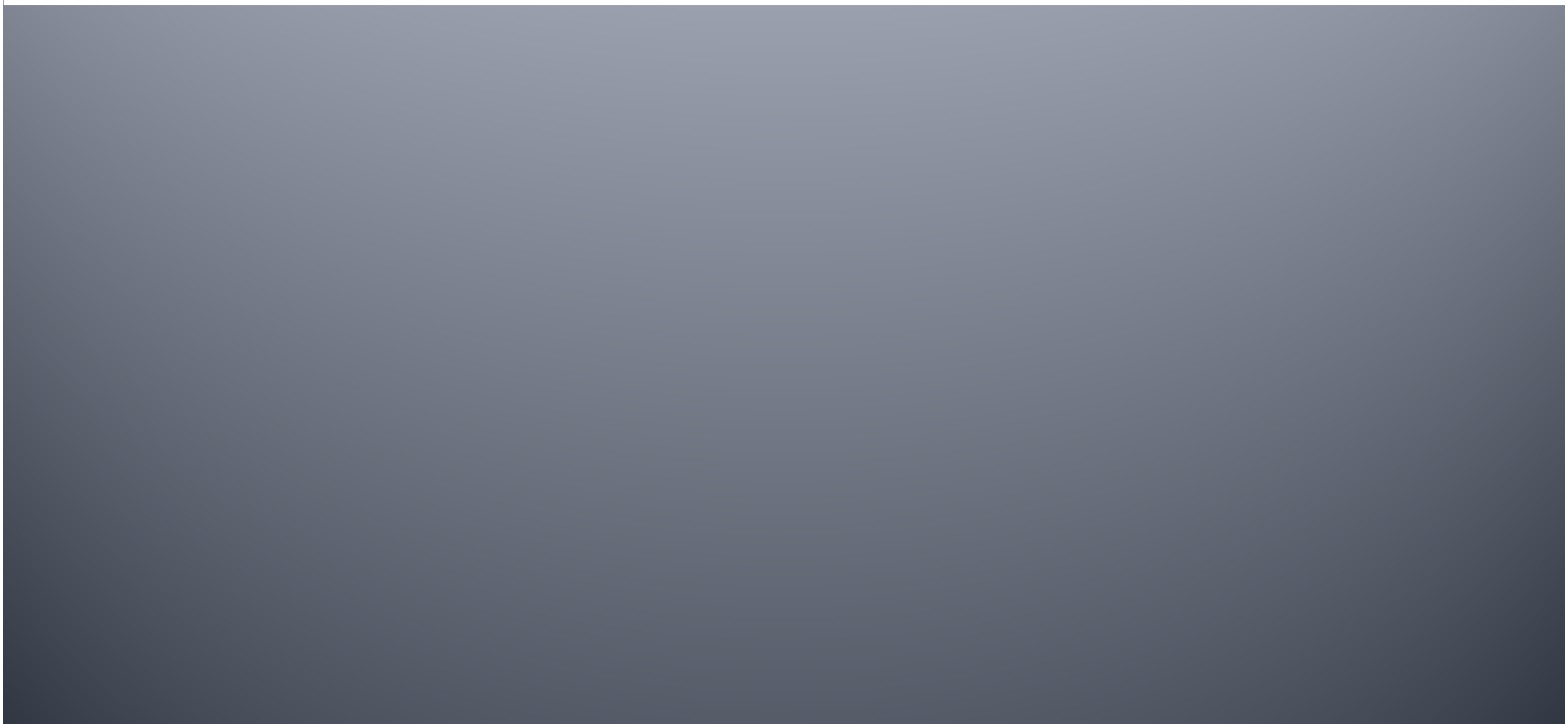
Master Thesis talk by Lyubomir Ganev
Supervisor: Prof. Dr. Antonio Krüger
Advisor: Matthias Böhmer

Evaluation of Context-aware Recommender Engines that Suggest Mobile Applications

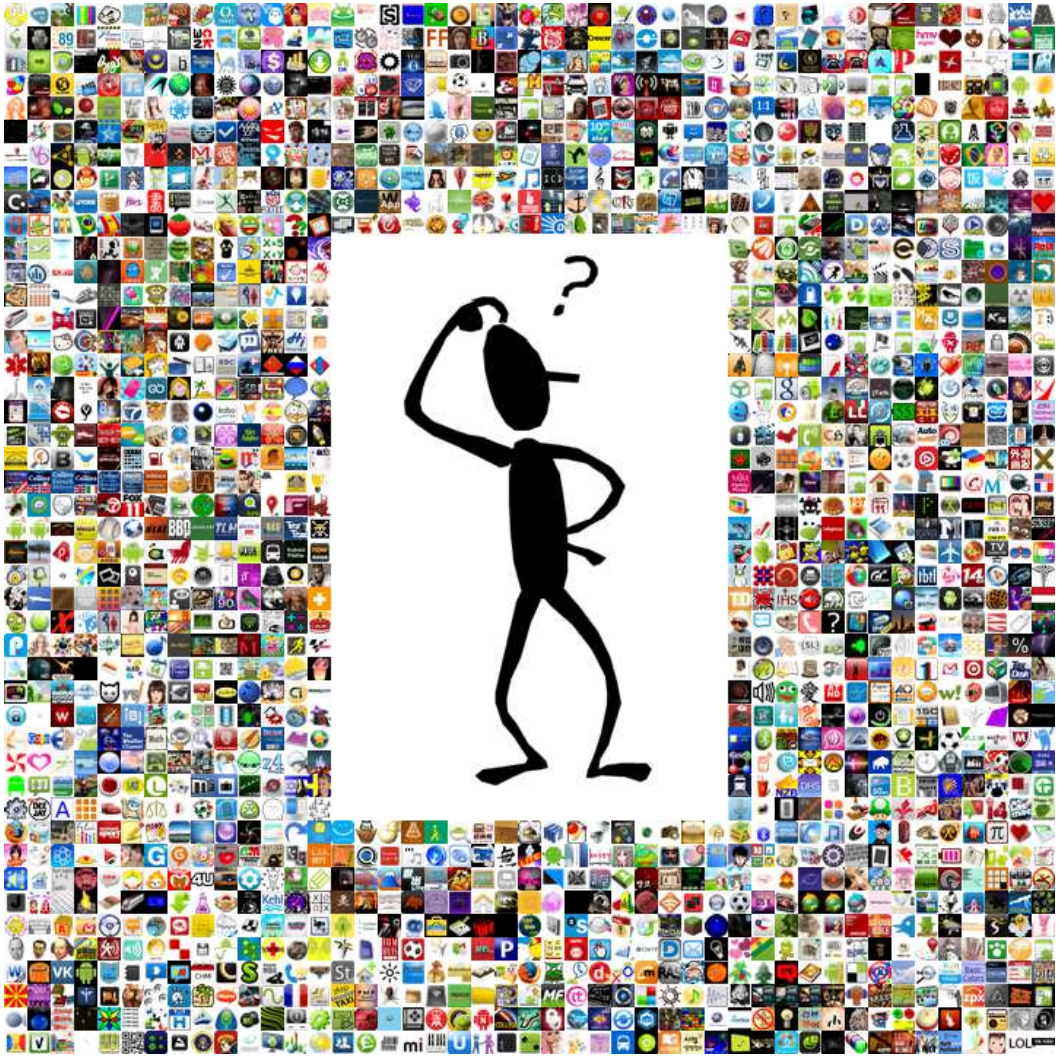
Outline

- Motivation
- Introduction
- Related Work
- Research Status
- Implementation
- Evaluation

Motivation



Motivation



- Mobile apps is a huge market
  - > 280,000 apps in Android Market (Sep.2011)
  - > 500,000 apps in Apple App Store (Oct.2011)
- Useful and interesting apps = needle in a haystack

[1] Wikipedia. List of digital distribution platforms for mobile devices. http://tiny.cc/joirz, accessed on October 21, 2011.

