Recommender Systems and Learning Analytics in TEL

Hendrik Drachsler
Centre for Learning Sciences and Technology
Open University of the Netherlands
Goals of the lecture

1. Crash course Recommender Systems (RecSys)
2. Overview of RecSys in TEL
3. Open research issues for RecSys in TEL
4. TEL RecSys and Learning Analytics
Introduction into Recommender Systems

Introduction
Objectives
Technologies
Evaluation
Introduction:: Application areas

Application areas

• E-commerce websites (Amazon)
• Video, Music websites (Netflix, last.fm)
• Content websites (CNN, Google News)
• Information Support Systems

Major claims

• Highly application-oriented research area, every domain and task needs a specific RecSys

• Always build around content or products they never exist as on their own
Introduction::Definition

Using the **opinions of a community** of users to help **individuals** in that community to identify more effectively **content of interest** from a potentially overwhelming set of choices.

Introduction:: Definition

Using the opinions of a community of users to help individuals in that community to identify more effectively content of interest from a potentially overwhelming set of choices.


Any system that produces personalized recommendations as output or has the effect of guiding the user in a personalized way to interesting or useful objects in a large space of possible options.

Introduction:: Example
Introduction:: Example
Introduction::Example
Introduction::Example

Today's Recommendations For You

Tell us more about your likes and dislikes by rating products you have an opinion about. The more we know about your interests, the more we can do to improve your recommendations. Learn more.

Search for items to rate: All Departments

1. Use the search box above to find your favorite books, movies, albums, artists, authors and brands.
2. Tell us what you think of the items we return for your search by rating the item or telling us you already own them.
3. Repeat until the Recommendations you find in Your Amazon.com reflect your tastes and interests.
Introduction::Example
Introduction::Example
Introduction:: Example

Hello, Hendrik Drachsler. We have recommendations for you. (Not Hendrik?)

Customer Reviews

Average Customer Rating

- 5 star: (153)
- 4 star: (38)
- 3 star: (10)
- 2 star: (8)
- 1 star: (4)

Appearance ...................... ★★★★★ (277)  
Portability ..................... ★★★★★ (270)  
Bluetooth compatibility ....... ★★★★★ (265)  
Battery life ..................... ★★★★★ (244)  

See and rate all 6 attributes.
Introduction:: Example

Hello, Hendrik Drachsler. We have recommendations for you. (Not Hendrik?)

Customers Who Bought This Item Also Bought

Today's Recommendations For You

Here's a daily sample of items recommended for you. Click here to see all recommendations.

- Audible for Android
  - Rating: 4.5/5
  - Price: $0.00
  - Fix this recommendation

- Apple Wireless Keyboard (Retail Packaging)
  - Rating: 4/5 (273)
  - Price: $0.00
  - Click for details
  - Fix this recommendation

- HootSuite
  - Rating: 4/5 (2)
  - Price: $0.00
  - Fix this recommendation

- Flip Video Carrying Case for Flip Video Ultra S
  - Rating: 4/5 (19)
  - Price: $11.95
  - Fix this recommendation

- Case Logic UN298-3 Neoprene Compact Camcorder Case
  - Rating: 4/5 (30)
  - Price: $12.99
  - Fix this recommendation

artists, authors and brands.

telling us you already own them.
What did we learn from the small exercise?

- There are different kinds of recommendations
  a. People who bought X also bought Y
  b. There are more advanced personalized recommendations

- When registering, we have to tell the RecSys what we like (and what not). Thus, it requires information to offer suitable recommendations and it learns our preferences.
Introduction: The Long Tail

"We are leaving the age of information and entering the age of recommendation".

Introduction:: Age of RecSys?

...10 minutes on Google.
Introduction:: Age of RecSys?

...10 minutes on Google.
Introduction:: *Age of RecSys?*

... another 10 minutes, research on RecSys is becoming main stream.

**Some examples:**
- ACM RecSys conference
- *ICWSM*: Weblog and Social Media
- *WebKDD*: Web Knowledge Discovery and Data Mining
- *WWW*: The original WWW conference
- *SIGIR*: Information Retrieval
- *ACM KDD*: Knowledge Discovery and Data Mining
- *LAK*: Learning Analytics and Knowledge
- *Educational data mining* conference
- *ICML*: Machine Learning
- ...

