

Requirements Engineering I

Chapter 11

RE Support and Guidance



11.1 Requirements engineering tools

[Carrillo de Gea et al. 2011]

What can be supported by a RE tool?

- **Elicitation** (e.g., analysis of textual artifacts)
- **Documentation** (generating and editing requirements work products)
- **Modeling** (primarily model editors)
- **Management** (Store and retrieve, prioritize, trace,...)
- **Validation** (finding quality problems, simulators, model checkers,...)

Support levels for RE tools

- **General purpose**
 - Word processors
 - Spreadsheet tools
 - General purpose graphic drawing tools
- **Database-level**
 - Requirements management tools for organizing, storing, retrieving and tracing requirements
- **Language & method-based**
 - Tools supporting specific requirements languages, e.g., drawing state machine diagrams
 - Tools for supporting specific methods, e.g., validation with model-checking, or checking a document for compliance

Which RE tool should I use / buy?

[Bruckhaus, Madhavji, Janssen, Henshaw 1996]

- No general recommendation possible
- Depends on what the tool(s) shall support
- An RE tool does not automatically improve productivity
- An up-to-date list of requirements tools is maintained at the VOLERE website:

<https://www.volere.org/requirements-tools/>

11.2 RE Standards

IEEE 830-1984 IEEE Guide to Software Requirements Specifications

- The first RE standard – very good by its time
- Revised 1993 and 1998
- IEEE 830-1998 is officially retired, but still in use, in particular for documenting requirements

ISO/IEC/IEEE 29148, originally from 2011, revised 2018

- A very heavyweight, document- and process-centric standard
- Does not work well for participative and lightweight RE processes

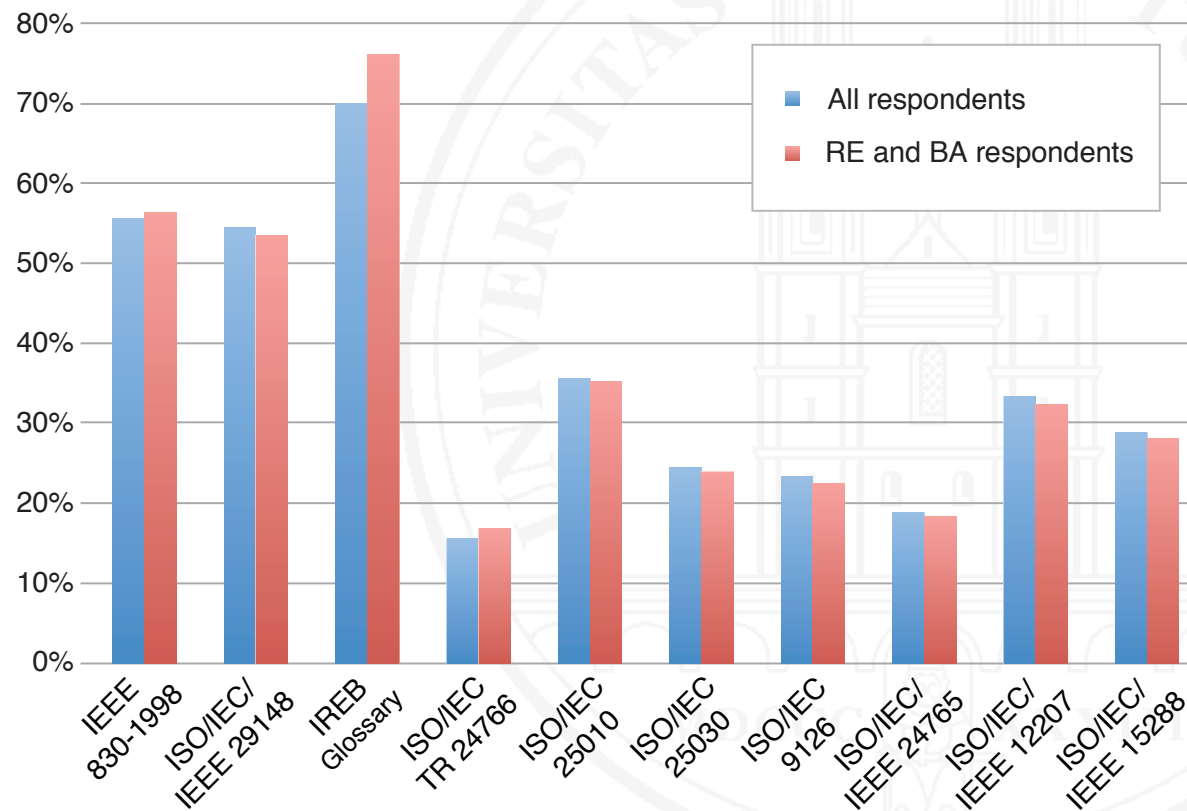
[IEEE 1998]