Naive Approach

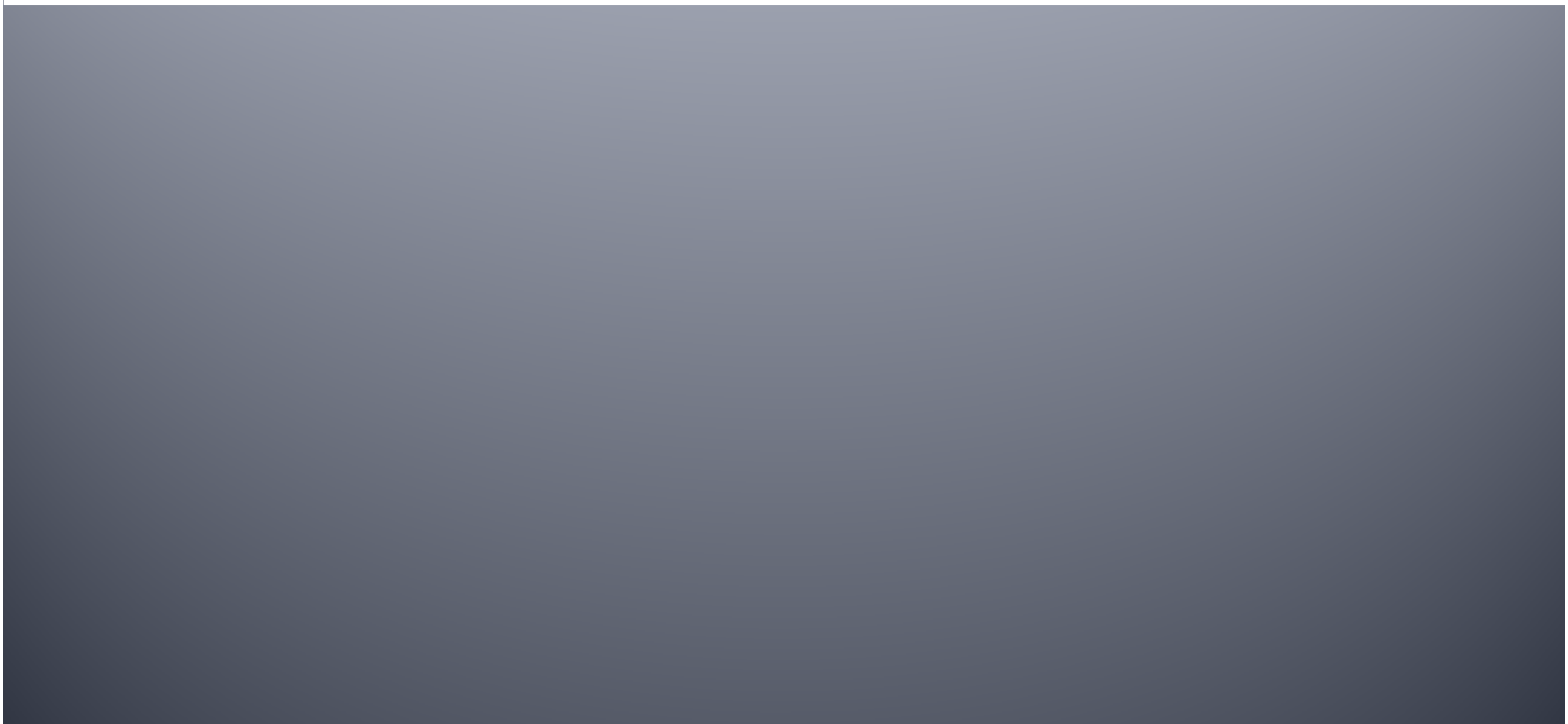
- Recommender systems that suggest mobile apps
  - Problem solved ? Not really!
  - *"Current systems that recommend mobile applications underachieve since they neglect the ever-changing context of their users and solely exploit sales figures and user ratings."*

[2] Böhmer, Bauer and Krüger. Exploring the Design Space of Context-aware Recommender Systems that Suggest Mobile Applications. In Proc. of Workshop CARS '10 (2010).

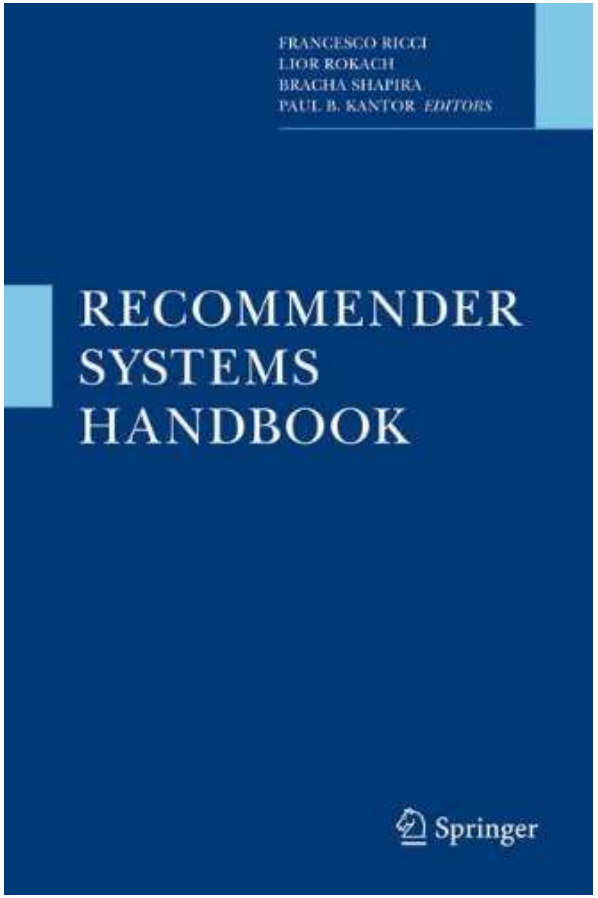
Problem Definition

- Mobile apps recommenders are still evolving
  - Different approaches for including context
- Specific evaluation approach is required
  - Following the lifecycle of mobile apps is important
- Goal: evaluate several recommendation engines inside a fully implemented mobile apps recommender system

Introduction



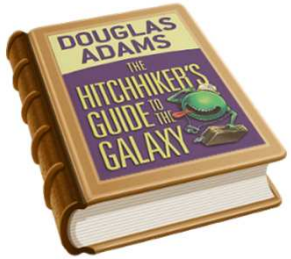
Recommender System Basics



- What is a Recommender System ?
  - *"Recommender Systems (RSs) are software tools and techniques providing suggestions for items to be of use to a user."*
- How does it work?
  - To identify the useful items for the user, a RS must be able to predict the utility of some items, or at least compare their utility, and then decide what items to recommend based on this comparison.

[3] Ricci, Rokach, Shapira and Kantor. Recommender systems handbook. Springer 2011.

Recommendable Items



- What kind of items?
  - Movies
  - Books
  - Electronics
  - Mobile Apps
- Items properties
  - Complexity
  - Utility

Recommendation approaches

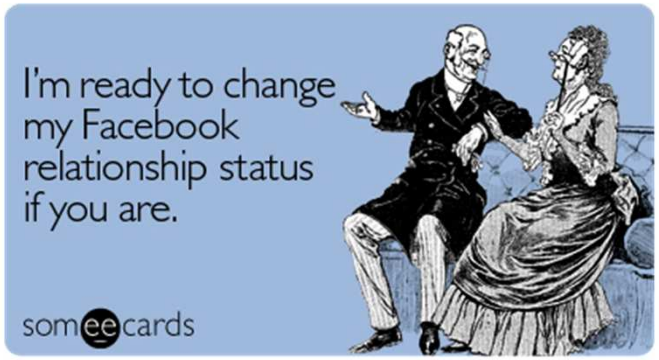
- Content-based – similar items based on item features
- Collaborative filtering – items which users with similar taste like
- Demographic – different demographic niches need different recommendations
- Knowledge-based – specific domain knowledge about how certain item features meet users needs
- Community-based – recommends items based on the preferences of the users friends
- Hybrid – combination of more than one approach

[3] Ricci, Rokach, Shapira and Kantor. Recommender systems handbook. Springer 2011.

Applications

- Entertainment – movies, music
- Content – news, documents, web pages
- E-commerce – consumer products to buy
- Services – travel, consulting, matchmaking

Context and Context-awareness



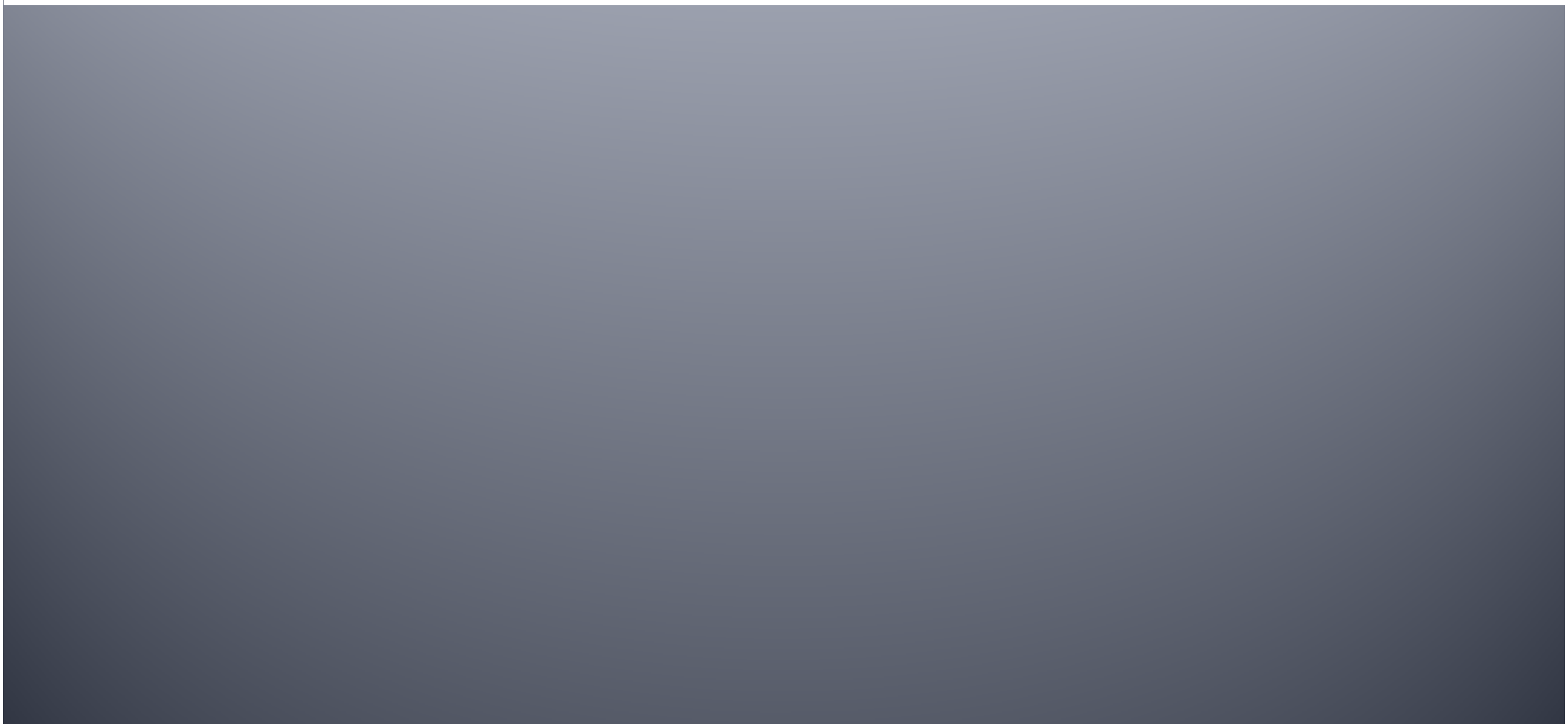
- Definitions:
  - *"Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves."*
  - *"A system is context-aware if it uses context to provide relevant information and/or services to the user, where relevancy depends on the user's task."*

[4] Dey. Understanding and Using Context. Personal Ubiquitous Computing, Vol. 5, No. 1. (20 January 2001), pp. 4-7.

