Twilm

PROJECT

ONELINER ABOUT

Authors
Martin Christian Havig

Supervisor:
Some One

February 27, 2014

Acknowledgments

Abstract

Contents

1 Introduction **1**

1.1 Purpose . . . 2

1.2 Motivation . . . 2

1.3 Context . . . 2

2 Preliminary Study **4**

2.1 State Of The Art . . . 5

2.1.1 System Coldstart Handling . . . 5

2.1.2 Fashion Recommendation . . . 5

2.1.3 Session Based Recommendation . . . 5

2.1.4 Recommenders (Similar systems? somethingsomething) . . . 5

2.1.5 Items clustering . . . 5

2.2 Data Findings . . . 5

2.2.1 What Can Be Understood From The Data . . . 5

2.2.2 Graphs N' Shit . . . 6

2.3 What to use . . . 6

2.3.1 Some Awesome Algorithms (Build up with project progress) . . . 6

2.3.2 Why Not To Use These (Same As above) . . . 6

2.4 How to evaluate . . . 6

2.4.1 What Has Been Done Before . . . 6

2.4.2 What To Use . . . 6

2.5 Evaluation . . . 6

3 Requirements **7**

3.1 Capturing the Requirements . . . 8

3.2 Functional Requirements . . . 8

3.3 Non Functional Requirements . . . 8

3.4 Prioritized Requirements 8

4 Design **9**

4.1 Architecture 10

4.1.1 Logical View 10

4.1.2 Process View 10

4.1.3 Physical View 10

4.2 Algorithm Design 10

4.2.1 Prediction 10

5 Implementation **11**

5.1 Major Requirements 12

5.1.1 FR1 12

5.1.2 FR6 12

5.1.3 FR7 12

5.1.4 NFR1 12

6 Evaluation **13**

6.1 Development Process 14

6.2 Result Evaluation 14

6.3 Issues 14

7 Conclusion **15**

7.1 Final Product 16

7.2 Related Work 16

7.3 Future Work 16

A Requirements **I**

A.1 Functional Requirements I

A.2 Non Functional Requirements I

B Design **II**

C Implementation **III**

C.1 Implemented Functional Requirements III

C.2 Implemented Non Functional Requirements III

References **IV**

List of Figures

List of Tables

1.1 Structure and chapters of the report. 3

Chapter 1

Introduction

Contents

1.1 Purpose **2**
1.2 Motivation **2**
1.3 Context **2**

1.1 Purpose

1.2 Motivation

1.3 Context

Chapter	Description
Chapter 1	The Introduction chapter gives an overview of the project to the reader. It also outlines the purpose and motivation of the project.
Chapter 2	The Preliminary Study chapter documents knowledge, research and technology that is relevant to the project, and how and why some of them were prioritized over others when it comes to how they are used in the project.
Chapter 3	The Requirements chapter describes the requirements of the project. It also describes how and why they were created.
Chapter 4	The Design chapter describes the design of the system and how it was made.
Chapter 5	The Implementation chapter describes the implementation of the system.
Chapter 6	Evaluation chapter discussed the development process, testing of results and major issues.
Chapter 7	The Conclusion chapter sums up the project and describes the findings and reflects on them. It also describes further work to be done.
Appendix	The appendix contains extended information such as a full list of the requirements.

Table 1.1: Structure and chapters of the report.

Chapter 2

Preliminary Study

Contents

2.1 State Of The Art 5
2.1.1 System Coldstart Handling 5
2.1.2 Fashion Recommendation 5
2.1.3 Session Based Recommendation 5
2.1.4 Recommenders (Similar systems? somethingsomething) 5
2.1.5 Items clustering 5
2.2 Data Findings 5
2.2.1 What Can Be Understood From The Data 5
2.2.2 Graphs N' Shit 6
2.3 What to use 6
2.3.1 Some Awesome Algorithms (Build up with project progress) 6
2.3.2 Why Not To Use These (Same As above) 6
2.4 How to evaluate 6
2.4.1 What Has Been Done Before 6
2.4.2 What To Use 6
2.5 Evaluation 6

2.1 State Of The Art

2.1.1 System Coldstart Handling

2.1.2 Fashion Recommendation

2.1.3 Session Based Recommendation

Init Hypothesis: Two users with similar session habits and similar product accessing pattern have a stronger correlation to one-another than two users with just similar product interests.