[ISO/IEC/IEEE 2018]

Knowledge and use of RE-related standards

[Franch, Glinz, Méndez and Seyff 2022]

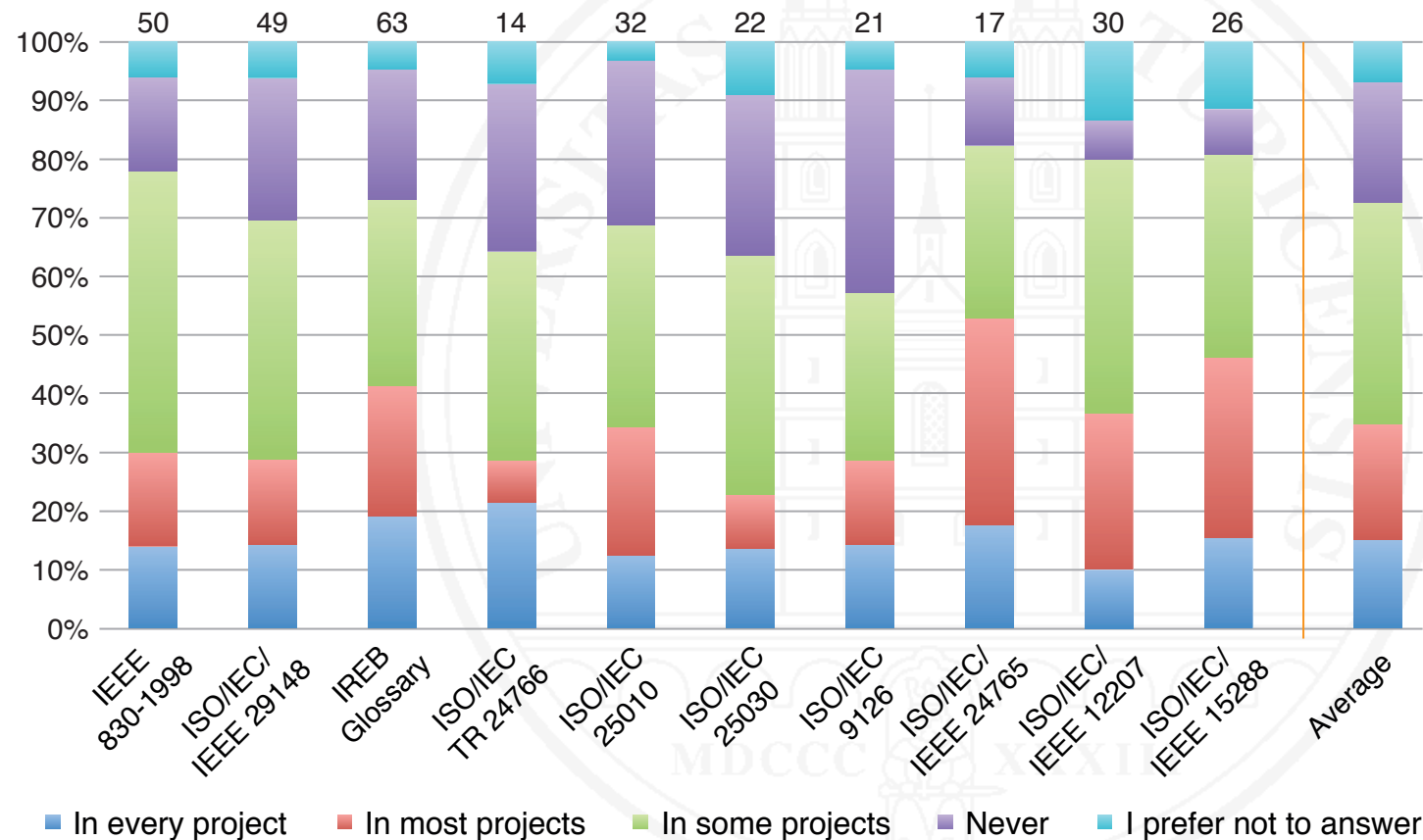
Results from an empirical study:



→ The **knowledge** of RE-related standards is **rather low**

Knowledge and use of RE-related standards – 2

Frequency of usage



→ The known standards are **barely used**

Related standards

Quality standards, particularly in conjunction with quality requirements

- ISO/IEC 25010 System and Software Quality Requirements and Evaluation: Quality Models
- ISO/IEC 25030 Software Product Quality Requirements and Evaluation: Quality Requirements
- ISO/IEC 9126 Software Engineering – Product Quality: Quality Model (superseded, predecessor of ISO/IEC 25010)

System and software engineering standards, e.g.,

- ISO/IEC/IEEE 12207 on software life cycle processes
- ISO/IEC/IEEE 15288 on system life cycle processes
- ISO/IEC/IEEE 24765 on systems & software engineering vocabulary

Domain-specific standards

Domain-specific standards may **impact** Requirements Engineering

Example:

ISO 26262 Road Vehicles — Functional Safety

If a customer or regulator demands **compliance** of a system with ISO 26262, then **traceability** between requirements and test cases is **mandatory**.

11.3 RE syllabi and body of knowledge

There is no Requirements Engineering **Body of Knowledge** (RE BoK) document

The IREB Certified Professional for Requirements Engineering (CPRE) foundation level syllabus and the associated handbook and glossary constitute a **de facto basic RE BoK**



[IREB 2022]
[Glinz, van Loenhoud,
Staal and Bühne 2022]
[Glinz 2020]