... and various workshops, books, and journals.
Objectives of RecSys

The idea is to pick from my previous list 20-50 movies that share similar audience with "Taken", then how much I will like depends on how much I liked those early movies – In short: I tend to watch this movie because I have watched those movies … or People who have watched those movies also liked this movie (Amazon style)

probabilistic combination of

– Item-based method
– User-based method
– Matrix Factorization
– (May be) content-based method
Objectives::Aims

- Converting Browsers into Buyers
- Increasing Cross-sales
- Building Loyalty

The idea is to pick from my previous list of 20-50 movies that share similar audience with "Taken," then how much I will like this movie depend on how much I liked those early movies — In short: I tend to watch this movie because I have watched those movies … or People who have watched those movies also liked this movie (Amazon style)

- Item-based method
- User-based method
- Matrix Factorization
- (May be) content-based method

Find good items
presenting a ranked list of recommendations.

Find *all* good items
user wants to identify *all* items that might be interesting, *e.g.* medical or legal cases

The idea is to pick from my previous list 20-50 movies that share similar audience with "Taken", then how much I will like depends on how much I liked those early movies – In short: I tend to watch this movie because I have watched those movies … or – People who have watched those movies also liked this movie (Amazon style)

Probabilistic combination of
- Item-based method
- User-based method
- Matrix Factorization
- (May be) content-based method

Find good items presenting a ranked list of recommendations.

Find all good items user wants to identify all items that might be interesting, e.g. medical or legal cases

Receive sequence of items sequence of related items is recommended to the user, e.g. music recommender

Annotation in context predicted usefulness of an item that the user is currently viewing, e.g. links within a website

The idea is to pick from my previous list 20-50 movies that share similar audience with "Taken", then how much I will like this movie depend on how much I liked those early movies – In short: I tend to watch this movie because I have watched those movies … or People who have watched those movies also liked this movie (Amazon style)

Probabilistic combination of
- Item-based method
- User-based method
- Matrix Factorization
- (May be) content-based method

Find good items
presenting a ranked list of recommendations.

Receive sequence of items
sequence of related items is recommended to the user, e.g. music recommender

There are more tasks available...

Find all good items
user wants to identify all items that might be interesting, e.g. medical or legal cases

Annotation in context
predicted usefulness of an item that the user is currently viewing, e.g. links within a website

RecSys Technologies

1. Predict how much a user may like a certain product

2. Create a list of Top-N best items

3. Adjust its prediction based on feedback of the target user and like-minded users

RecSys Technologies

1. Predict how much a user may like a certain product

2. Create a list of Top-N best items

3. Adjust its prediction based on feedback of the target user and like-minded users

Just some examples there are more technologies available.

User-based filtering
(Grouplens, 1994)

Take about **20-50** people who share **similar taste** with you, afterwards predict how much you might like an item depended on how much the others liked it.

You may like it because your “friends” liked it.
Technologies:: Collaborative filtering

User-based filtering  
(Grouplens, 1994)

Take about 20-50 people who share similar taste with you, afterwards predict how much you might like an item depended on how much the others liked it.

You may like it because your “friends” liked it.

Item-based filtering  
(Amazon, 2001)

Pick from your previous list 20-50 items that share similar people with “the target item”, how much you will like the target item depends on how much the others liked those earlier items.

You tend to like that item because you have liked those items.
Information needs of user and characteristics of items are represented in **keywords, attributes, tags** that describe past selections, e.g., TF-IDF.
The idea is to pick from my previous list 20-50 movies that share similar audience with "Taken", then how much I will like depends on how much I liked those early movies – In short: I tend to watch this movie because I have watched those movies ... or – People who have watched those movies also liked this movie (Amazon style)

Technologies:: Hybrid RecSys

Combination of techniques to overcome disadvantages and advantages of single techniques.

Advantages
• No content analysis
• Quality improves
• No cold-start problem
• No new user / item problem

Disadvantages
• Cold-start problem
• Over-fitting
• New user / item problem
• Sparsity
The idea is to pick from my previous list 20-50 movies that share similar audience with “Taken”, then how much I will like depends on how much I liked those early movies … or People who have watched those movies also liked this movie (Amazon style)

Technologies:: Hybrid RecSys

Combination of techniques to overcome disadvantages and advantages of single techniques.

**Advantages**
- No content analysis
- Quality improves
- No cold-start problem
- No new user / item problem

**Disadvantages**
- Cold-start problem
- Over-fitting
- New user / item problem
- Sparsity

Just some examples there are more (dis)advantages available.
Evaluation of RecSys

The idea is to pick from my previous list 20-50 movies that share similar audience with "Taken", then how much I will like depend on how much I liked those early movies – In short: I tend to watch this movie because I have watched those movies … or – People who have watched those movies also liked this movie (Amazon style)

probabilistic combination of

– Item-based method
– User-based method
– Matrix Factorization
– (May be) content-based method
Evaluation:: General idea

Most of the time based on performance measures ("How good are your recommendations?")

