

# Incorporating Context and Trends In News Recommender Systems

Andreas Lommatzsch, Benjamin Kille, and Sahin Albayrak

@INRA WS 2017, Leipzig, Germany, August 23<sup>rd</sup>, 2017

# News in a rapidly changing world

## Motivation

- Recommender Systems have been developed as powerful tools
- Popular application domains: Entertainment (movies, books, music), and online shops
- Most recommender system use **Collaborative Filtering**
- Trained on **large static** datasets (describing user-item interactions)



# News in a rapidly changing world

## Motivation

---

### ► News

- **Continuous changes** in set of news items, **frequent updates** in news items
- News must provide new/unknown information
- Users are interested in a wide spectrum of topics, e.g. unexpected **breaking news**
- The relevance of news depends on the **context**
- Most news are published online – users do not have to register explicitly

### ► Challenge

- Traditional CF-based approaches do not work well – due to the **cold-start problem** (item/user)
- The changes in the news sets require **continuous** model **updates**
- **Context** and user **habits** must be considered





# Incorporating Context and Trends In News Recommender Systems

## Objectives

---

### ► Objectives

- Analyze and optimize models for **recommending news**
- Develop models for learning recommender models likely to **fit best the near future**
- Models for incorporating **contexts** and **trends**





# Incorporating Context and Trends In News Recommender Systems

## Outline

---

- ▶ **Motivation**
- ▶ **The analyzed News Recommendation Scenario**
- ▶ **Approach**
- ▶ **Evaluation**
- ▶ **Conclusion**



## Problem Description

### NewsREEL - Overview

#### ► The NewsREEL Scenario

- Recommending News
- Online News Portals
- **Live user feedback**
- Evaluation with respect to CTR (Click-Through-Rate)
- Ensure **Technical constraints**

# News Portal Name

## Connecting the World



Recommending news articles is a challenging task due to the continuous changes in the set of available news articles and the context-dependent preferences of users.

Traditional recommender approaches are optimized for the analysis of static data sets. In news recommendation scenarios, characterized by continuous changes, high volume of messages, and tight time constraints, alternative approaches are needed. In this work we present a highly scalable recommender system optimized for the processing of streams. We evaluate the

system in the CLEF NewsREEL challenge. Our system is built on Apache Spark enabling the distributed processing of recommendation requests ensuring the scalability of our approach. The evaluation of the implemented system shows that our approach is suitable for the news recommendation scenario and provides high-quality results while satisfying the tight time constraints.

**Keywords**  
Apache Spark, stream recommender, distributed

### Also interesting

Recommendation 1

██████████  
██████████  
██████████

Recommendation 3

██████████  
██████████  
██████████

Recommendation 5

██████████  
██████████  
██████████

Recommendation 2

██████████  
██████████  
██████████

Recommendation 4

██████████  
██████████  
██████████

Recommendation 6

██████████  
██████████  
██████████



#### Tweets

- ██████████  
██████████
- ██████████  
██████████
- ██████████  
██████████
- ██████████  
██████████

#### Advertisement

██████████  
██████████  
██████████

#### Social Media

Twitter Google+ YouTube facebook

## Problem Description

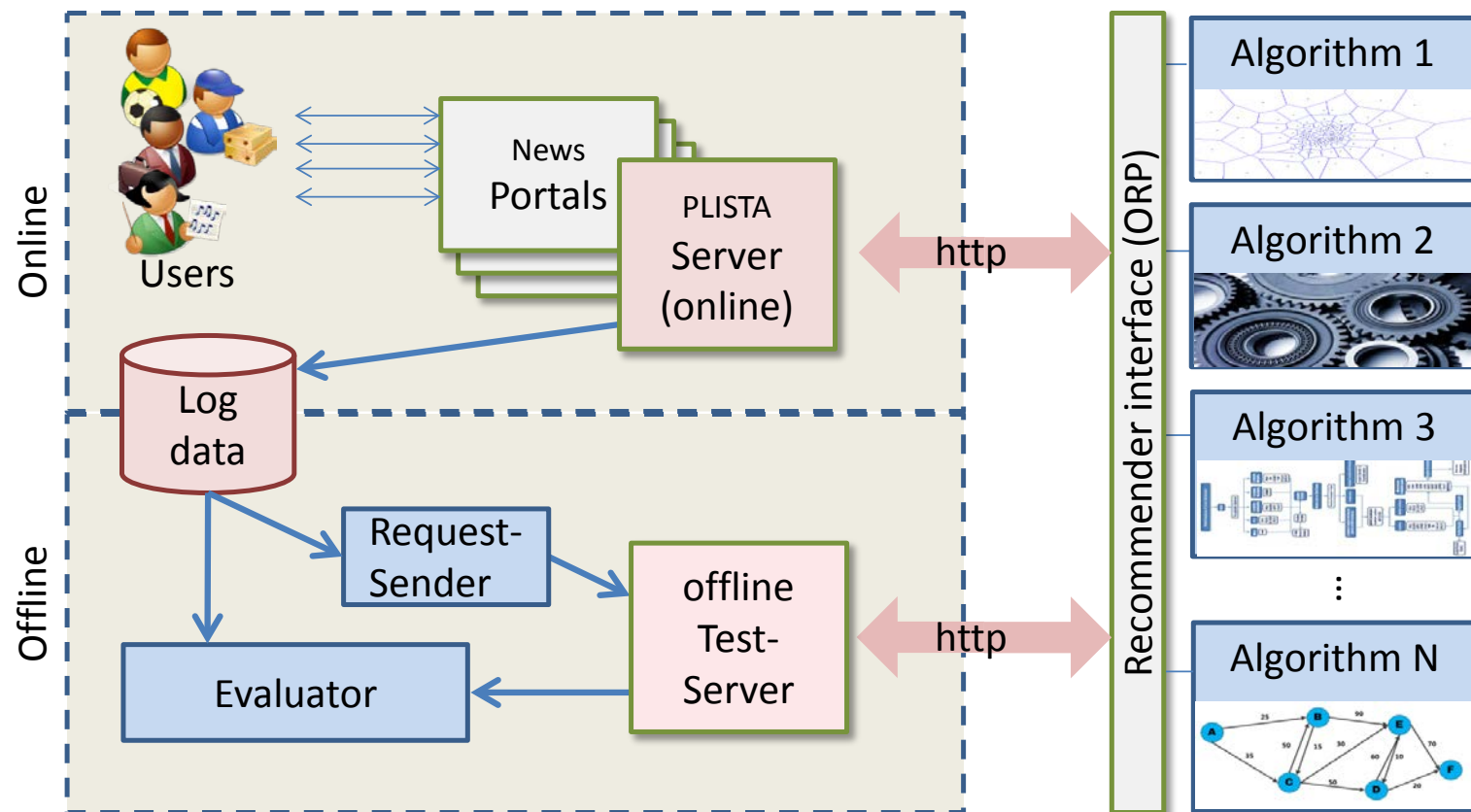
### NewsREEL – Challenge Architecture

#### ► The NewsREEL challenge architecture

- **Online** – Live feedback | **Offline** – Replay log data
- http-based communication, JSON-formatted data