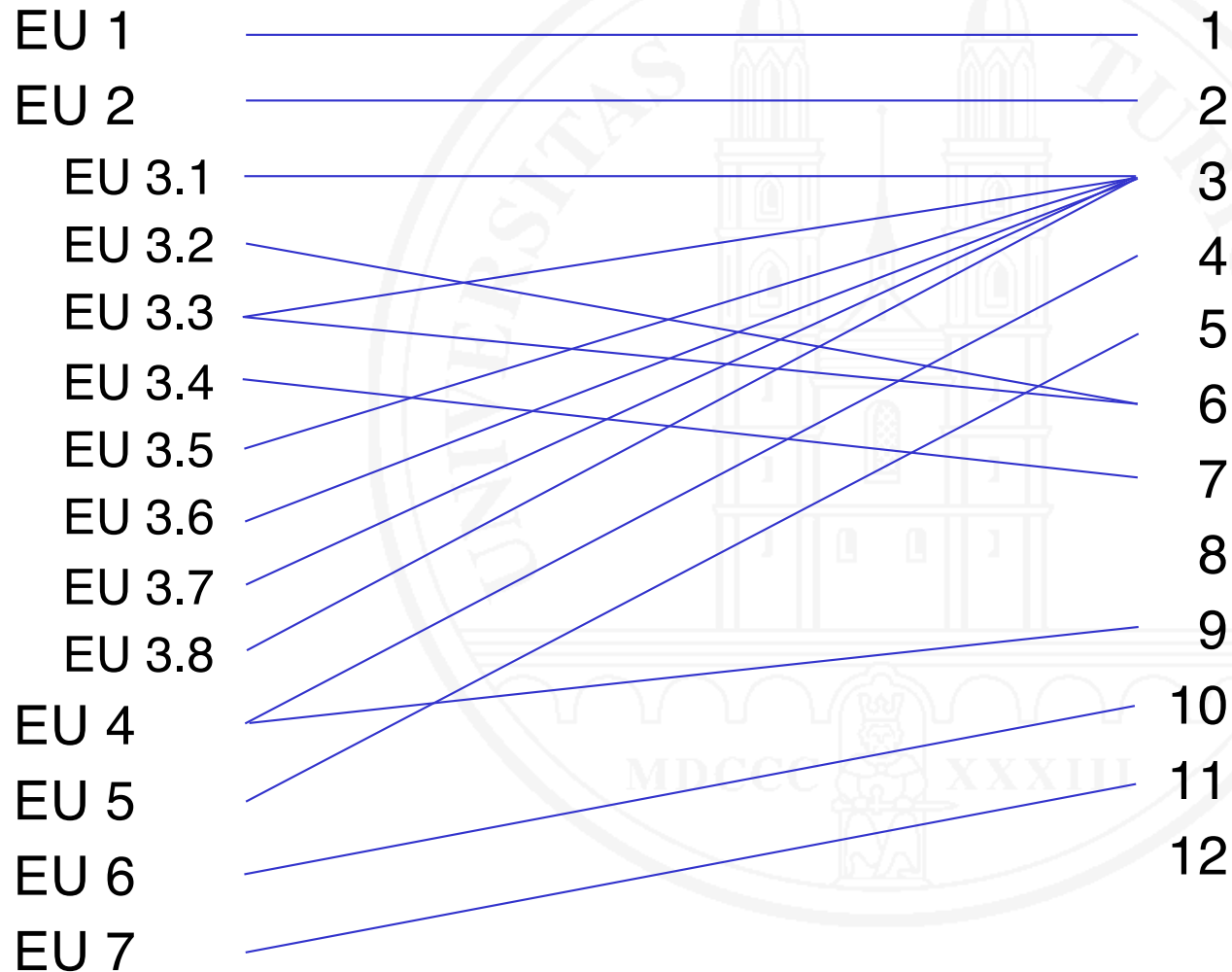
IREB CPRE vs. this course

- This course covers all topics of the IREB CPRE foundation level syllabus
- The terminology is the same
- Some topics of this course go beyond the CPRE FL, for example:
 - COTS-aware processes (Chapter 5)
 - Formal specification (Chapter 8)
 - Standards, AI for RE, RE Ethics (Chapter 11)

Synopsis of topics

CPRE FL Syllabus

Chapter in this course



11.4 AI for RE

[Dalpiaz and Niu 2020]

What can AI do for RE?

Primary means: processing natural language text with **machine learning**

- **Find** and **classify**, for example
 - Identify potential requirements in user feedback (app reviews, tweets)
 - Classify sentences in a document into requirements and informational statements
 - Extract glossary candidates from textual requirements
 - Find smells in requirements
 - Find trace links between RE documents

What can AI do for RE – 2

- **Recommend**, for example
 - Recommend further stakeholders / stakeholder roles during stakeholder analysis
 - Provide advice for configuring requirements in a product line
- **Analyze**, for example
 - Automated impact analysis when requirements change
- **Support**, for example
 - Support human interaction between stakeholders and requirements engineers
 - Chatbots for autonomous interaction with a big number of stakeholders

Example: The ALERT.me approach

[Guzmán, Ibrahim, Glinz 2017]

Context: Large product or service providers continuously receive thousands of tweets about their product.

Problem: Some of these tweets contain user needs that are a source of requirements for evolving the product or service. Manually finding these tweets is tedious and expensive.

Illustration: Two tweets to Slack:

@SlackHQ At my company we share code snippets around a lot. There should be a quick way to copy a raw code snippet to your clipboard.

User need

I always wanted t-shirts, but I didn't know socks were an option. I've got the start with my @SlackHQ faves - gotta catch 'em all!