For example:

• Predict what rating will a user give an item?
• Will the user select an item?
• What is the order of usefulness of items to a user?

Evaluation:: Reference datasets

... and various commercial datasets.
Evaluation:: Approaches

1. Simulation

- User preference
- Prediction accuracy
- Coverage
- Confidence
- Trust
- Novelty
- Serendipity
- Diversity
- Utility
- Risk
- Robustness
- Privacy
- Adaptivity
- Scalability

2. User study

Dataset of interactions → Training set → Test set → Predictions

Test compare

+
**Evaluation::Metrics**

**Precision** – The portion of recommendations that were successful. (Selected by the algorithm and by the user)

**Recall** – The portion of relevant items selected by algorithm compared to a total number of relevant items available.

**F1** - Measure balances Precision and Recall into a single measurement.

**Evaluation::**  

**Metrics**

**Precision** – The portion of recommendations that were successful. (Selected by the algorithm and by the user)

**Recall** – The portion of relevant items selected by algorithm compared to a total number of relevant items available.

**F1** - Measure balances Precision and Recall into a single measurement.

<table>
<thead>
<tr>
<th></th>
<th>Recommended</th>
<th>Not recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Used</strong></td>
<td>True-positive</td>
<td>False-negative</td>
</tr>
<tr>
<td><strong>Not used</strong></td>
<td>False-positive</td>
<td>True-negative</td>
</tr>
</tbody>
</table>

Evaluation::Metrics

**Precision** – The portion of recommendations that were successful. (Selected by the algorithm and by the user)

**Recall** – The portion of relevant items selected by algorithm compared to a total number of relevant items available.

**F1** - Measure balances Precision and Recall into a single measurement.

<table>
<thead>
<tr>
<th>Measures</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>Precision (p)</td>
<td>$\frac{TP}{(TP + FP)}$</td>
</tr>
<tr>
<td>Recall (tpr)</td>
<td>$\frac{TP}{(TP + FN)}$</td>
</tr>
<tr>
<td>F1-Measure</td>
<td>$\frac{(2 \times tpr \times p)}{(tpr + p)}$</td>
</tr>
</tbody>
</table>

Evaluation::Metrics

**Precision** – The portion of recommendations that were successful. (Selected by the algorithm and by the user)

**Recall** – The portion of relevant items selected by algorithm compared to a total number of relevant items available.

**F1** - Measure balances Precision and Recall into a single measurement.

<table>
<thead>
<tr>
<th>Measures</th>
<th>Formula</th>
</tr>
</thead>
</table>

Just some examples there are more metrics available like MAE, RSME.

Evaluation::Metrics

Conclusion:
Pearson is better than Cosine, because less errors in predicting TOP-N items.

Evaluation:: Metrics

Conclusion:
Pearson is better than Cosine, because less errors in predicting TOP-N items.

Conclusion:
Cosine better than Pearson, because of higher precision and recall value on TOP-N items.

RecSys::TimeToThink

What do you expect that a RecSys in a MUP/PLE should do with respect to ...

- Aims
- Tasks
- Technology
- Evaluation

Blackmore’s custom-built LSD Drive
http://www.flickr.com/photos/rootoftwo/
Goals of the lecture

1. Crash course Recommender Systems (RecSys)

2. Overview of RecSys in TEL

3. Open research issues for RecSys in TEL

4. TEL RecSys and Learning Analytics
Recommender Systems for TEL

Introduction
Objectives
Technologies
Evaluation
Using the **experiences of a community** of learners to help **individual learners** in that community to identify more effectively **learning content** from a potentially overwhelming set of choices.

Find an appropriate recommendation system for a set of goals, tasks, limitations, and constraints. More accurately – choose the most appropriate system from a set of candidates.

Major claim: there is no silver bullet – a system that is both the most accurate, the fastest, the cheapest, …

We need to select for each application an appropriate recsys that fits its needs.