Related Work

A Case Study on the Effectiveness of Recommendations in the Mobile Internet

Jannach and Hegelich (2009) [5]



Mobile Internet Recommendations

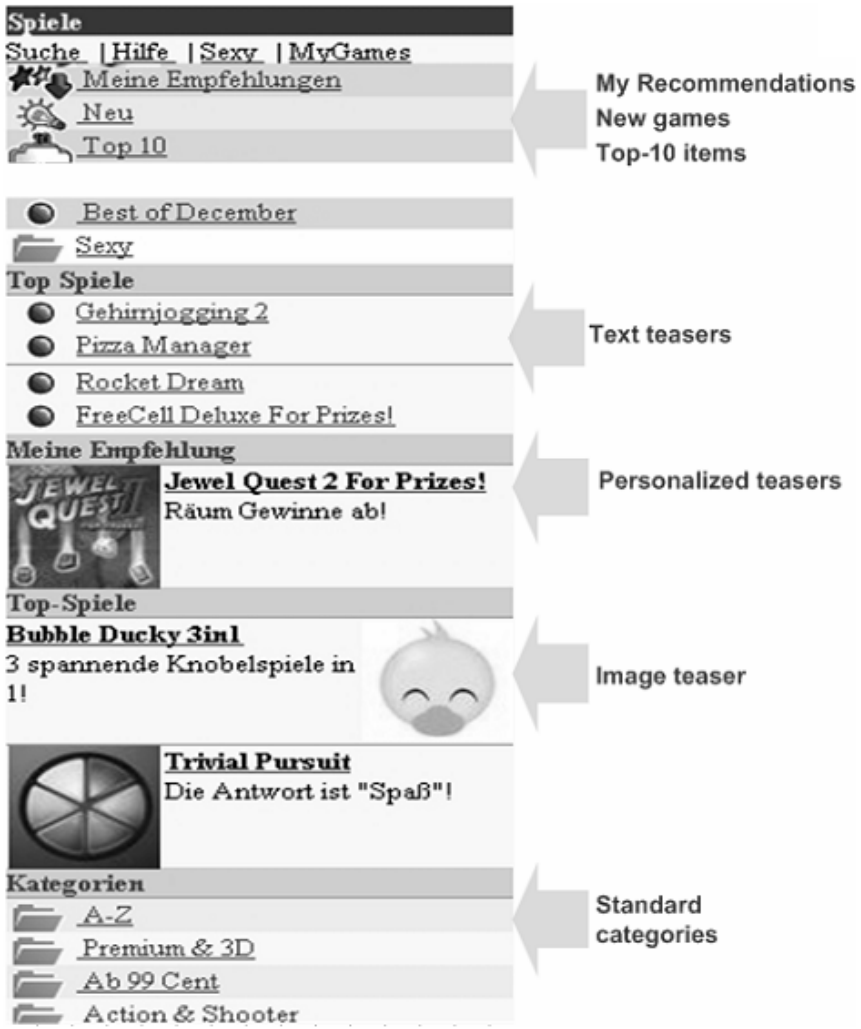


Fig 1. Layout of the website

- Study context:
  - A large-scale, commercial Mobile Internet platform for games
  - Sample of more than 155,000 different customers
  - 4 weeks evaluation period
- Analysis results:
  - Personalized recommendations increase the number of viewed and sold items

Mobile Internet Recommendations

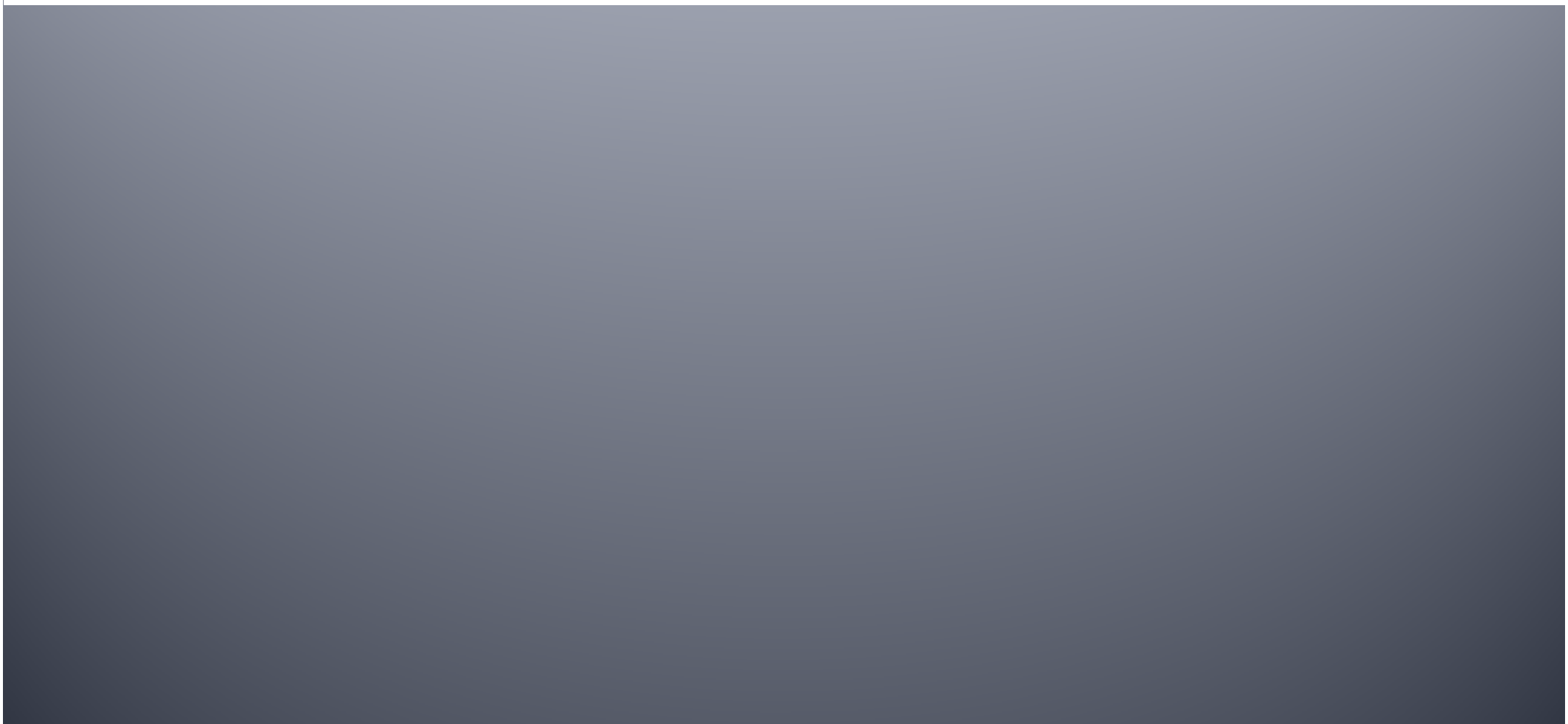
Method	Description
CF-Item	Item-based collaborative filtering
Content-Based	Item description and cosine similarity based
Hybrid	Combination of first two
SlopeOne	Item-based filtering
TopSeller	Ordering based on sales rank
TopRating	Average customer rating based ordering
Control	Manually-edited lists

Table 1. Description of groups

- Experimental setup
  - Participants divided in 7 groups on initial entering of the portal
  - Implicit customer ratings:
    - rating scale from -2 to +2
    - view = 0
    - purchase = 1
  - Explicit customer ratings, which override the implicit ones

An on-line Evaluation Framework for Recommender Systems

Hayes, Massa, Avesani and Cunningham (2002) [6]



On-line Evaluation Framework

- A comparative measure of how one recommendation strategy performs against another
- Problems with existing approaches
  - People tend to show variance when given the same test on two different occasions
  - Domain specific datasets used in many evaluation studies cannot be used to test every RS