Other stuff

Example: The ALERT.me approach – 2

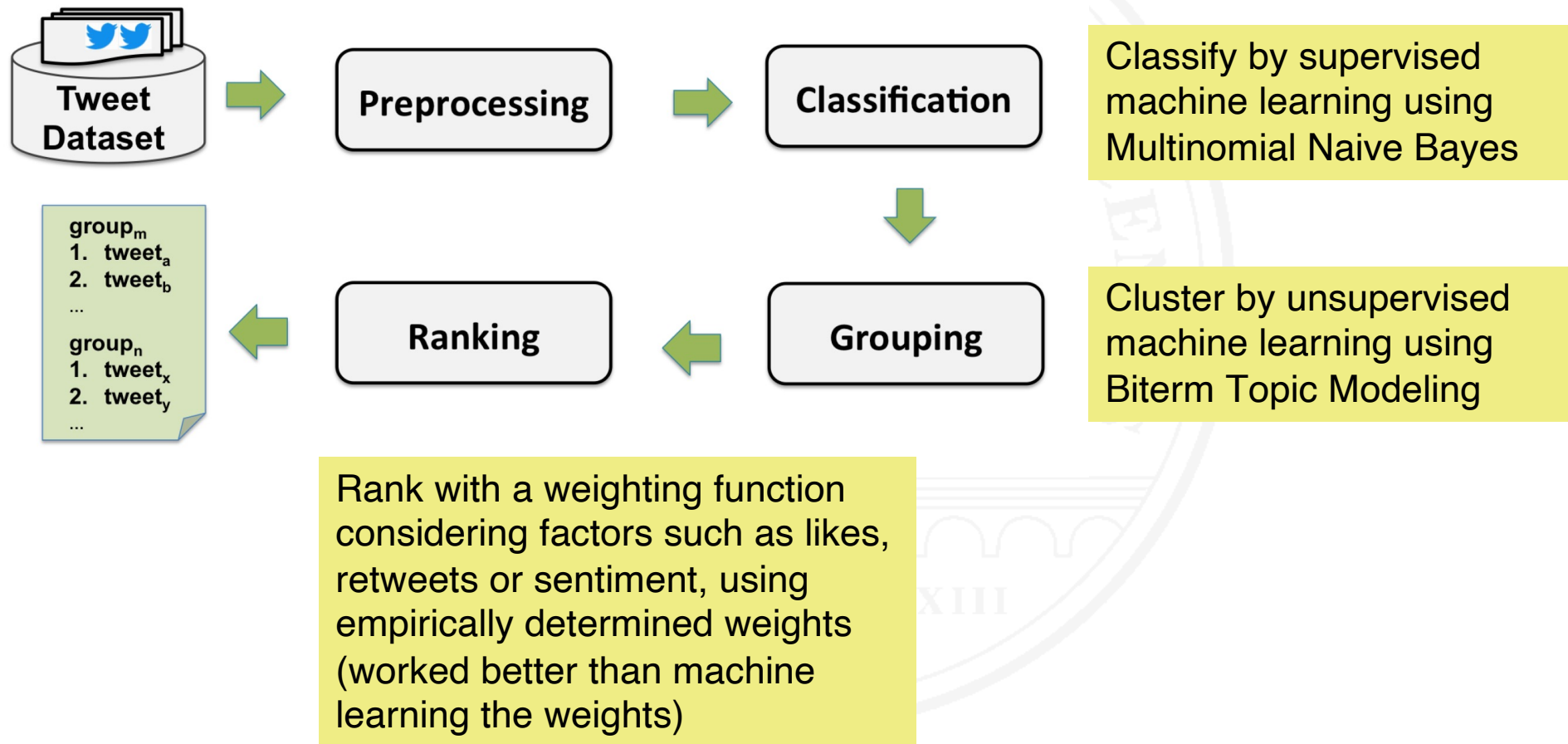
Solution: Create a tool that extracts user needs and presents them in a convenient form to the requirements engineers

Three steps:

- 1 **Classify** tweets into **improvement requests** and **other**
- 2 **Cluster** improvement requests by **grouping** them into **topics**
- 3 **Rank** the grouped requests by their **relevance**

Example: The ALERT.me approach – 3

Architecture of ALERT.me



The recall problem of AI-based RE tools

[Berry 2021]