The Long Tail

The Long Tail of Learning

The Long Tail of Learning

Formal
- Planned
- Scheduled and assigned
- Common knowledge
- “Top down” and codified
- Semi-permanent
- Expert-created
- Expensive
- Pushed

Informal
- Ad Hoc
- Time of need, in the “flow”
- Specialized knowledge
- Emergent and collaborative
- Temporary or semi-permanent
- Crowd-sourced
- Low cost
- Pulled

TEL RecSys:: Technologies

OpenScout

STELLAR

ReMashed
Recommendations for Mash-ups

Organic.Edunet

apostle
learn @ work

VAA3R

TEN Competence

mavsel
mining, analysis and visualization based in social aspects of e-learning

funded by the Spanish Ministry of Science & Innovation, Reference TIN2010-21715-C02-01

ROLE
It is mandatory to specify at least one Web 2.0 service. If you specify more than one Web 2.0 service we can predict better recommendations for you.
RecSys Task: Find good items

Hybrid RecSys:
- Content-based on interests
- Collaborative filtering
The idea is to pick from my previous list 20-50 movies that share similar audience with "Taken", then how much I will like depends on how much I liked those early movies – In short: I tend to watch this movie because I have watched those movies … or – People who have watched those movies also liked this movie (Amazon style)

Find good items e.g. relevant items for a learning task or a learning goal

The idea is to pick from my previous list 20-50 movies that share similar audience with "Taken", then how much I will like depends on how much I liked those early movies – in short: I tend to watch this movie because I have watched those movies … or – people who have watched those movies also liked this movie (Amazon style).

TEL RecSys::Tasks

Find good items
e.g. relevant items for a learning task or a learning goal

Receive sequence of items
e.g. recommend a learning path to achieve a certain competence

Find good items
e.g. relevant items for a learning task or a learning goal

Receive sequence of items
e.g. recommend a learning path to achieve a certain competence

Annotation in context
e.g. take into account location, time, noise level, prior knowledge, peers around

Evaluation of TEL RecSys
Table 12.3: Implemented TEL recommender systems reported in literature

<table>
<thead>
<tr>
<th>System</th>
<th>Status</th>
<th>Evaluator focus</th>
<th>Evaluation roles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Altered Vista [61, 62, 82, 106]</td>
<td>Full system</td>
<td>Interface, Algorithm, System usage</td>
<td>Human users</td>
</tr>
<tr>
<td>RACOT [72, 85]</td>
<td>Prototype</td>
<td>Algorithm</td>
<td>System designers</td>
</tr>
<tr>
<td>QAII [78, 79]</td>
<td>Full system</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>CYCLES [8]</td>
<td>Full system</td>
<td>Algorithm</td>
<td>System designers</td>
</tr>
<tr>
<td>Colind [29, 30]</td>
<td>Prototype</td>
<td>System usage</td>
<td>Human users</td>
</tr>
<tr>
<td>Learning object sequencing [88]</td>
<td>Prototype</td>
<td>System usage</td>
<td>Human users</td>
</tr>
<tr>
<td>Evolving e-learning system [90, 91, 92, 93]</td>
<td>Full system</td>
<td>Algorithm, System usage</td>
<td>Simulated users, Human users</td>
</tr>
<tr>
<td>ISIS - Hybrid Personalised Recommender System [28]</td>
<td>Prototype</td>
<td>Algorithm, System usage</td>
<td>Human users</td>
</tr>
<tr>
<td>Multi-Attribute Recommendation Service [62]</td>
<td>Prototype</td>
<td>Algorithm</td>
<td>System designers</td>
</tr>
<tr>
<td>Learning Object Recommendation Model [95]</td>
<td>Design</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>RecSearch [32]</td>
<td>Design</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Simulation environment [72]</td>
<td>Full system</td>
<td>Algorithm</td>
<td>Simulated users</td>
</tr>
<tr>
<td>ReMashed [56, 27]</td>
<td>Full system</td>
<td>Algorithm, System usage</td>
<td>Human users</td>
</tr>
<tr>
<td>CourseRank [55, 56]</td>
<td>Full system</td>
<td>System usage</td>
<td>Human users</td>
</tr>
<tr>
<td>CBR Recommender Interface [93]</td>
<td>Prototype</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>APOSTLE Recommender Service [31]</td>
<td>Prototype</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>A2M Recommender System [96]</td>
<td>Prototype</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Moodle Recommender System [44]</td>
<td>Prototype</td>
<td>Algorithm, System usage</td>
<td>Human users</td>
</tr>
<tr>
<td>LHS [44]</td>
<td>Prototype</td>
<td>System usage, Learner performance</td>
<td>Human users</td>
</tr>
<tr>
<td>RTL recommender [49]</td>
<td>Prototype</td>
<td>System usage</td>
<td>System designers, Human users</td>
</tr>
</tbody>
</table>
**Table 12.3: Implemented TEL recommender systems reported in literature**