#### ► Message types:

- Item updates
- Impressions
- Recommendation requests
- Clicks
- Errors



## Problem Description

### NewsREEL – Challenge Architecture

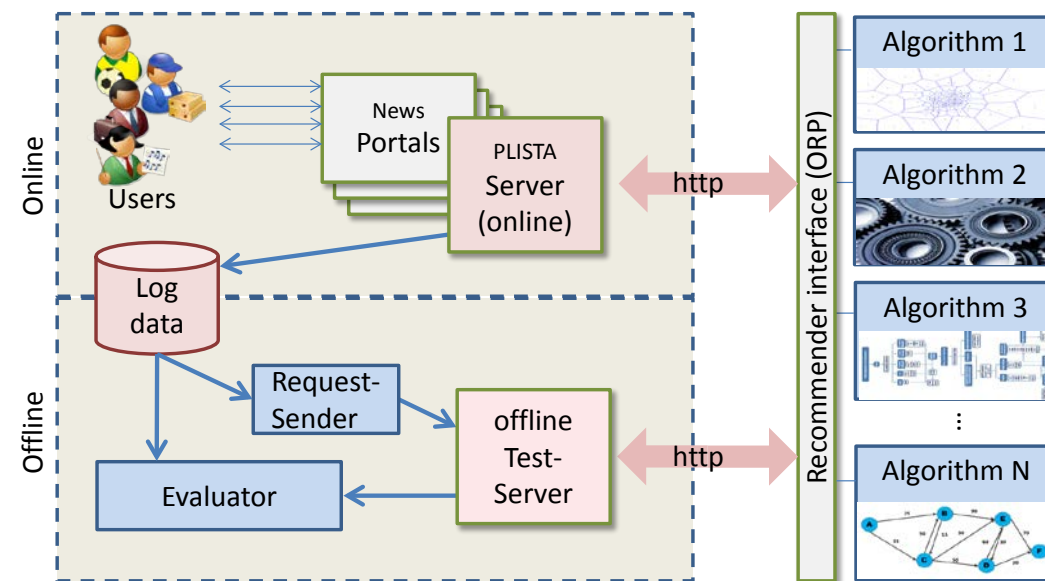
The NewsREEL challenge key figures

#### ► Online

- 3-5 new portals
- 40 messages per second (typically)  
→ up to 200 messages per second possible

#### ► Offline

- Data stream recorded over 2 month
- 100 GB JSON data
- Item creates, item updates, impressions, recommendation requests, clicks, errors





## Problem Description

### NewsREEL - Overview

---

#### ► The NewsREEL Scenario

- **Online / offline** (academic and industry-style evaluation)
- **Live** evaluation / Large dataset – **real world data**
- **Stream**-based recommendation (time and context as success-critical aspects)
- **Dynamic** sets of users and items
- **Multi-dimensional** benchmarking

#### ► More Details available at:

- <http://www.clef-newsreel.org/>

News Portal Name

## Connecting the World

Recommendation 1 Recommendation 3 Recommendation 5  
Recommendation 2 Recommendation 4 Recommendation 6

Tweets

Advertisement

Social Media

Twitter Google+ YouTube facebook

9



# Incorporating Context and Trends In News Recommender Systems

## Outline

---

- ▶ **Motivation**
- ▶ **The analyzed News Recommendation Scenario**
- ▶ **Data Analysis**
- ▶ **Approach and Evaluation**
- ▶ **Conclusion**



## Data - Analysis

### NewsREEL – Data Analysis

---

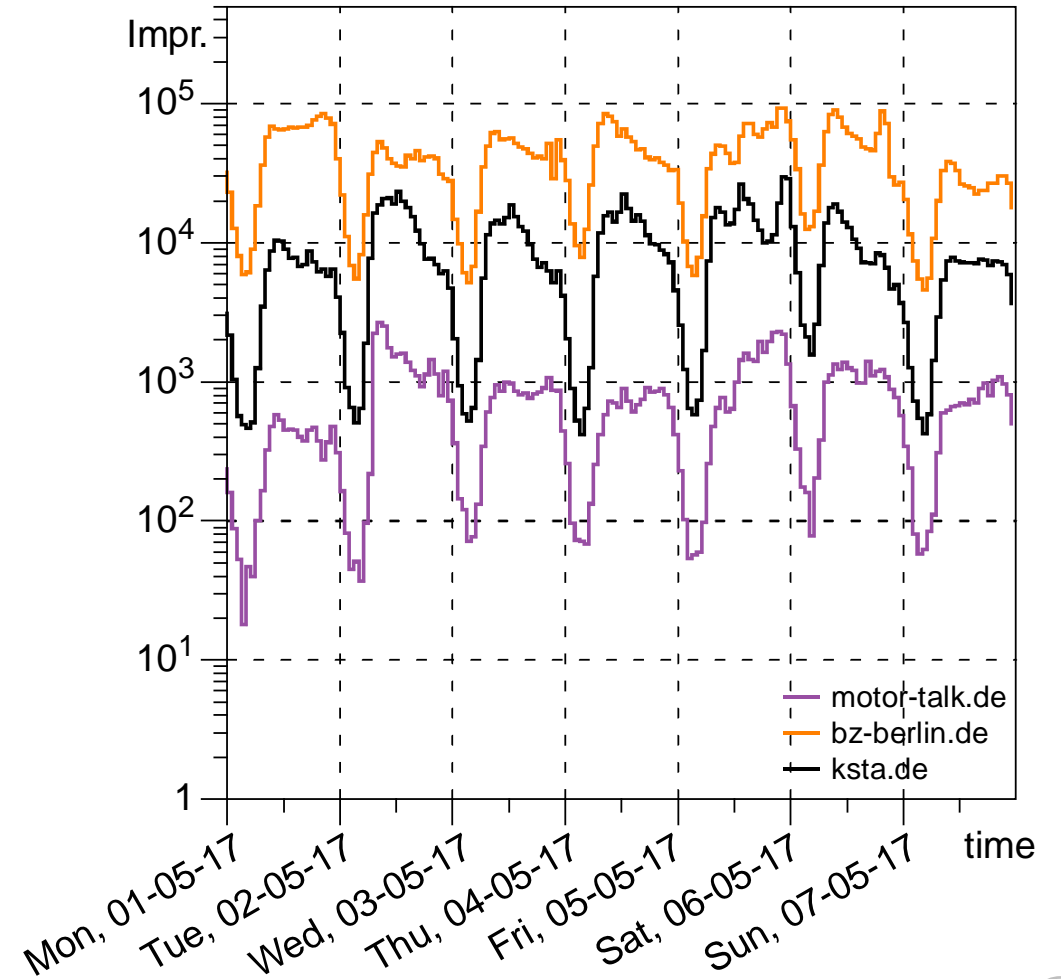
- ▶ **Data Analysis** with respect to **context** and temporal dynamics (**trends**)
  - Number of interactions in the system
  - Number of accepted recommendations
  - Device usage in the system
  - Item popularity
  - Item lifecycle
  - User reading preferences