'product_purchase_intended' (user pushed to the product web store) shows a wider specter of information about the product, including additional colors, images and colors. For some it might be natural to explore the item there before "wanting" it. Making both

"product_purchase_intended" $\Rightarrow$ "*product_wanted*"

and

"product_purchase_intended" $\not\Rightarrow$ "*product_wanted*"

produce valuable information.

Must make different rules for the different stores: "Bik Bok", "Cubus", "Gina Trik", "H&M", "Bianco" has a broad specter of extra functions inside the web store, whereas others might not, only shows the product and a add to chart button. This might divide the use pattern of the users into a:

"product_detail_clicked" $\Rightarrow$ "*product_purchase_intended*" $\Rightarrow$ "*product_wanted*"

"product_detail_clicked" $\Rightarrow$ "*product_purchase_intended*" $\not\Rightarrow$ "*product_wanted*",

and

"product_detail_clicked" $\Rightarrow$ "*product_wanted*"

based on the store accessed.

Use this to make a "rule set" whit a probability. Then again use this to recommend items for the users with that given probability.

F:

Articles 4 l8er:

2.1.4 Recommenders (Similar systems? somethingsomething)

2.1.5 Items clustering

2.2 Data Findings

2.2.1 What Can Be Understood From The Data

The Expected

Event "app_started", all have user_id's Event "app_first_started", all user_id's are NULL Event "user_logged_in", all have user_id's... (assigned with login, event saved after login?)

The Strange

NULL valued events: (Not all strange, but put together for readability) facebook_share_changed collection_viewed wantlist_menu_entry_clicked app_became_active

app_first_started facebook_login_failed

> db.prod.distinct('event_json.ipAddress').length 9033 > db.prod.distinct('event_json.eventData.device_id').length 2644 > db.prod.distinct('user_id').length 1660

More devices than users, can't fill the blanks with device_id

Q's: app_became_active id's for better sessions? store_clicked vs. storefront_clicked (23 vs. 19744) API item-id's mapping to event product_id's; how to map?

2.2.2 Graphs N' Shit

2.3 What to use

2.3.1 Some Awesome Algorithms (Build up with project progress)

The Good

The Bad

2.3.2 Why Not To Use These (Same As above)

The Good

The Bad

2.4 How to evaluate

2.4.1 What Has Been Done Before

2.4.2 What To Use

The Good

The Bad

2.5 Evaluation

Thoughts:

Chapter 3

Requirements

Contents

3.1 Capturing the Requirements **8**
3.2 Functional Requirements **8**
3.3 Non Functional Requirements **8**
3.4 Prioritized Requirements **8**

3.1 Capturing the Requirements

3.2 Functional Requirements

FR1

FR6

FR7

FR1

FR6

FR7

3.3 Non Functional Requirements

NFR1

NFR1

3.4 Prioritized Requirements

Chapter 4

Design

Contents

4.1 Architecture 10
4.1.1 Logical View 10
4.1.2 Process View 10
4.1.3 Physical View 10
4.2 Algorithm Design 10
4.2.1 Prediction 10

4.1 Architecture

4.1.1 Logical View

4.1.2 Process View

4.1.3 Physical View

4.2 Algorithm Design

4.2.1 Prediction

Chapter 5

Implementation

Contents

5.1 Major Requirements 12
5.1.1 FR1 12
5.1.2 FR6 12
5.1.3 FR7 12
5.1.4 NFR1 12

5.1 Major Requirements

5.1.1 FR1

5.1.2 FR6

5.1.3 FR7

5.1.4 NFR1

Chapter 6

Evaluation

Contents

6.1 Development Process **14**
6.2 Result Evaluation **14**
6.3 Issues **14**

6.1 Development Process

Good

Bad

6.2 Result Evaluation

Testing of preliminary study

Testing of code functionality

Types of testing not used

6.3 Issues

Chapter 7

Conclusion

Contents

7.1 Final Product **16**
7.2 Related Work **16**
7.3 Future Work **16**

7.1 Final Product

7.2 Related Work

7.3 Future Work

Appendix A

Requirements

A.1 Functional Requirements

A.2 Non Functional Requirements

Appendix B

Design

Appendix C

Implementation

C.1 Implemented Functional Requirements

FR 1: Blablaba

FR 2: ~~Blablaba~~

C.2 Implemented Non Functional Requirements

NFR 1: Blablaba

NFR 2: ~~Blablaba~~

Bibliography