On-line Evaluation Framework

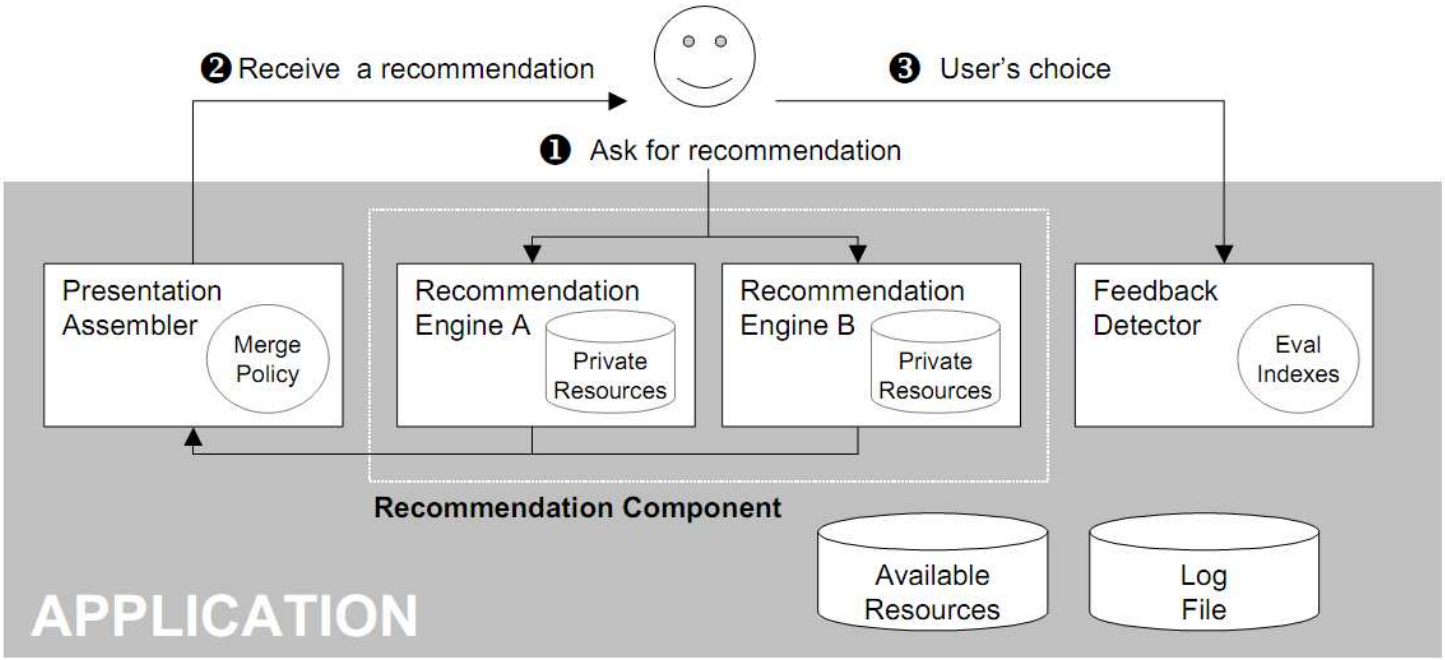


Fig 2.The framework architecture

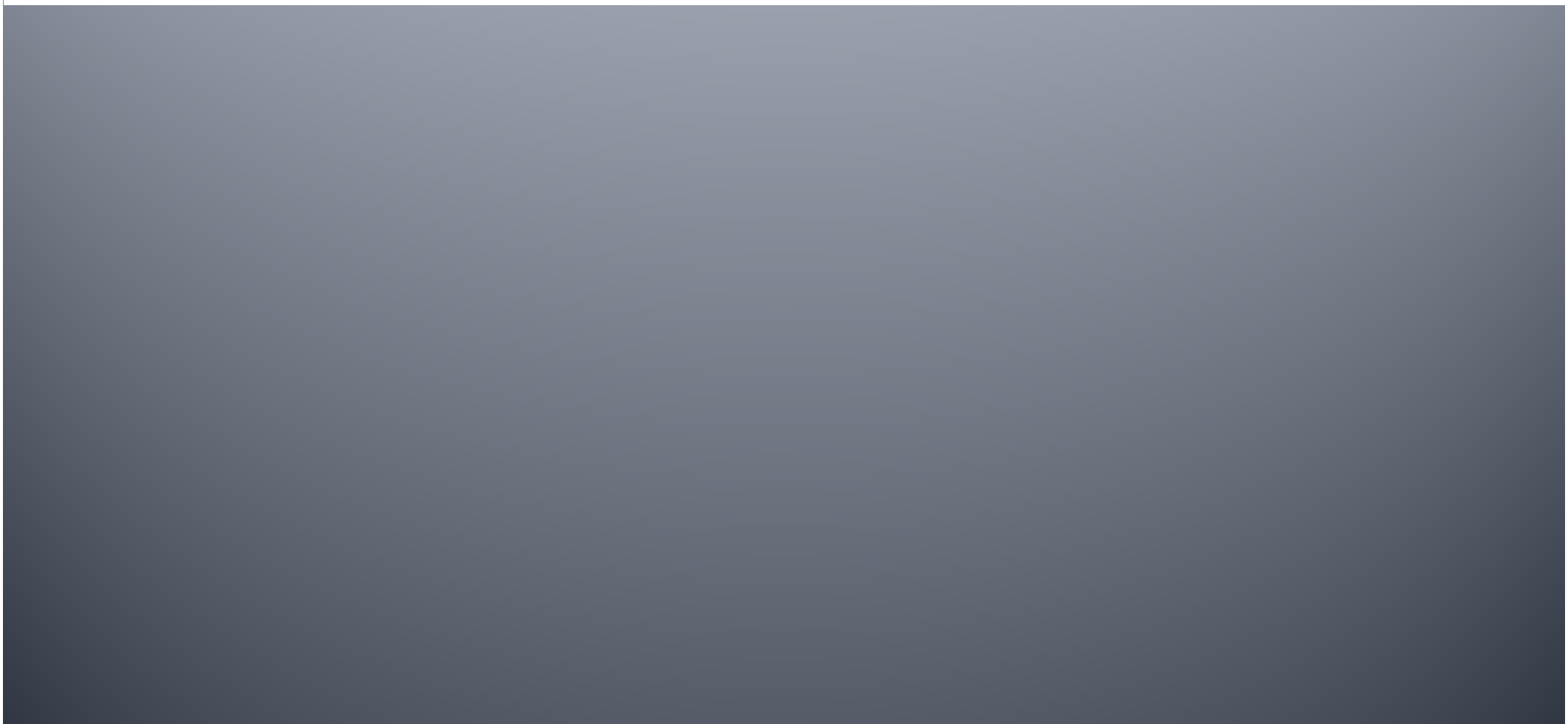
- Users are unaware of the source of recommendation
- Same conditions and same resources
- Presentation policy – merged; contrasting; cascading
- Evaluation feedback – implicit; explicit

On-line Evaluation Framework

- Advantages
  - Evaluation does not suffer from changes in user community or conditions
  - Possibility to extend the framework with an API for integration third party RS
- Drawbacks
  - Fully realized system with users is needed
  - Real users may be less tolerant of recommender error

A Hybrid Recommender System for Context-aware Recommendation of Mobile Applications

Woerndl, Schueller and Wojtech (2007) [7]



Hybrid Recommender System

- Goal
  - Deal with the added complexity of context by applying a hybrid recommender system
- Evaluation
  - Limited user evaluation
    - only 7 users and 6 applications
  - No explicit ratings
    - Information stored when user starts an application
    - Users are able to dislike an application