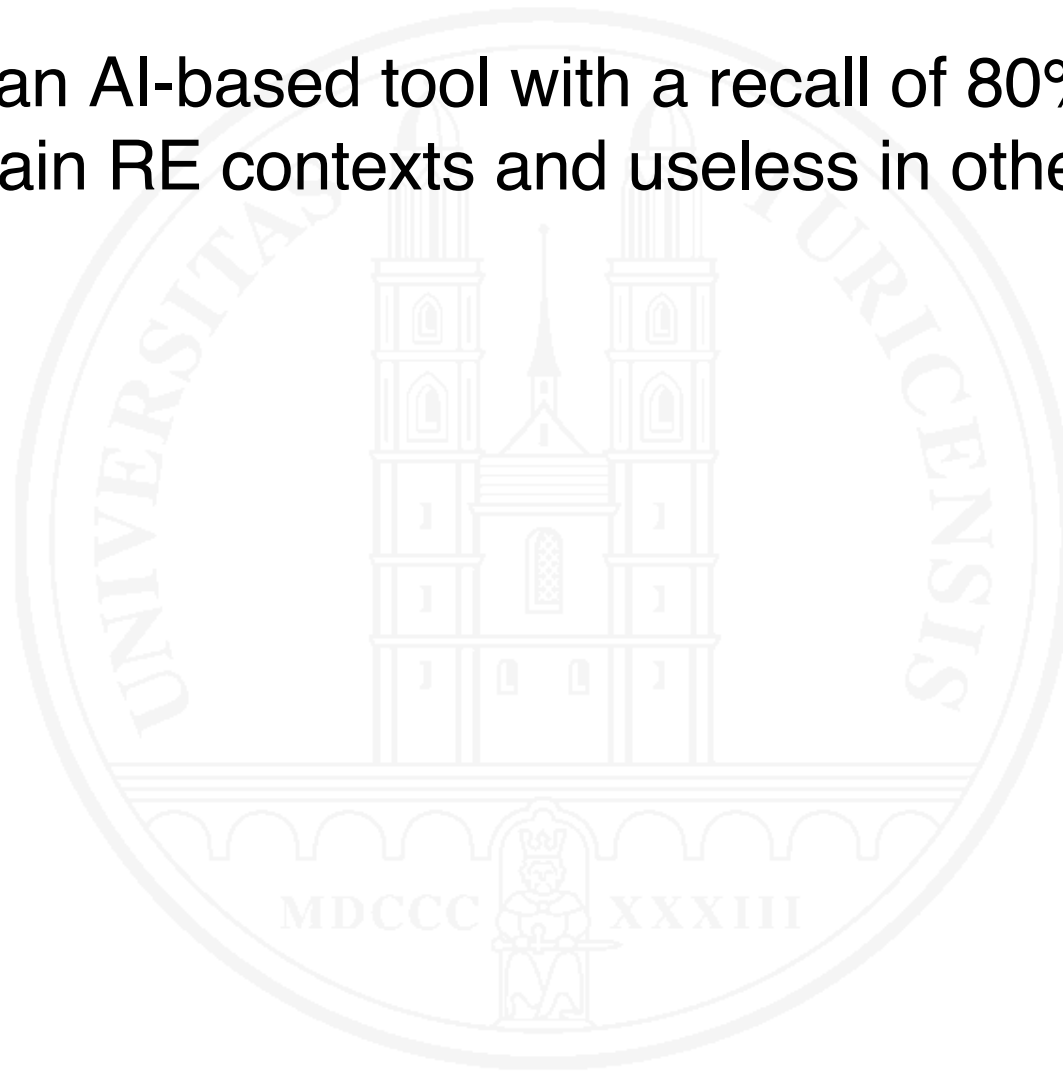
- Automated classifiers make **mistakes**:
 - Not including relevant items in the result set (**false negatives**; **recall** < 100%)
 - Including irrelevant items (**false positives**; **precision** < 100%)
- A tool such as ALERT.me is still useful when recall is only about 80%.
- In other contexts, a tool with 80% recall can be useless.

Imagine a case where full requirements to test traceability is mandatory for complying with regulations.

When a human would find ~98 percent of the links manually, a traceability tool with 80 percent recall is useless.

Mini-Exercise

Explain why an AI-based tool with a recall of 80% can be very useful in certain RE contexts and useless in other RE contexts.



11.5 Requirements Engineering Ethics

Three relevant ethical dimensions

- Ethics of **profession**
- Ethics of **use**
- Ethics of **design**



in RE

[Simon 2022]

[Barker&Ferguson 2022]

[Norman 2013]