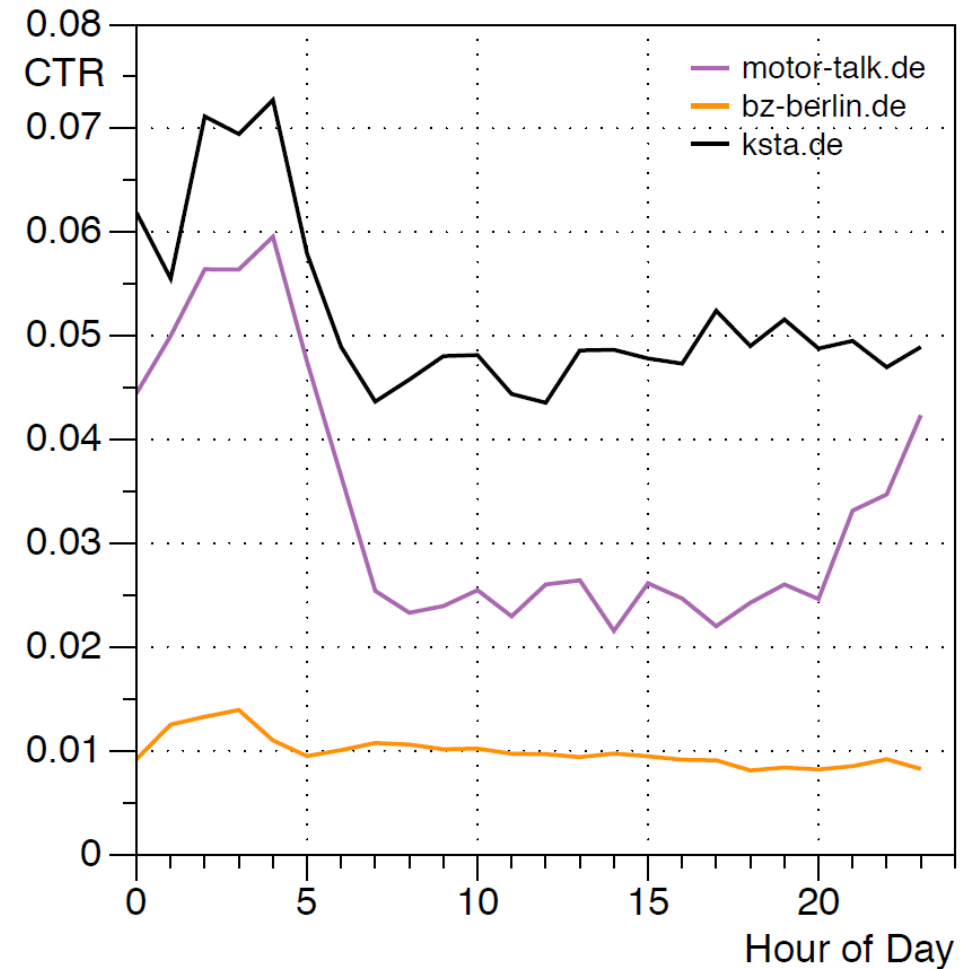
► We analyze the number of **impressions over one week**

► Observations:

- Huge changes in the number of impressions in the week
- Day/night pattern
- Working day / week end pattern
- Highly domain dependent



- ▶ We analyze the fraction of **accepted recommendations over the day**
- ▶ Observations:
  - The CTR highly depends on the time and the domain
  - At night the CTR is higher than during the day (for most portals)