Hybrid Recommender System

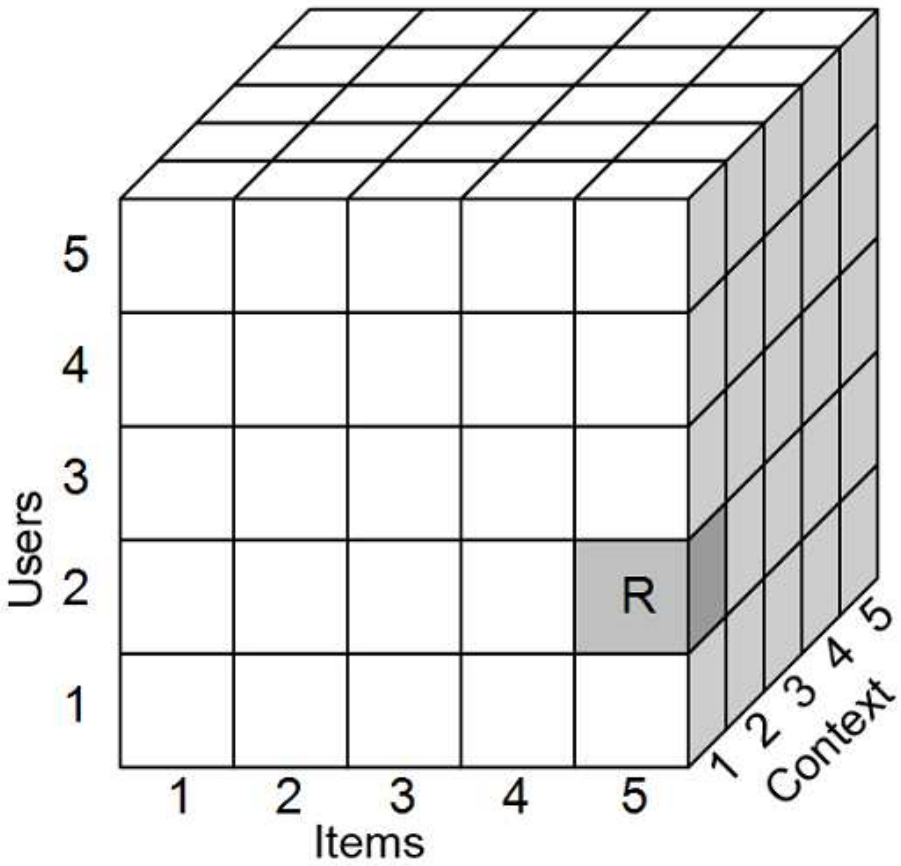
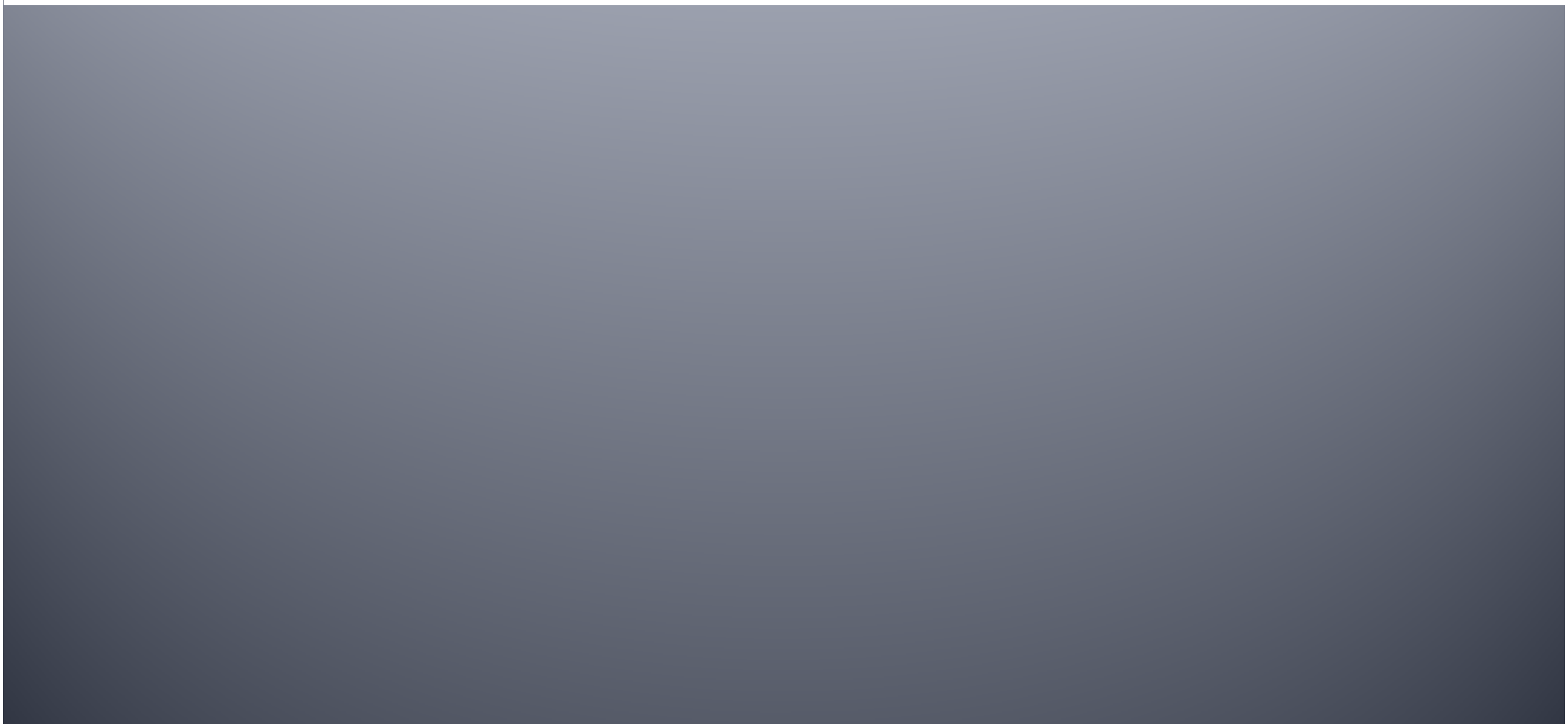


Fig 3.Integrating context in CF

- Cascading approach with 2 step recommendation
  - Content- or knowledge based filtering of relevant items based on context
  - Collaborative filtering to rank and additionally filter the result set
- Users could choose which recommender module to use

AppJoy: Personalized Mobile Application Discovery

Yan and Chen (2011) [8]



AppJoy

- Goal
  - Provide better personalized mobile application recommendations
- Motivation
  - User downloading an app is a weak indicator of whether the user likes that app
- Approach
  - Measuring real application usage and generate recommendations based on the usage scores

AppJoy

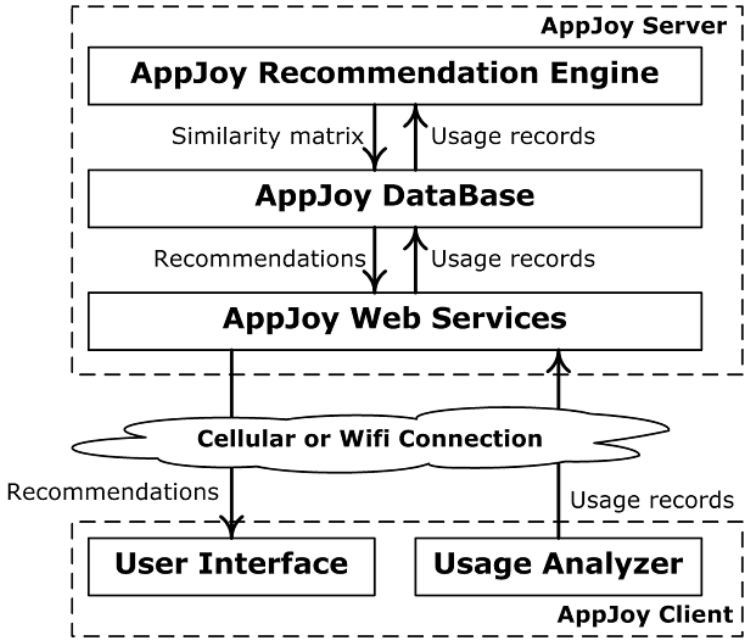


Fig 4.Architecture overview

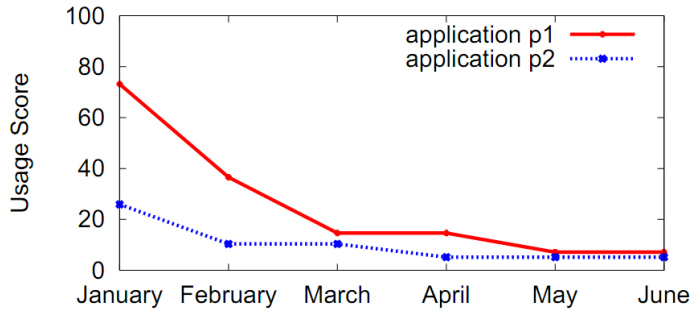


Fig 5. Usage score decreases over time once the apps are no longer used

- Client-Server architecture
- Assumption
  - The more an application is used suggests that the more the user likes it
- Usage score – RFD model
  - Recency – how recent
  - Frequency - how frequently for a time interval
  - Duration – how long

AppJoy

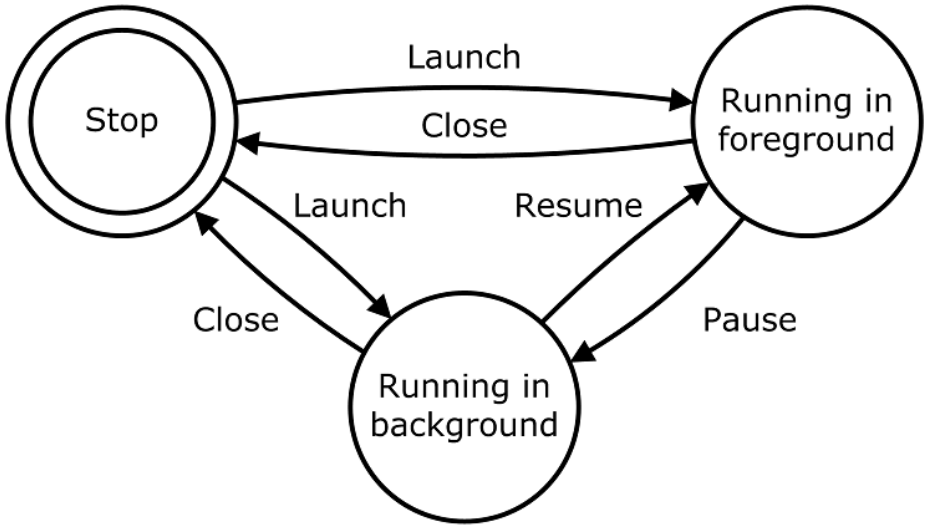


Fig 6.State transition of Android apps

- Automated data collection
- Application usage
  - Every second checking which application is in foreground
  - Application usage records containing the RFD model data
- User actions related to recommended apps
  - Recommended
  - Viewed
  - Installed
  - Rejected