Ethic principles

○ No harm



○ For the good



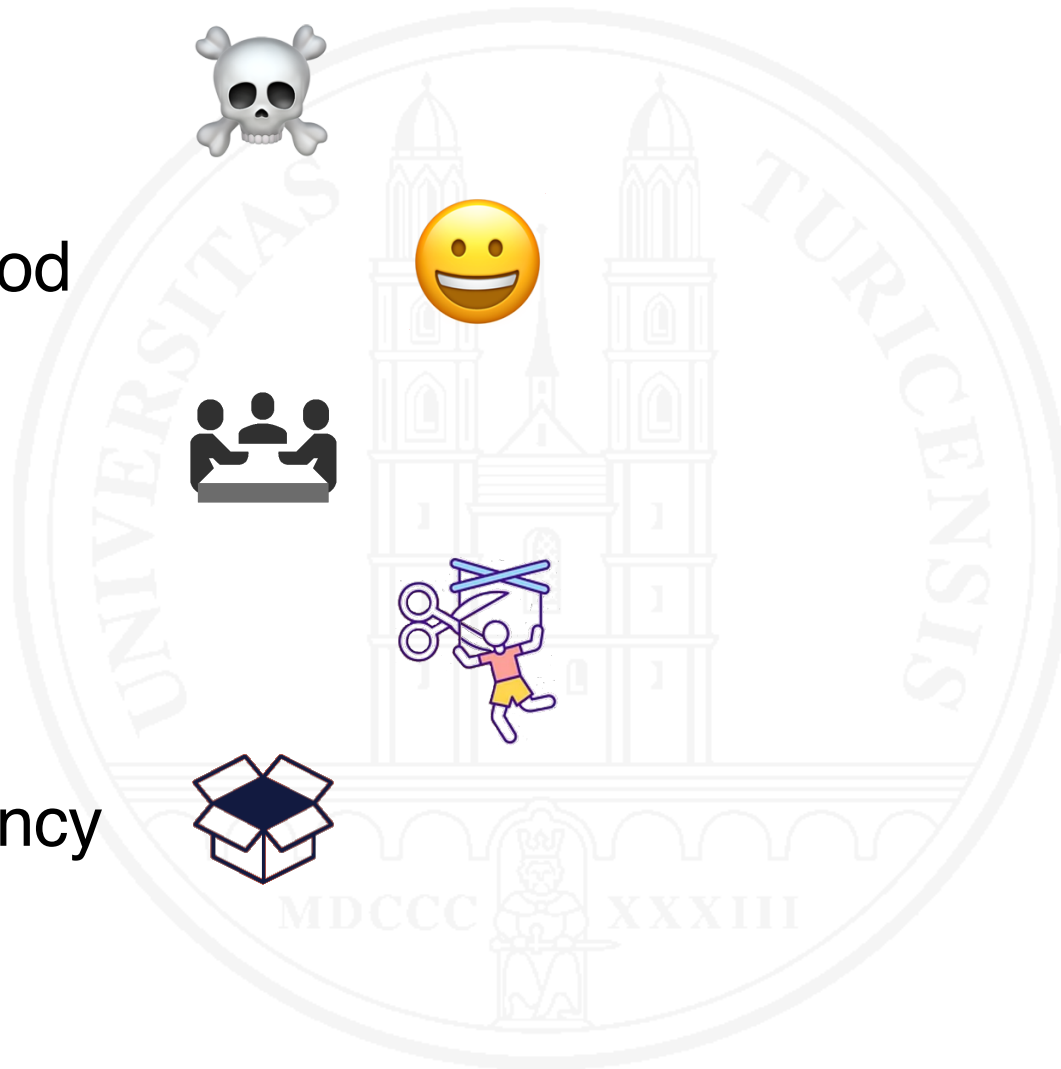
○ Fairness



○ Autonomy



○ Transparency



Ethics of profession in RE

- Consider **how to act ethically** as a requirements engineer
- Comply with the **code of ethics**
 - of your organization
 - of professional societies where you are a member

Some advice

- Refuse working on maleficent requirements (**no harm**)
- Assess benefits and risks of systems built according to the requirements (**for the good**)
- Treat equal stakeholders equal (**fairness**)
- Guide stakeholders, but do not force them (**autonomy**)
- Be able to explain what you are doing and why (**transparency**)

Ethics of use in RE

- Consider the **impact of your requirements** on the **users** of the system to be built
 - Can the system in use **do harm** (to people, the environment, the society,...)?
 - Does the system **help** its **users doing things better** than before?
 - Does or can the system **discriminate** certain users or or **favor** them over others without a valid reason?
 - Does the system help **empower** its users?
 - Does the system help users **understand** what the system does when they use it?

Ethics of design in RE

- Consider the **impact of your requirements** on the **design** of the system to be built
 - Are there requirements that **prevent** the system from doing **harm** (safety, security, reliability,...)
 - Do the requirements enable designing a system that provides **benefit** for, for example, its users, the environment or the society – and do this **with controllable risks**?
 - Do the requirements enable designing a system that is **user-friendly** and **empowers** its users?
 - Are there requirements asking for **explainability** of what the system does when in operation?