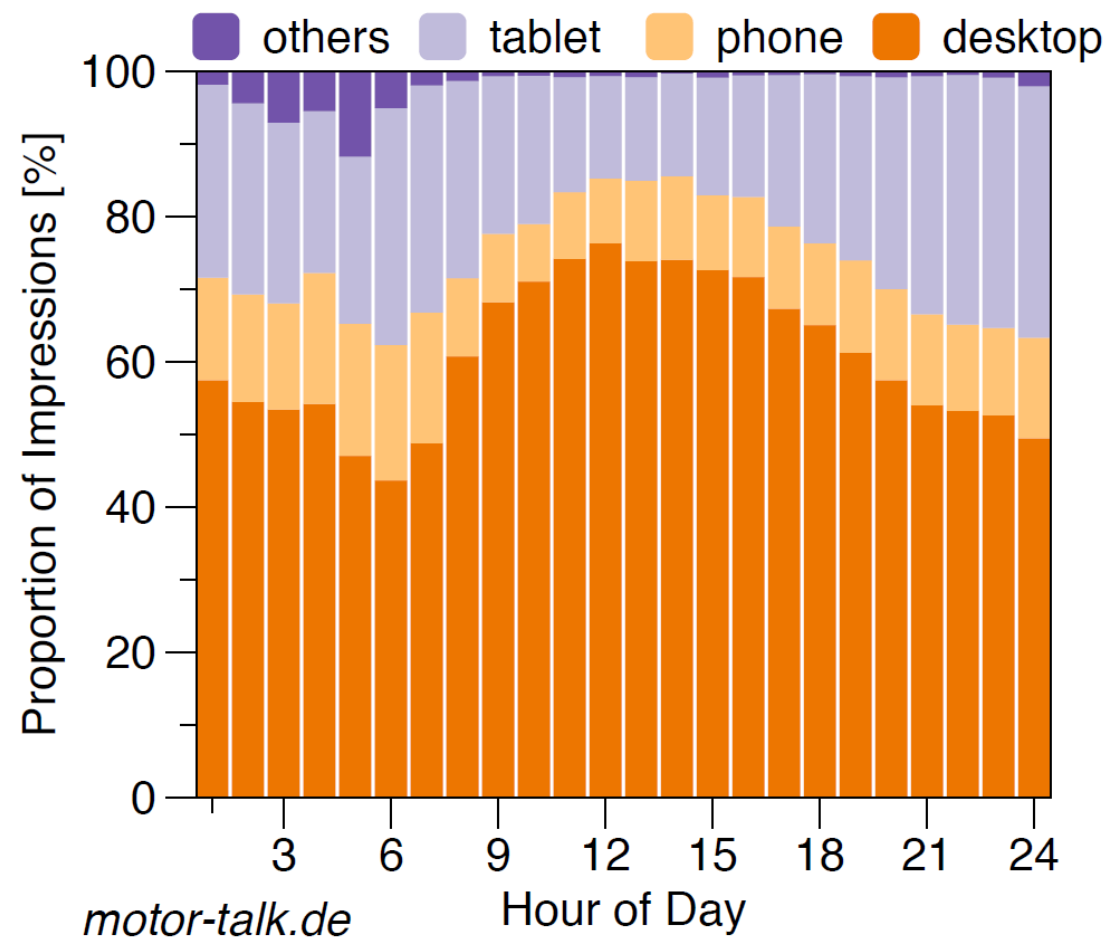


## Data - Analysis

### NewsREEL – Data Analysis 3

---

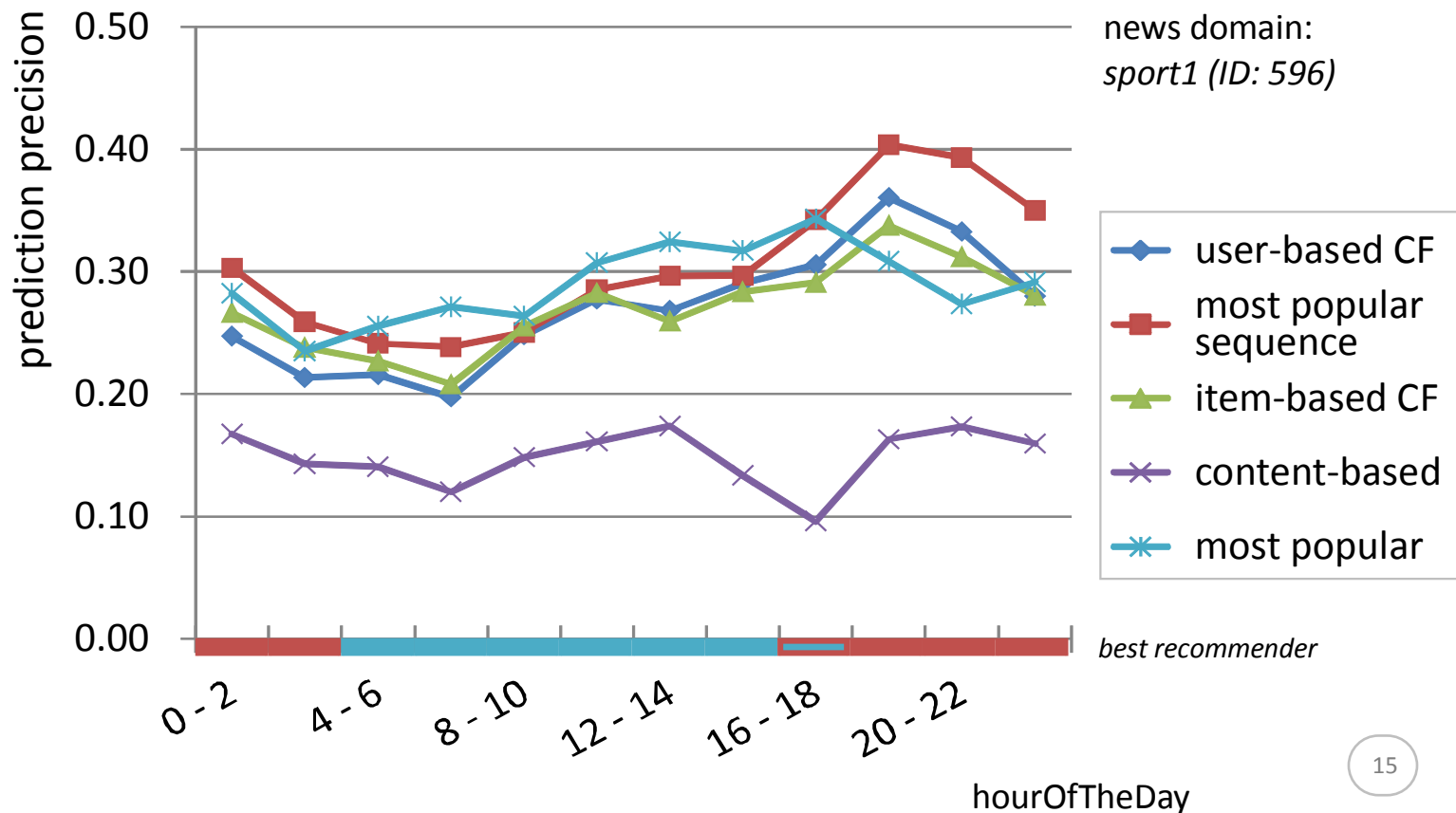
- ▶ How users consume news?  
What **devices** are mostly used?
- ▶ Observations:
  - The proportion between the used devices changes significantly over the day.
  - The device used typically influences the user preferences and the presentation of recommendations.



- ▶ What recommender algorithm performs best?  
**What type of recommendations** do users expect.

- ▶ Observations:
  - The recommender performance strongly depends on the time of day
  - Example: At night users tend to have a higher interests in longer news sequences