AppJoy

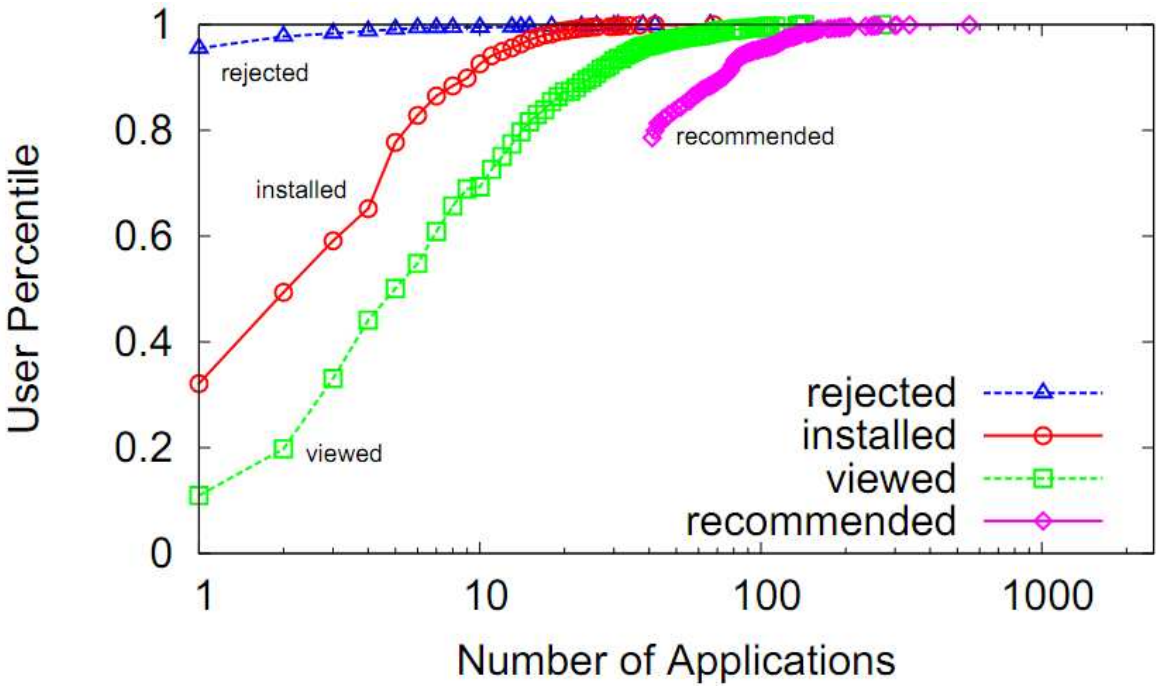


Fig 7. User actions statistic

- Evaluation characteristics
  - Fully implemented system
  - February 2010 – March 2011
- User actions
  - About 4.96% of all recommended apps were installed
  - Users tend to interact with the recommended apps longer

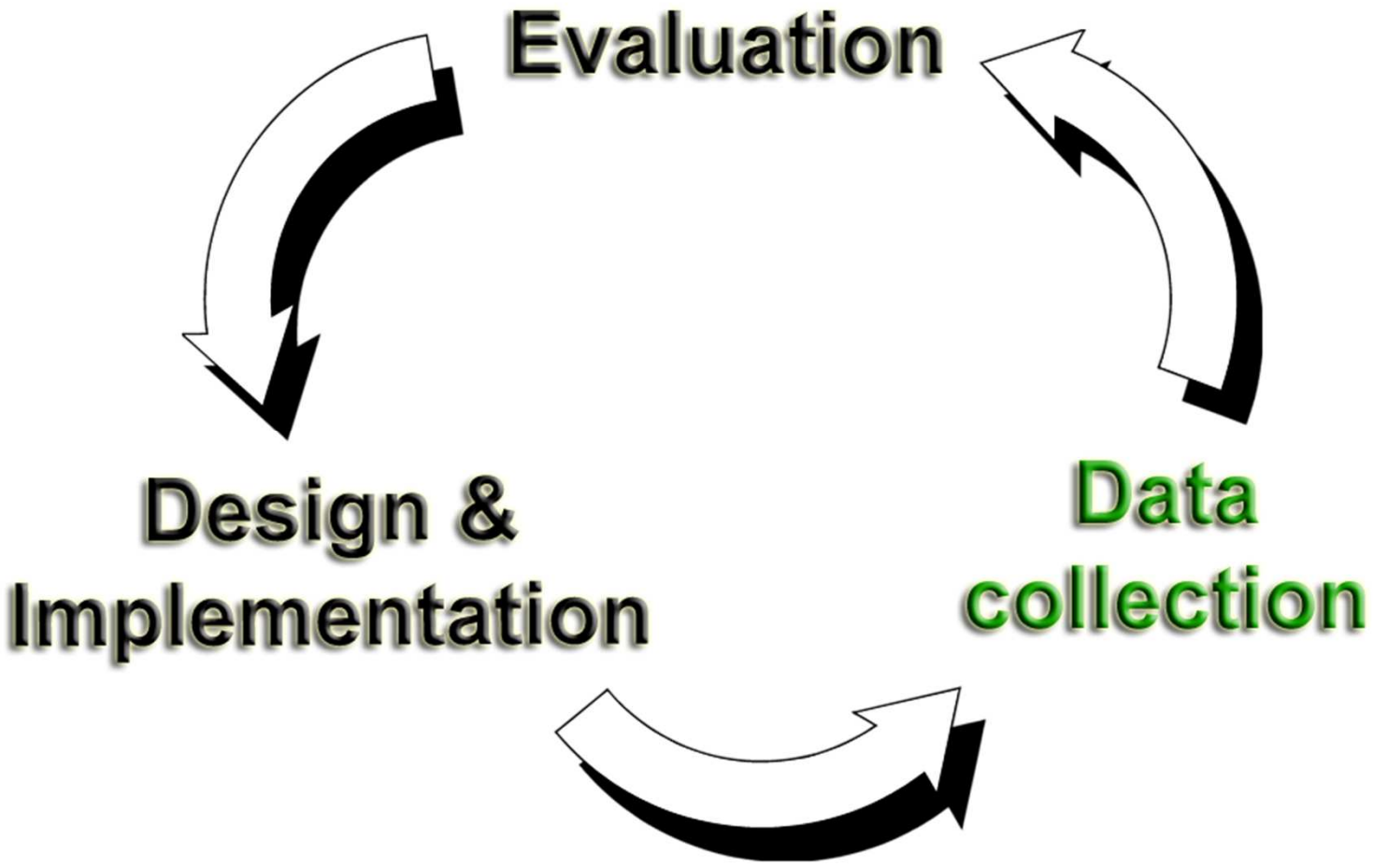
Contribution of my work

- Context-aware recommendation of mobile applications
- Fully implemented system with real users
- Monitoring of mobile application usage, user interactions and context entities
- Evaluation of several recommendation engines against each other

