pages of this thesis that are extracted from a co-authored work and give the title and publication details or details of submission of the co-authored work.

Parts of Chapter 3 have been published in:

Algawiaz, D., Dobbie, G., Alam, S. (2019). PRARS: Risk Assessment Model for Recommender Systems. In IEEE 18th International Conference on Intelligent Software Methodologies, Tools, and Techniques, pages 701-710.

Nature of contribution by PhD candidate	Main author
Extent of contribution by PhD candidate (%)	80

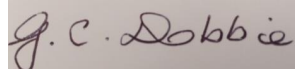

CO-AUTHORS

Name	Nature of Contribution
Gillian Dobbie	helped develop ideas, editing
Shafiq Alam	Co_author

Certification by Co-Authors

The undersigned hereby certify that:

- ❖ the above statement correctly reflects the nature and extent of the PhD candidate's contribution to this work, and the nature of the contribution of each of the co-authors; and
- ❖ that the candidate wrote all or the majority of the text.

Name	Signature	Date
Gillian Dobbie		22/10/20
Shafiq Alam		23/10/20

Co-Authorship Form

This form is to accompany the submission of any PhD that contains published or unpublished co-authored work. **Please include one copy of this form for each co-authored work.** Completed forms should be included in all copies of your thesis submitted for examination and library deposit (including digital deposit), following your thesis Acknowledgements. Co-authored works may be included in a thesis if the candidate has written all or the majority of the text and had their contribution confirmed by all co-authors as not less than 65%.

Please indicate the chapter/section/pages of this thesis that are extracted from a co-authored work and give the title and publication details or details of submission of the co-authored work.

Parts of Chapter 4 have been published in:

Algawiaz, D., Dobbie, G., Alam, S. (2019). Predicting a User's Purchase Intention Using AdaBoost. In IEEE 14th International Conference on Intelligent Systems and Knowledge Engineering, pages 324-328.

Nature of contribution by PhD candidate	Main author
Extent of contribution by PhD candidate (%)	80

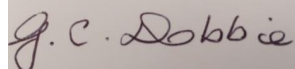

CO-AUTHORS

Name	Nature of Contribution
Gillian Dobbie	helped develop ideas, editing
Shafiq Alam	Co_author

Certification by Co-Authors

The undersigned hereby certify that:

- ❖ the above statement correctly reflects the nature and extent of the PhD candidate's contribution to this work, and the nature of the contribution of each of the co-authors; and
- ❖ that the candidate wrote all or the majority of the text.

Name	Signature	Date
Gillian Dobbie		22/10/20
Shafiq Alam		23/10/20

Co-Authorship Form

This form is to accompany the submission of any PhD that contains published or unpublished co-authored work. **Please include one copy of this form for each co-authored work.** Completed forms should be included in all copies of your thesis submitted for examination and library deposit (including digital deposit), following your thesis Acknowledgements. Co-authored works may be included in a thesis if the candidate has written all or the majority of the text and had their contribution confirmed by all co-authors as not less than 65%.

Please indicate the chapter/section/pages of this thesis that are extracted from a co-authored work and give the title and publication details or details of submission of the co-authored work.

Parts of Chapter 5 have been submitted to:

Algawiaz, D., Dobbie, G., Alam, S. (2020). Risk Assessment to Predict Optimal Number of Item Category. In review at PAKDD 2021.

Nature of contribution by PhD candidate	Main author
Extent of contribution by PhD candidate (%)	80

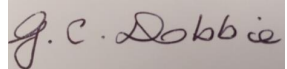

CO-AUTHORS

Name	Nature of Contribution
Gillian Dobbie	helped develop ideas, editing
Shafiq Alam	Co_author

Certification by Co-Authors

The undersigned hereby certify that:

- ❖ the above statement correctly reflects the nature and extent of the PhD candidate's contribution to this work, and the nature of the contribution of each of the co-authors; and
- ❖ that the candidate wrote all or the majority of the text.

Name	Signature	Date
Gillian Dobbie		22/10/20
Shafiq Alam		23/10/20

Last updated: 28 November 2017