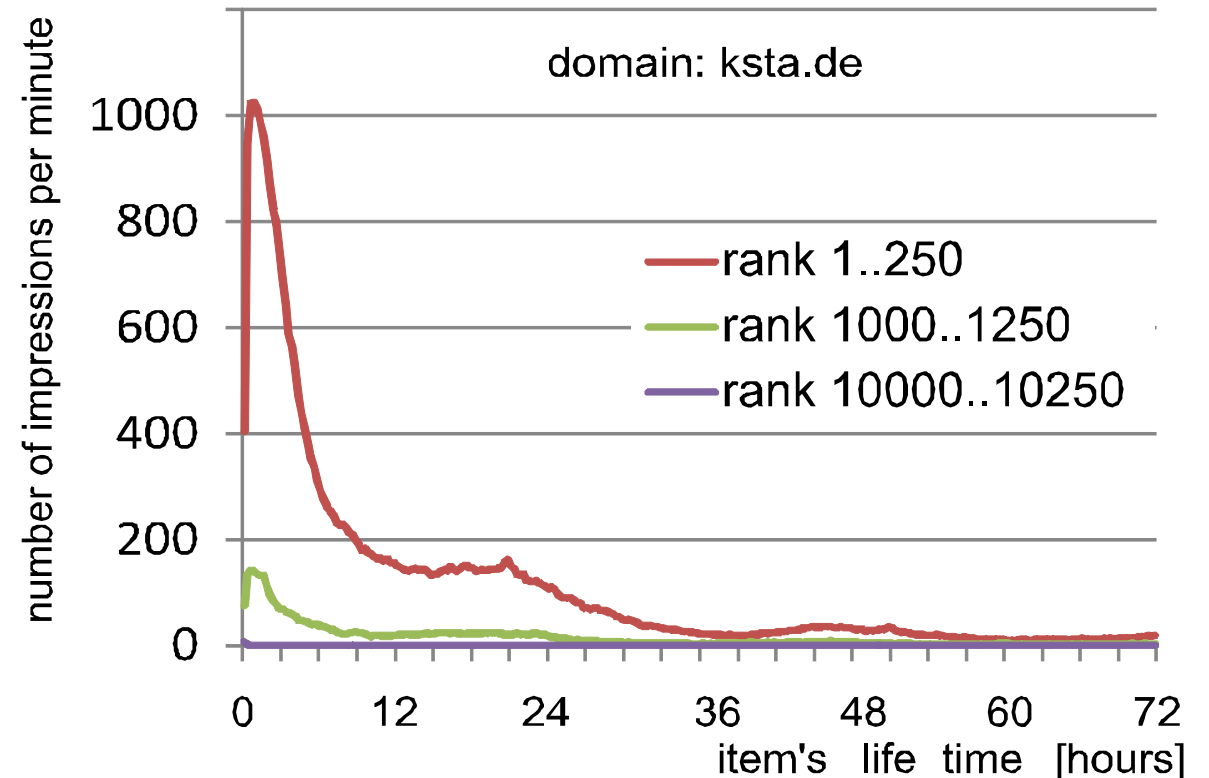
How does the performance depend on the hour of the day?



- ▶ How does the interest in specific items change over time?  
How does the **typical item lifecycle** look like?

- ▶ Observations:

- Items show a specific lifecycle
- After the release the interest in the news items grows fast,
- Having reached the maximum, the number of impressions decays exponentially



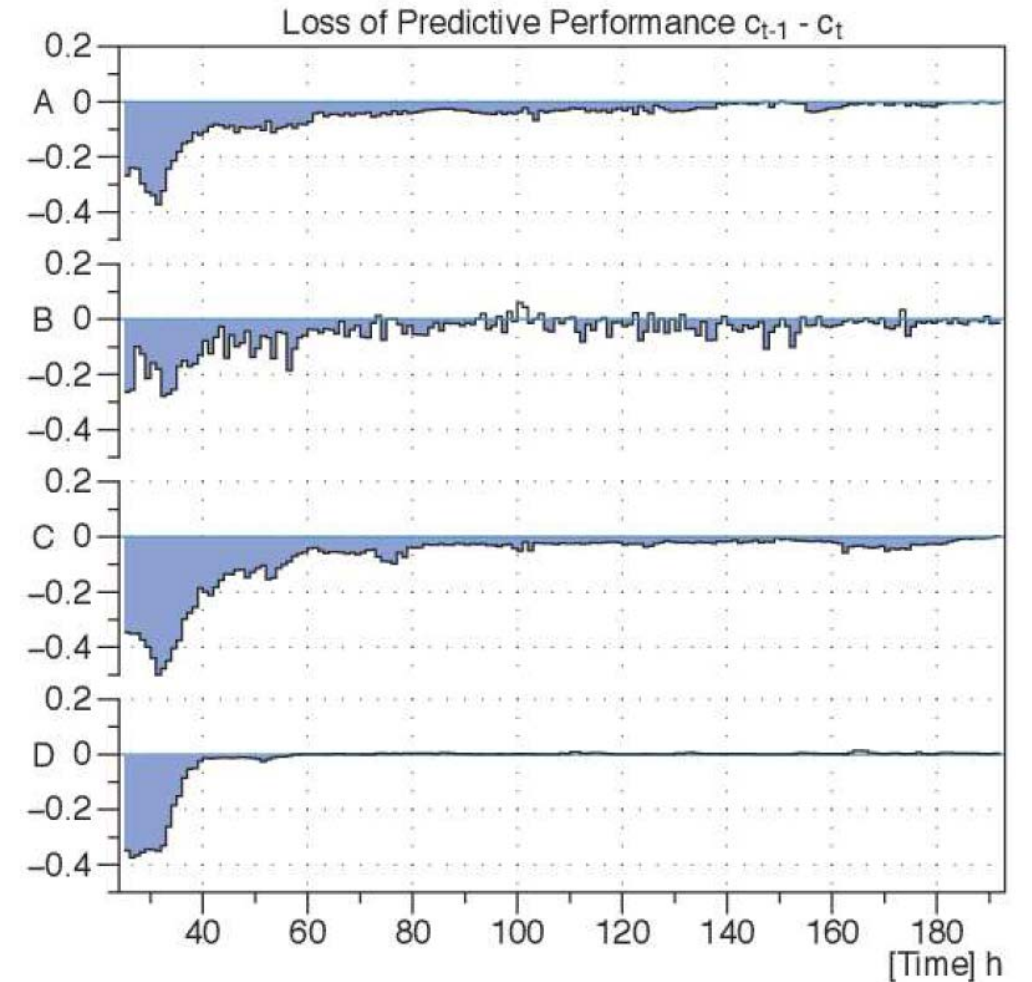


## Data - Analysis

### NewsREEL – Data Analysis 6

---

- ▶ How does the set of recommendable **items changes** over time?
- ▶ Observations:
  - We analyze the freshness of the recommender model
  - We analyze the loss of predictive performance



#### ▶ Conclusion

- **Context** has an high influence of the user behavior / user preferences
- **Dynamics** in the set of items
- Static recommender models do not cover the requirements

#### ▶ Research Questions:

- How to optimize models for incorporating the context
- How recommender models can considering the item lifecycle / trends in the item set





# Incorporating Context and Trends In News Recommender Systems

## Outline

---

- ▶ **Motivation**
- ▶ **The analyzed News Recommendation Scenario**
- ▶ **Data Analysis**
- ▶ **Approach and Evaluation**
- ▶ **Conclusion**