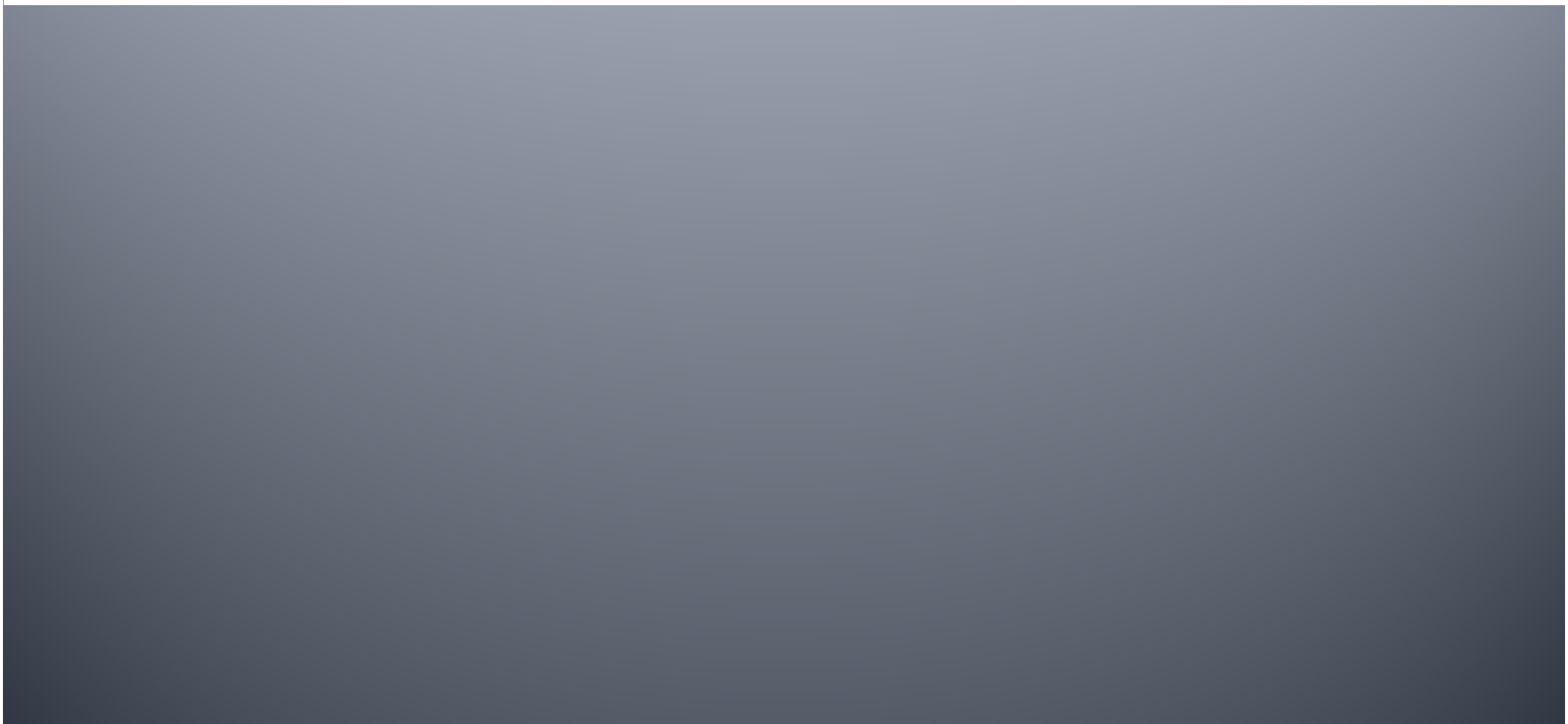
Research Status

Research Status

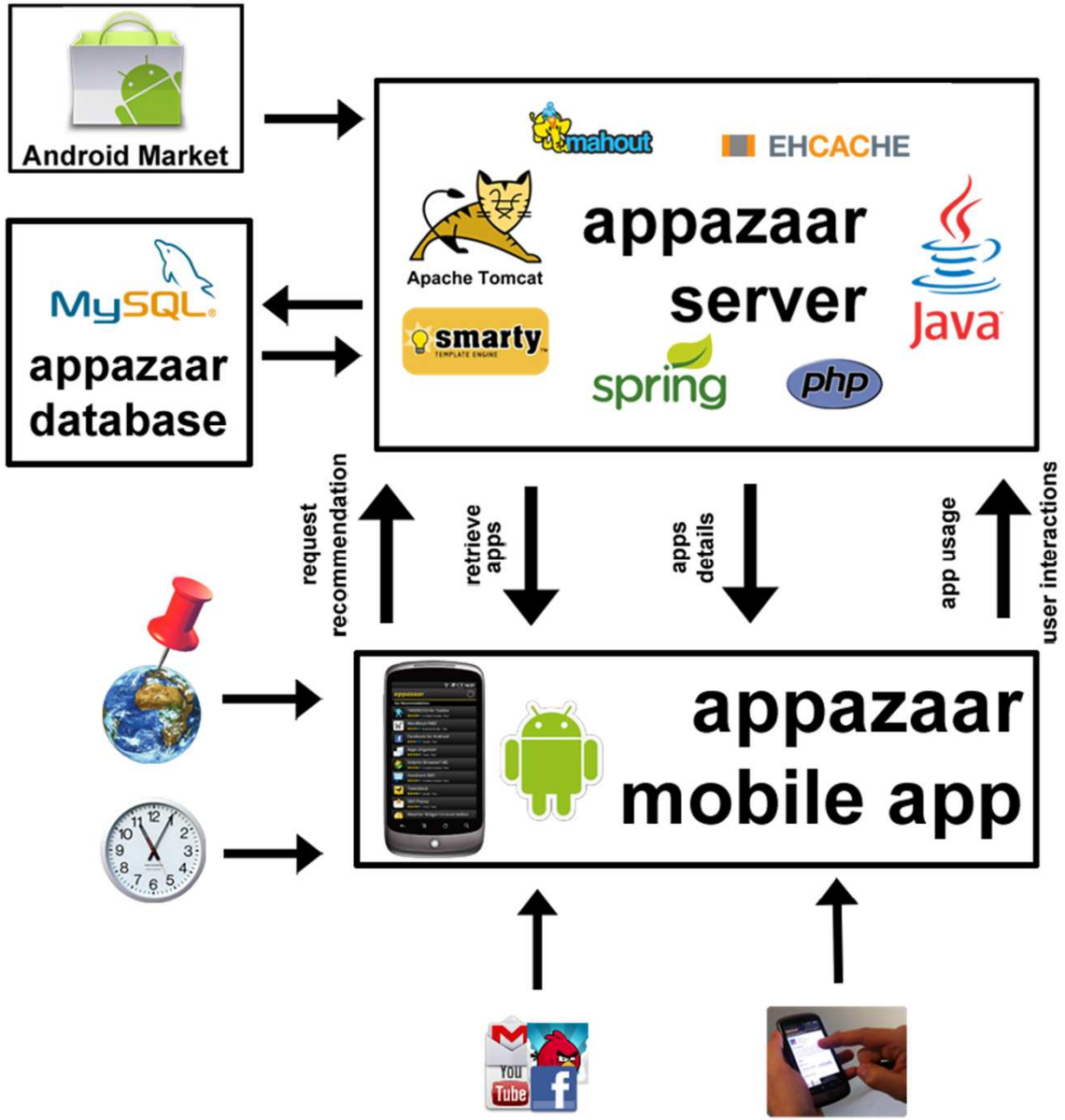
- The research is composed of 3 phases



Implementation



Implementation - Overview



- System architecture overview
  - appazaar server
    - Generate recommendations
    - Persist data from all devices
    - Maintain up to date apps details
  - appazaar mobile app
    - Present recommendations
    - Record various context entities
    - Record app usage data
    - Record user actions related to recommended apps

Implementation – appazaar Mobile App

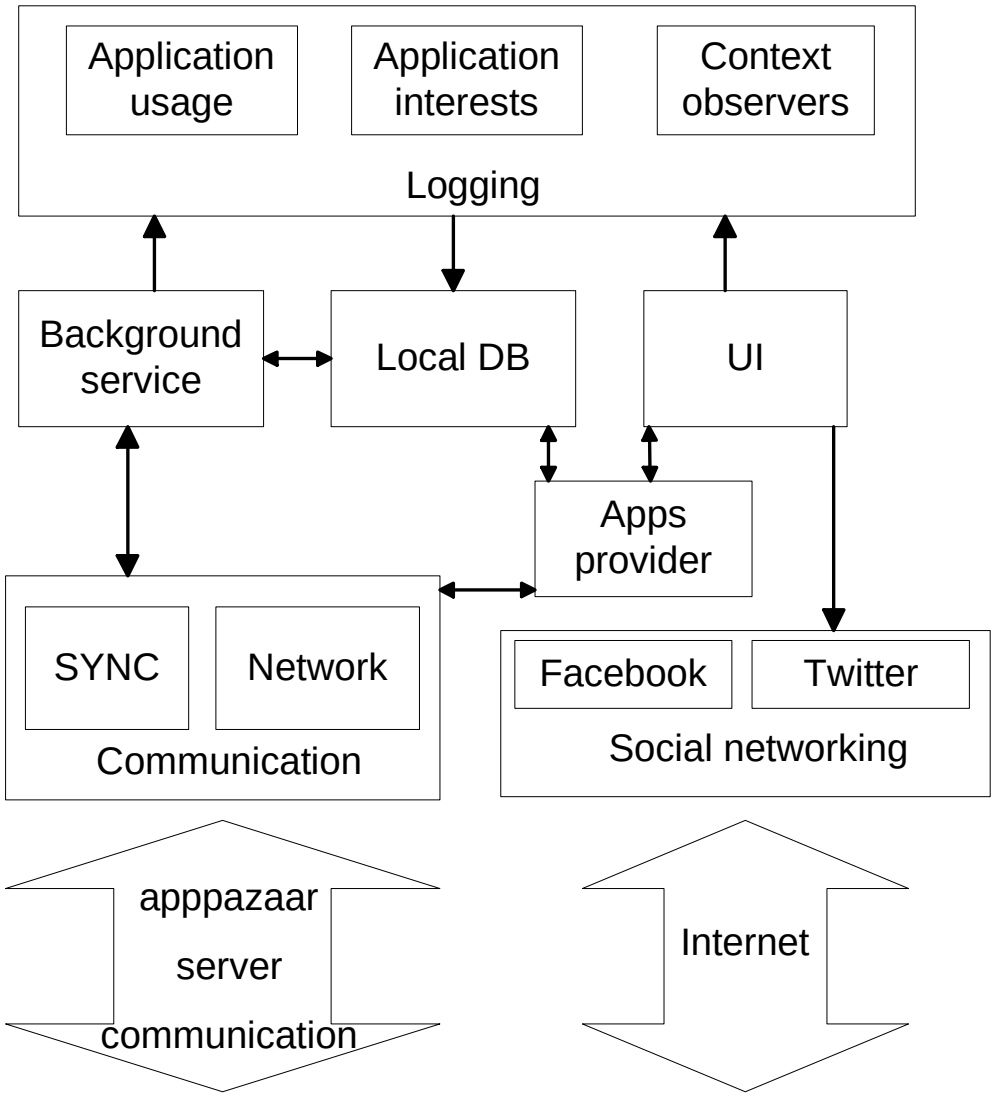


Fig 10. appazaar mobile app architecture

- Android app
  - Available to download in Android market
- Context entities
  - Device state – resolution, network state, model
  - Location – LON,LAT, UTC offset, country code
- App usage data
  - App lifecycle events
  - Data sampling rate 2Hz
- App interests
  - View, install, use recommended apps
  - Social networks sharing