<table>
<thead>
<tr>
<th>System</th>
<th>Status</th>
<th>Evaluator focus</th>
<th>Evaluation roles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Altered Vista</td>
<td>Full system</td>
<td>Interface, Algorithm, System usage</td>
<td>Human users</td>
</tr>
<tr>
<td>[81, 82, 82, 104]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RACOFI</td>
<td>Prototype</td>
<td>Algorithm</td>
<td>System designers</td>
</tr>
<tr>
<td>[2, 61]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>QSAI</td>
<td>Full system</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[78, 79]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CYCLADES</td>
<td>Full system</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[4]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CoFind</td>
<td>Prototype</td>
<td>System usage</td>
<td>Human users</td>
</tr>
<tr>
<td>[29, 30]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Learning object sequencing</td>
<td>Full system</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[88]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISSIS: Hybrid Personalised</td>
<td>Full system</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Recommender System</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[28]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Metric Attribute Recommendation Service</td>
<td>Full system</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[67]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Learning object Recommendation Model</td>
<td>Full system</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[95]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RecSearch</td>
<td>Full system</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[33]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simulation environment</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[72]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BeReMash</td>
<td>Full system</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[26, 27]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CourseRank</td>
<td>Prototype</td>
<td>System usage</td>
<td>Human users</td>
</tr>
<tr>
<td>[55, 56]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CBR Recommender Interface</td>
<td>Prototype</td>
<td>System usage</td>
<td>Human users</td>
</tr>
<tr>
<td>[73]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>APENDLE Recommendation Service</td>
<td>Prototype</td>
<td>System usage</td>
<td>Human users</td>
</tr>
<tr>
<td>[88]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Evolving e-learning system</td>
<td>Full system</td>
<td></td>
<td>Simulated users</td>
</tr>
<tr>
<td>[88]</td>
<td></td>
<td></td>
<td>Human users</td>
</tr>
</tbody>
</table>

---

Conclusions:

Half of the systems (11/20) still at design or prototyping stage only 8 systems evaluated through trials with human users.
Thus...

“The performance results of different research efforts in recommender systems are hardly comparable.”

(Manouselis et al., 2010)
Thus...

TEL recommender experiments lack transparency. They need to be repeatable to test:

• Validity
• Verification
• Compare results
### TEL RecSys:: Evaluation/datasets

<table>
<thead>
<tr>
<th></th>
<th>Mendeley</th>
<th>APOSDLE</th>
<th>ReMashed</th>
<th>Organic edunet</th>
<th>Mace</th>
<th>Melt</th>
</tr>
</thead>
<tbody>
<tr>
<td>Collection period</td>
<td>1 year</td>
<td>3 months</td>
<td>2 years</td>
<td>9 months</td>
<td>3 years</td>
<td>6 months</td>
</tr>
<tr>
<td>Users</td>
<td>200.000</td>
<td>6</td>
<td>140</td>
<td>1.000</td>
<td>1.148</td>
<td>98</td>
</tr>
<tr>
<td>Items</td>
<td>1.857.912</td>
<td>163</td>
<td>96.000</td>
<td>11.000</td>
<td>12.000</td>
<td>1.923</td>
</tr>
<tr>
<td>Activities</td>
<td>4.848.725</td>
<td>1.500</td>
<td>23.264</td>
<td>920</td>
<td>461.982</td>
<td>16.353</td>
</tr>
<tr>
<td>reads</td>
<td>+</td>
<td>+</td>
<td>-</td>
<td>-</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td>tags</td>
<td>+</td>
<td>(+)</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>ratings</td>
<td>(+)</td>
<td>-</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>downloads</td>
<td>+</td>
<td>+</td>
<td>-</td>
<td>-</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>search</td>
<td>-</td>
<td>+</td>
<td>-</td>
<td>-</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td>collaborations</td>
<td>-</td>
<td>+</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>tasks/goals</td>
<td>-</td>
<td>+</td>
<td>+</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>sequence</td>
<td>-</td>
<td>+</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>competence</td>
<td>-</td>
<td>+</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>+</td>
</tr>
<tr>
<td></td>
<td>Mendeley</td>
<td>APOSDLE</td>
<td>ReMashed</td>
<td>Organic .edunet</td>
<td>Mace</td>
<td>Melt</td>
</tr>
<tr>
<td>------------------</td>
<td>----------</td>
<td>---------</td>
<td>----------</td>
<td>-----------------</td>
<td>------</td>
<td>------</td>
</tr>
<tr>
<td>Collection period</td>
<td>1 year</td>
<td>3 months</td>
<td>2 years</td>
<td>9 months</td>
<td>3 years</td>
<td>6 months</td>
</tr>
<tr>
<td>Users</td>
<td>200,000</td>
<td>6</td>
<td>140</td>
<td>1,000</td>
<td>1.148</td>
<td>98</td>
</tr>
<tr>
<td>Items</td>
<td>1,857,912</td>
<td>163</td>
<td>96,000</td>
<td>11,000</td>
<td>12,000</td>
<td>1.923</td>
</tr>
<tr>
<td>Activities</td>
<td>4,848,725</td>
<td>1,500</td>
<td>23,264</td>
<td>920</td>
<td>461,982</td>
<td>16,353</td>
</tr>
<tr>
<td>reads</td>
<td>+</td>
<td>+</td>
<td>-</td>
<td>-</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td>tags</td>
<td>+</td>
<td>(+)</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>ratings</td>
<td>(+)</td>
<td>-</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>downloads</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
</tbody>
</table>