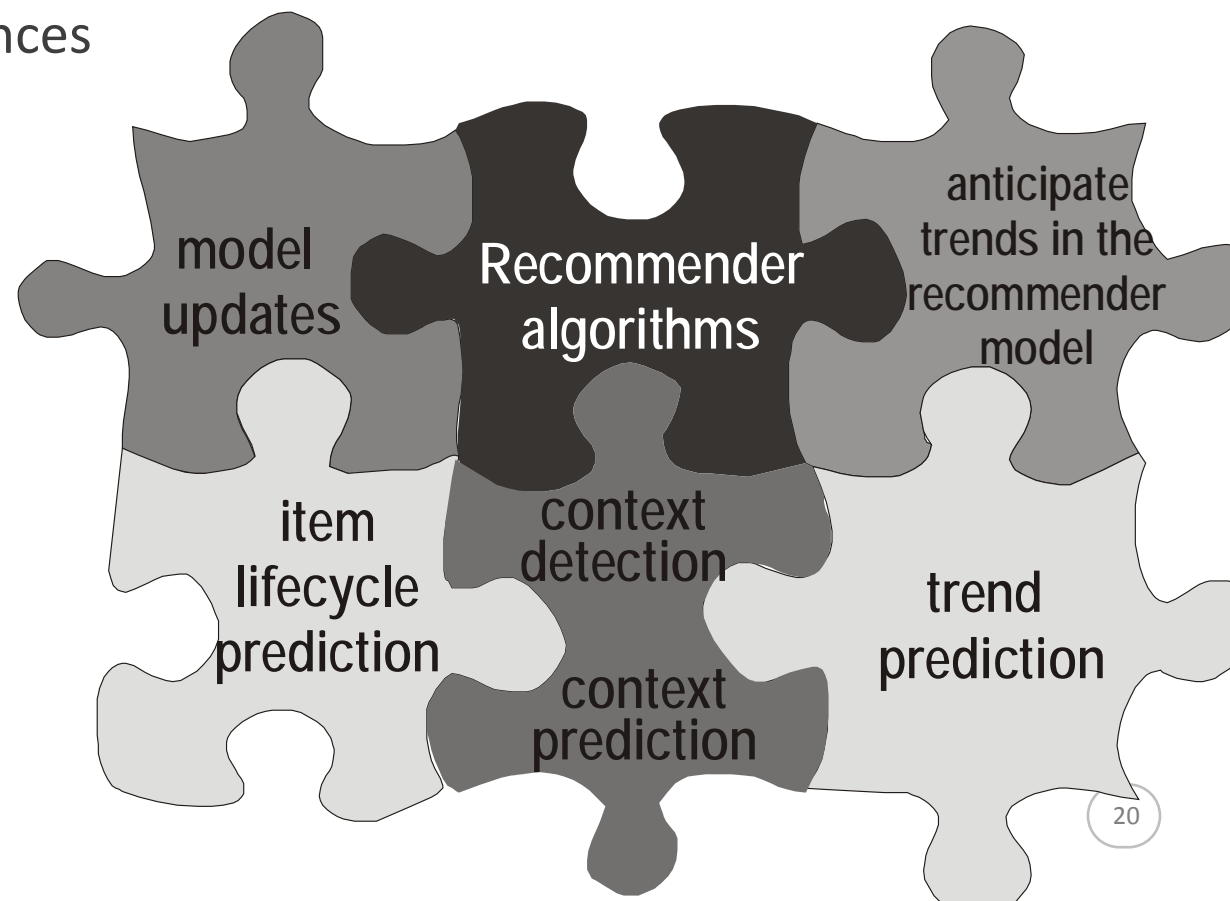


## Approach

### Models for Predicting the Context and Incorporating Trends

---

- ▶ Objective:
  - Study approaches for
    - Incorporating time-dependent preferences
    - The item life cycle
    - Adapting to current trends



## Approach

### Method 1: Idea

---

#### Method 1: Handle Data **streams** - Ensuring the model **freshness**

##### ▶ Idea:

- Aggregate stream data in batches
- Use traditional recommender frameworks optimized for sets
- Scheduler-based model updates

##### ▶ Challenge

- Components for parallelizing batch-building model-building, and computing recommendations
- Definition of optimal batch sizes
- Model building may slow down the recommender



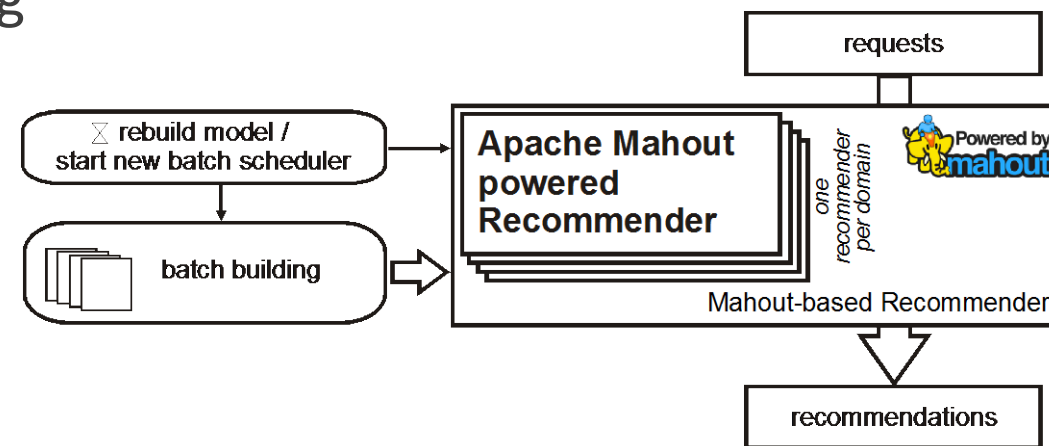
## Approach

### Method 1: Evaluation – Discussion

#### Method 1: Ensuring the model freshness

##### ► Evaluation - Discussion:

- Works well in the typical news recommendation scenario
- Use existing, mature recommender components
- Delayed incorporation of trends
- Cold-start problem cause by new batches
- Load peaks when caused by model rebuilding