Implementation – appazaar Server

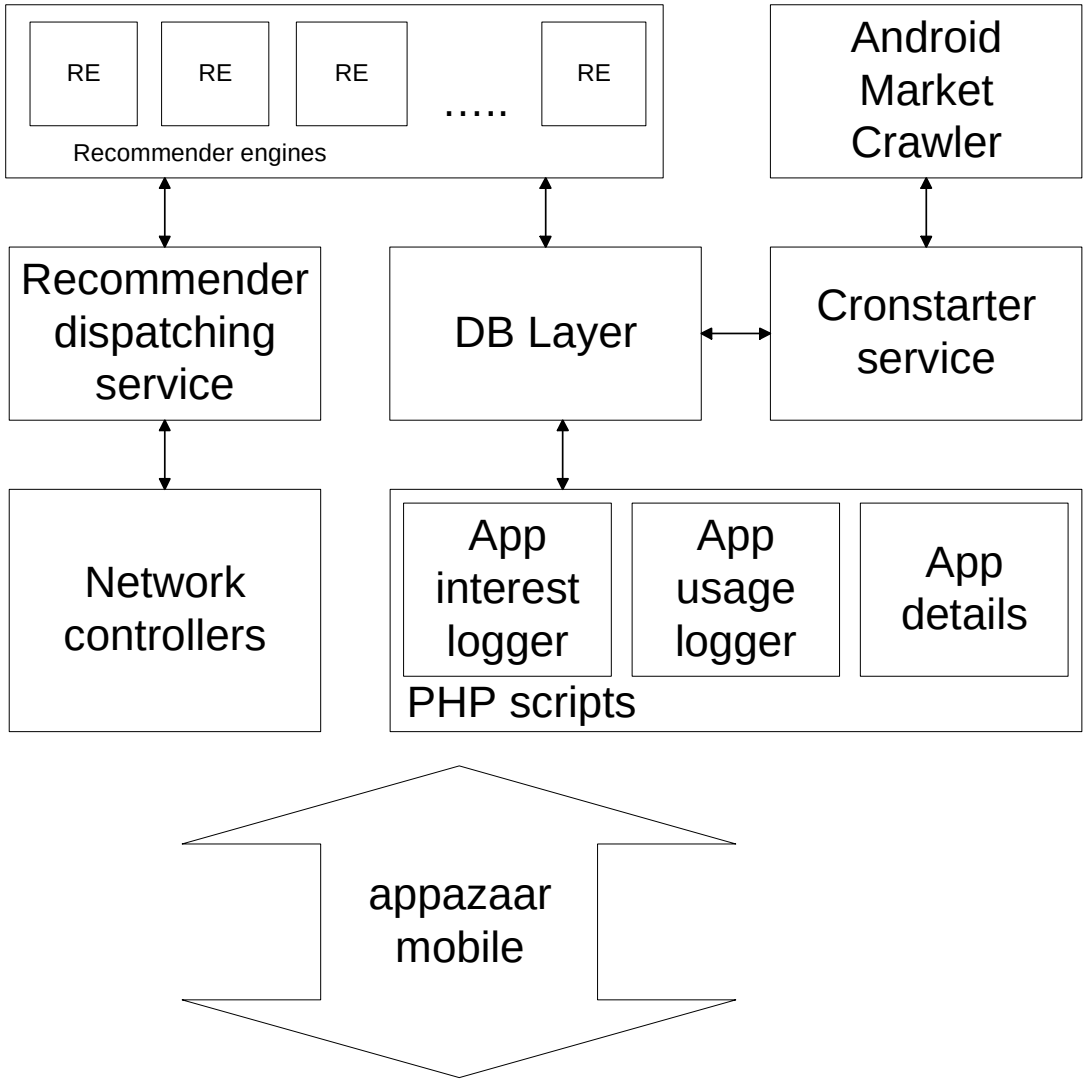


Fig 11. appazaar server architecture

- Android apps crawler
  - Crawl the market for app details, icons, etc.
- Random recommender engine dispatcher
- PHP scripts
  - logging of data coming from clients
  - application details in web pages
- Single database containing all data

Evaluation

Evaluation – Recommender Engines

	Non-personalized	Personalized
Non-contextualized	Most used apps Top rated apps Random apps	Usage based collaborative filtering
Contextualized	Running apps based	Location-aware usage collaborative filtering Time of day aware usage collaborative filtering

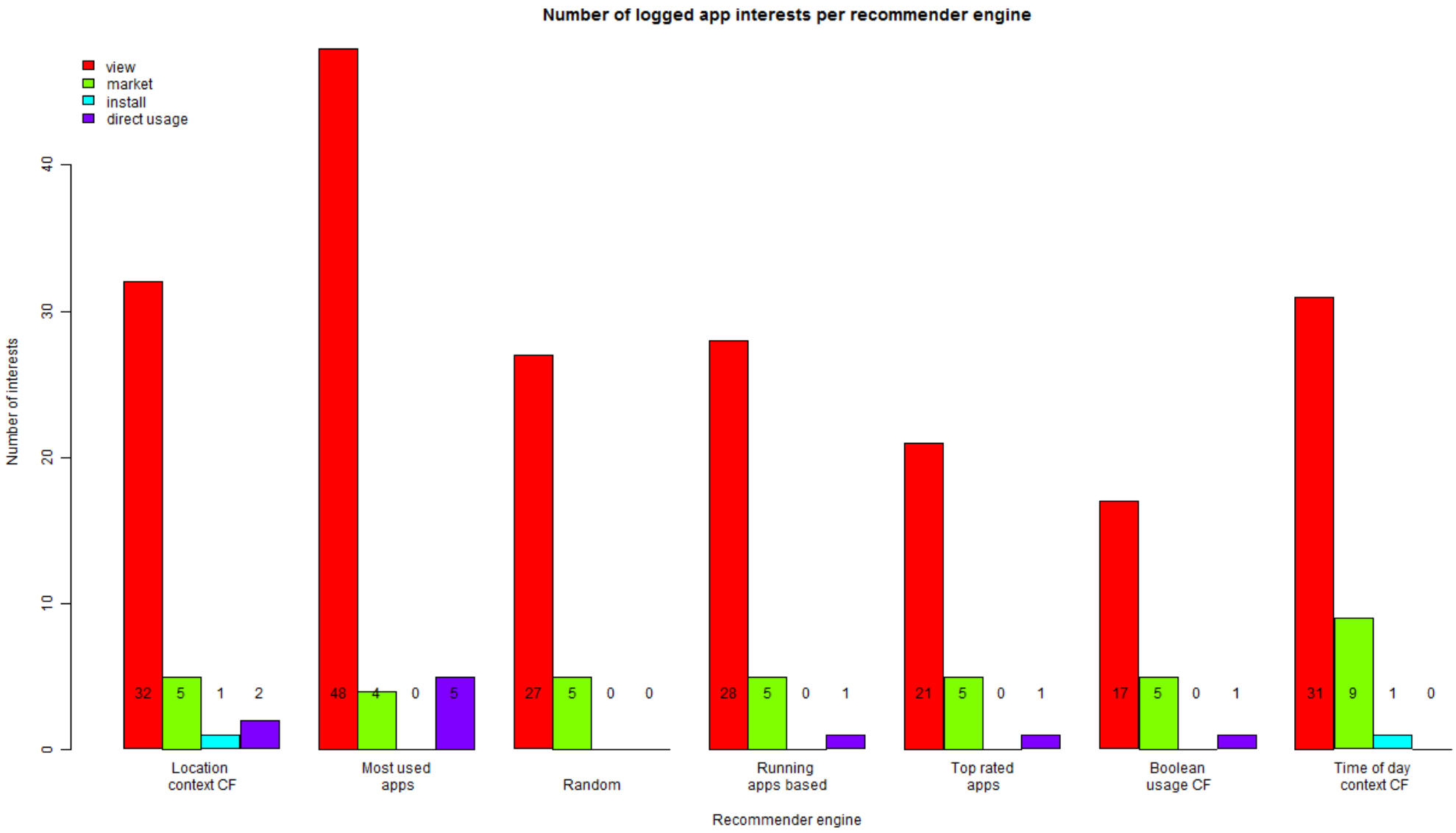
Table 2. Categorization of recommender engines

Evaluation – App Interest Model

- Metric – usefulness of recommendation
- Assumption – the more apps from a recommendation are installed and used, the more useful it is
- App interest model – chain of events caused by user interaction with recommended apps
- Every interest model is related to a particular recommendation

Fig 12. Chain of events in the application interest model

Evaluation – Example



Thank you!

Questions

References

1. Wikipedia. List of digital distribution platforms for mobile devices. http://tiny.cc/j0irz, accessed on October 21, 2011.
2. Böhmer, Bauer and Krüger. Exploring the Design Space of Context-aware Recommender Systems that Suggest Mobile Applications. In Proc. of Workshop CARS ’10 (2010).
3. Ricci, Rokach, Shapira and Kantor. Recommender systems handbook. Springer 2011.
4. Dey. Understanding and Using Context. Personal Ubiquitous Computing, Vol. 5, No. 1. (20 January 2001), pp. 4-7.
5. Jannach, Hegelich. A Case Study on the Effectiveness of Recommendations in the Mobile Internet. In Proc. of RECSYS ’09 (2009).
6. Hayes, Massa, Avesani and Cunningham. An on-line evaluation framework for recommender systems. In Proc. of Workshop on Personalization and Recommendation in E-Commerce Malaga (2002).
7. Woerndl, Schueller and Wojtech. A Hybrid Recommender System for Context-aware Recommendations of Mobile Applications. In Proc. of Workshop in ICDE ’07 (2007).
8. Yan and Chen. AppJoy - Personalized Mobile Application Discovery. In Proc. of MobiSys ’11 (2011).