Evaluation:

**Metrics**

**MAE** – Mean Absolute Error:
Deviation of recommendations from the user-specified ratings. The lower the MAE, the more accurately the RecSys predicts user ratings.

Evaluation::Metrics

**Outcomes:**
Tanimoto similarity + item-based CF was the most accurate.

**MAE – Mean Absolute Error:**
Deviation of recommendations from the user-specified ratings. The lower the MAE, the more accurately the RecSys predicts user ratings.

Outcomes:
• User-based CF Algorithm that predicts the top 10 most relevant items for a user has a F1 score of almost 30%.

• Implicit ratings like download rates, bookmarks can successfully used in TEL.

TEL RecSys::Evaluation

Combined approach by
Drachsl er et al. 2008

Kirkpatrick model by
Manouselis et al. 2010
TEL RecSys::Evaluation

1. Accuracy
2. Coverage
3. Precision

Combined approach by 
*Drachsler et al. 2008*

Kirkpatrick model by 
*Manouselis et al. 2010*
TEL RecSys::Evaluation

1. Accuracy
2. Coverage
3. Precision

1. Effectiveness of learning
2. Efficiency of learning
3. Drop out rate
4. Satisfaction

Combined approach by Drachsl er et al. 2008
Kirkpatrick model by Manouselis et al. 2010
TEL RecSys::Evaluation

1. Accuracy
2. Coverage
3. Precision

1. Effectiveness of learning
2. Efficiency of learning
3. Drop out rate
4. Satisfaction

1. Reaction of learner
2. Learning improved
3. Behaviour
4. Results

Combined approach by Drachsl er et al. 2008

Kirkpatrick model by Manouselis et al. 2010
Goals of the lecture

1. Crash course Recommender Systems (RecSys)

2. Overview of RecSys in TEL

3. Open research issues for RecSys in TEL

4. TEL RecSys and Learning Analytics
TEL RecSys::Open issues

1. Evaluation of TEL RecSys
2. Publicly available datasets
3. Comparable experiments
4. Body of knowledge
5. Privacy and data protection
6. Design learning driven RecSys
Goals of the lecture

1. Crash course Recommender Systems (RecSys)

2. Overview of RecSys in TEL

3. Open research issues for RecSys in TEL

4. TEL RecSys and Learning Analytics
Learning Analytics

Greller, W., & Drachsler, H., 2011.
Learning Analytics

Greller, W., & Drachsler, H., 2011.
Greller, W., & Drachsler, H., 2011.
Greller, W., & Drachsler, H., 2011.
Greller, W., & Drachsler, H., 2011.
Learning Analytics

Greller, W., & Drachsler, H., 2011.
Learning Analytics

Greller, W., & Drachsler, H., 2011.
Learning Analytics:: TimeToThink

• Consider the Learning Analytics framework and imagine some great TEL RecSys that could support you in your stakeholder role

alternatively

• Name a learning task where a TEL RecSys would be useful for.
Thank you for attending this lecture!

This slide is available at:
http://www.slideshare.com/Drachsler

Email: hendrik.drachsler@ou.nl
Skype: celstec-hendrik.drachsler
Blogging at: http://www.drachsler.de
Twittering at: http://twitter.com/HDrachsler