## Approach

### Method 2: Idea

---

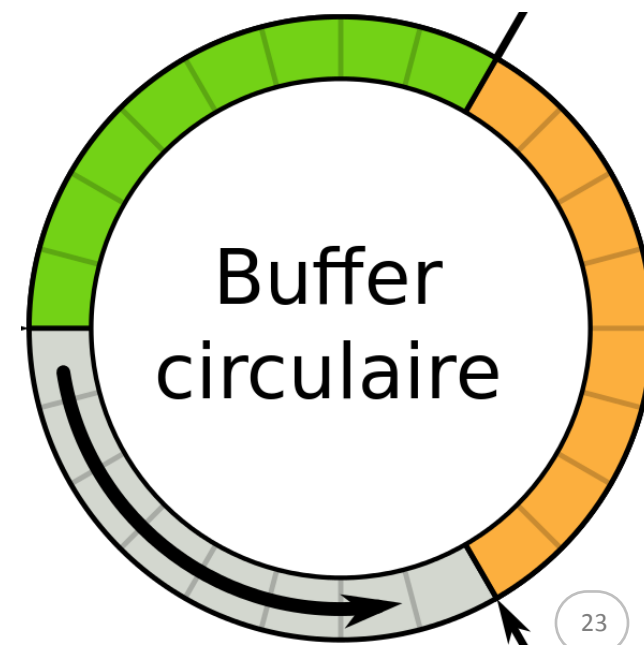
#### Method 2: Handle Data streams - Ensuring the **continuous** model **freshness**

##### ► Idea:

- Continuous updates in order to ensure fast adaptation to new trends
- Fixed model size (remove data when new data is added)

##### ► Challenge

- Adapt algorithms so that continuous updates are supported
- Find optimal model size considering the context
- Handle the concurrency of model updates and model deployment





## Approach

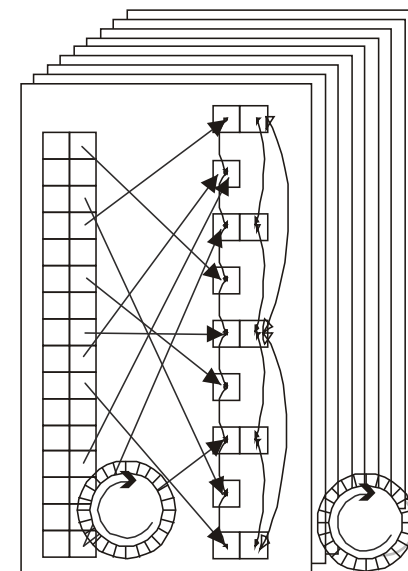
### Method 2: Evaluation – Discussion

---

#### Method 2: Ensuring the continuous model freshness

##### ► Evaluation - Discussion

- Overcomes the most problems of batch-based model updates
- Constant load in model adaptation
- Limited complexity of the supported recommender models
- Adapts faster than batch updates, but does not anticipate trends and changes in the context





## Approach

### Method 3: Idea

---

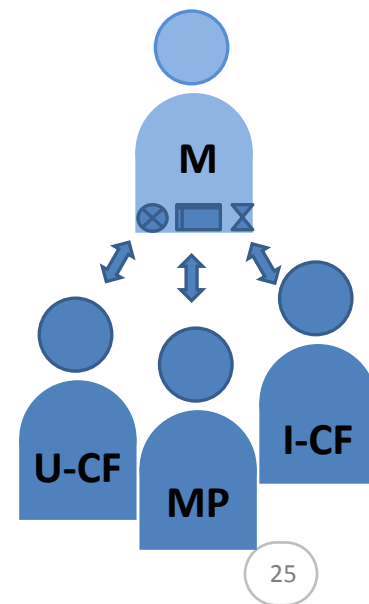
#### Method 3: Recommender ensembles for specific contexts

##### ► Idea:

- Define an ensemble
- Learn an algorithm for each relevant context
- Learn what algorithm should be selected considering context and trends

##### ► Challenge:

- What are relevant contexts?
- How to ensure that the context is relevant over time
- Handle unexpected/unusual situations
- How to ensure that enough training data exists (fine-grained contexts vs. significant amount of training data)



## Approach

### Method 3: Evaluation – Discussion

#### ► Method 3: Recommender Ensembles for specific contexts

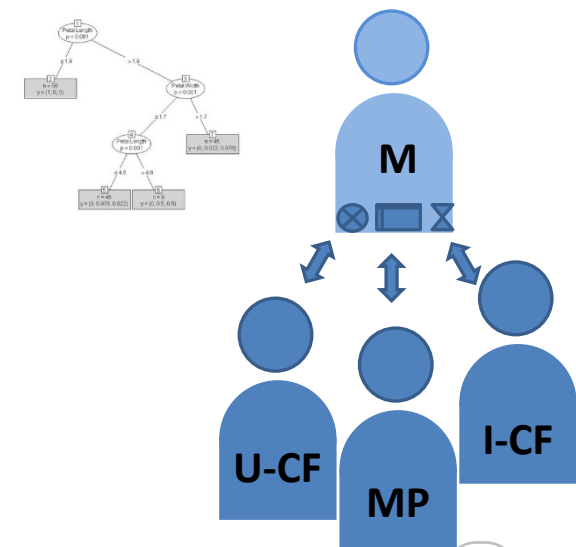
#### ► Evaluation:

- Works well for “normal” days
- Cover well daily and weekly usage patterns
- Anticipate periodical usage pattern

#### ► Discussion:

- Ensemble lead to a significantly higher complexity, several different algorithms must be trained
- Rare events/contexts are often not well considered (due to limited training data)

```
getDomain = cio.de: most-popular (3985.0/325.0)
getDomain = ksta.de
| workingDay = yes
| | getNumberOfRequestedResults = one
| | | getHour <= 8: user-based-CF (1489.0/927.0)
| | | getHour <= 10: mp-sequence (926.0/632.0)
| | | getHour <= 14: item-based CF (1633.0/1061.0)
| | | getHour <= 18: mp-sequence (1103.0/720.0)
| | | getHour > 18: user-based-CF (710.0/435.0)
| | | getNumberOfRequestedResults = more
| | | getHour <= 8
| | ...
| ...
...
```



## Approach

### Method 4

---

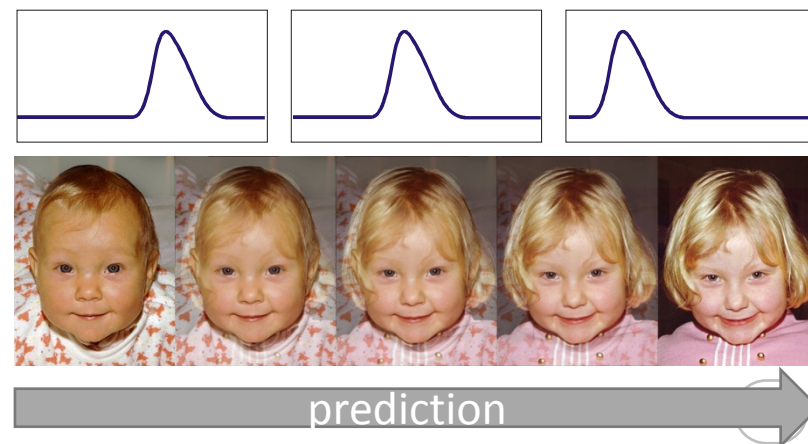
#### Method 4: Consider the item lifecycle

