Usability of intelligent systems
Case example: search engines

Antti Kangasrääsiö
Curious AI

AI Helsinki 29.10.2018
Usability of The Hammer

3,000,000 years ago

30,000 years ago

now
Usability

A system is usable for a task if it is easy for the user to complete the task with the help of the system.

- We need to consider who is the user
- What is the system
- What is the task

Many tools are designed to be easy to hold and use for their intended purpose. For example, a screwdriver typically has a handle with rounded edges and a grippable surface, to make it easier for the user to hold the handle and twist it to drive a screw.
Mental models

Mental model is the user’s internal representation of the used system

- Allows the user to mentally simulate behavior of system
- Thus, allows selection of suitable actions
- (= model-predictive control)

=> In order to be usable, the system’s behavior and the user’s mental model must be aligned
Usability of Statistical Models

300 years ago

30 years ago

future?
Usability of a Recommender System

Task: Find interesting items from a vast catalogue

How:

- Construct a model of the user’s interests based on user input
- Rank catalogue using model, display top results to user
- If user is not satisfied, allow her to give further input
- Repeat until convergence
Usability of a Recommender System

“If user is not satisfied, allow her to give further input”

“A system is usable for a task if it is easy for the user to complete the task with the help of the system”

● What kind of input can the user give?
● How is the input given?
● How can the user know what input will improve the results?
Usability of a Recommender System

What kind of input can the user give?
- What are the input variables of the recommender model?

How is the input given?
- How is the UI implemented?

How can the user know what input will improve the results?
- How to make the system behavior match the mental model of the user

Search keywords
Relevance feedback
...

Keyword box
Feedback on model/results
...

???
How to make the system match the mental model?

Two options:

Change the mental model
   Make it easier for the user to calibrate her mental model

Change the system behavior
   Make the system behave ‘more intuitively’
Allowing user to calibrate her mental model

One solution:

Visualize the predicted effects of user actions

- Improves predictability of user actions
- Facilitates calibration of mental model

Low relevance feedback?  High relevance feedback?

Antti Kangasrääsiö, Dorota Glowacka, and Samuel Kaski. Improving Controllability and Predictability of Interactive Recommendation Interfaces for Exploratory Search, Intelligent User Interfaces 2015
Make the system behave ‘more intuitively’

One solution: Interpret user feedback as requirements for next user model state

- Not just: “add this observation to training set”
- But instead: “find out what observations should be added to the training set for the system to behave like I want”

Not like this: user wants to make a certain feature relevant, but resulting user model state does not reflect this

Antti Kangasrääsiö, Dorota Głowacka, and Samuel Kaski. Improving Controllability and Predictability of Interactive Recommendation Interfaces for Exploratory Search, Intelligent User Interfaces 2015
User errors

Users make various kinds of errors

- Unintentional errors
  - “Oops!”
- Mistaken actions
  - “I thought that would do something else”

In order to quickly recover from errors, users need feedback

- Example: automatic spell-checking
  - “Oh, i made a typo?”
  - “Oh, the word is not spelled like that?”
Error detection in one-shot search

Spell-checking like behavior can be used to detect and highlight typos

However, there is no feedback on queries that do not make sense
Error detection in iterative query refinement

When user refines the search query by giving relevance feedback, the final model is a function of the *entire training data set*.

User’s search interest may change over time and there might be errors in the feedback user gives.

**Solution:** Visualize the user’s feedback history and highlight outliers.