##### ► Idea:

- Traditionally, recommender are trained on past data
- Knowing the item characteristic lifecycle for all items, we can predict the item characteristic in the near future
- We can build a model fitting the near future best

##### ► Challenges

- Train a model allowing us to predict the lifecycle of all items
- Fit the model parameters



## Approach

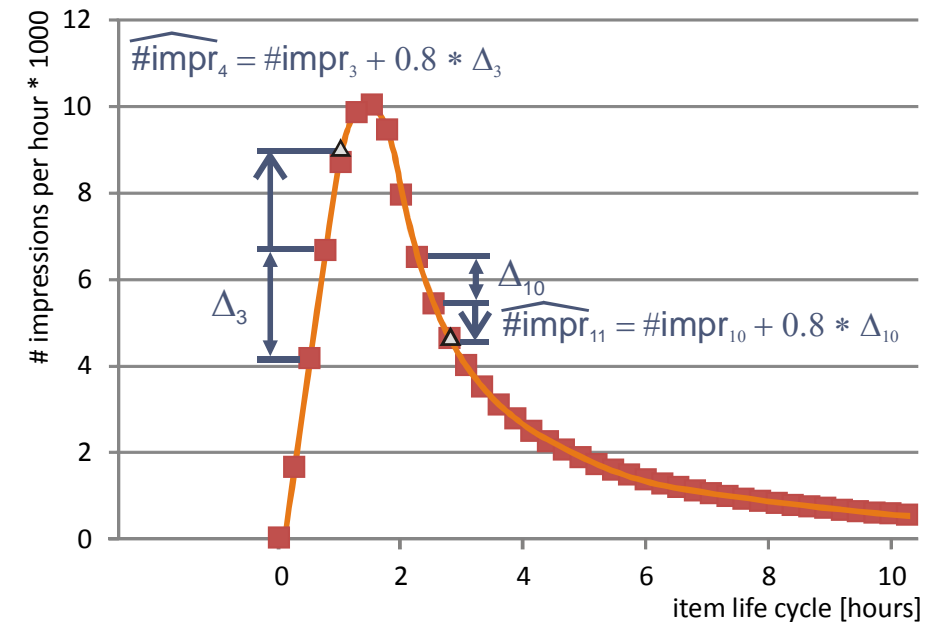
### Method 4

---

#### Method 4: Consider the item lifecycle

##### ► Evaluation – Discussion

- Anticipate observed / periodically trends
- Avoid the problem of late adaptation
- The model cannot predict breaking events (untypical events)
- Models computed based on predicted data are often noisy
- An extended evaluation is needed



## Approach and Evaluation

### Discussion

---

#### ► Discussion

- Different approaches tailored for solving specific challenges
- Each approach focus on one specific aspect – each approach has **specific strength and weaknesses**
- The **combination is promising** but reduces the number of training data for each context/situation
- Extended evaluation is needed, is planned for the next month



## Outline

---

- ▶ **Motivation**
- ▶ **The analyzed News Recommendation Scenario**
- ▶ **Data Analysis**
- ▶ **Approach and Evaluation**
- ▶ **Conclusion**

# Incorporating Context and Trends In News Recommender Systems

## Conclusion

---

### ► Results

- Recommender system require special methods for considering contexts and anticipating trends
- We have presented four methods (each focusing on one aspect)
- Improved performance compared with traditional static recommender models
- Combination of different methods is promising
- “news” are difficult to predict, user preferences are changing  
=> news recommendation is challenging

### ► Future Work

- Extended Evaluations
- More combinations
- Consider additional types of items, e.g., blogs



# CLEF-NEWSREEL

## NEWS RECOMMENDATION EVALUATION LAB



HOME TASKS HOW TO PARTICIPATE PUBLICATIONS ORGANISATION PREVIOUS CAMPAIGNS CLEF 2017 PROGRAM

### Overview

Commercial providers of information access systems (such as Amazon or Google) usually evaluate the performance of their algorithms by observing how large numbers of their customers interact with different instances of their services. Unfortunately, due to the lack of access to large-scale systems, university-based research is struggling to catch up with this large-scale evaluation methodology. NewsREEL, short for News Recommendation Evaluation Lab, aims to bridge this "evaluation gap" between Academia and Industry.

NewsREEL is organised as a campaign-style evaluation lab of CLEF 2017 and addresses the following information access task:

**Whenever a visitor of an online news portal reads a news article on their side, the task is to recommend other news articles that the user might be interested in.**

NewsREEL offers two tasks to study this use case. The first task, **NewsREEL Live**, implements the idea of a "living lab" where the provider of a recommendation service provides access to its infrastructure and user base. The second task, **NewsREEL Replay**, replays a live setting using the recommender system reference framework Idomaar.

By providing this service for millions of users, the recommendation scenario requires solutions to significant research challenges, such as processing information in real-time, handling vast amount of data, and providing suitable recommendations. By providing access to the infrastructure of a company, we offer professional and students the opportunity to develop systems that are in high demand in industry, while at the same time allowing them to familiarize themselves with the academic practice of evaluation of information access systems.

Please follow @CLEFNewsREEL for further updates.

See also:



### Important Dates

- Labs registration opens: 4 November 2016
- Registration Closes: 21 April 2017
- Test Period 1: 13-19 March 2017
- Test Period 2: 27 March-3 April 2017
- Evaluation Period: 24 April - 7 May 2017
- Submission of Working Notes: 2 June 2017
- Feedback on Working Notes: 16 June 2017
- Camera Ready Working Notes due: 3 July 2017
- CLEF 2017: 11-14 September 2017

### @clefnewsreel

RT @FTHopf: Working notes papers of @CLEFNEWSREEL 2017 are now available online at [#recsys17](https://t.co/lmXM0bfot4) from Twitter at ClefNewsREEL. Participated in #newsreel this year? Don't forget to tell us about your experience at #clef2017. Submit your working notes 2 June 2017, 01:15:41 PM May 30, 2017

# <http://www.clef-newsreel.org/>

▶ Do you have own algorithms? Evaluate algorithms online live – **it's free!**





 **DAI-Labor**  
TU Berlin

DAI-Labor, Sekr. TEL 14  
Ernst-Reuter-Platz 7  
D-10587 Berlin, Germany



Get In Touch



[Andreas.Lommatzsch@dai-labor.de](mailto:Andreas.Lommatzsch@dai-labor.de)  
[Benjamin.Kille@dai-labor.de](mailto:Benjamin.Kille@dai-labor.de)

<http://www.clef-newsreel.org>