

Multimodal Machine Learning Lab

Winter Semester 2025/2026

Niklas Deckers and Martin Potthast

Agenda

- ❑ Refactoring Experiment Code
- ❑ Abstracting Tasks

Introducing Flexibility

- ❑ For the Sender and Retriever components, both human and machine agents should be possible to drop in (i.e., implement the interfaces)
- ❑ The specific terms and symbols that are used in the experiments should be independent of the mechanic of the task and should be interchangeable
- ❑ Separating and harmonizing the task mechanic should allow to easily develop agents (e.g., by providing prompt templates for different types of tasks)

UI Output

- ❑ Flexible format that allows to display multimodal contents
- ❑ Showing content of the form $(\text{image}|\text{text})^+$ for every UI position (static and interactive elements)
- ❑ May be implemented via formats such as Markdown or HTML
- ❑ Adaption as agent model input may require some restrictions

Abstract Types for User Input

- ❑ The input types are closely linked to the task mechanic
- ❑ Over the alphabet Σ , the following input types would be feasible:
 - Single choice (from Σ)
 - Multiple choice (2^Σ)
 - Words (Σ^+)
 - n-length words (Σ^n)
 - Permutations (S_Σ)
 - Variation (i.e., words) without repetitions of a certain length (e.g., how Duolingo helps to input translations)
 - Drawing free-form images (but this might be difficult as an LLM-generated input)
 - Matching some given items to some given entries (can be modeled as variation or permutation)
- ❑ When using LLM agents, each of these input types requires a specific validation method
- ❑ Giving multiple instances of the input types described above should also be possible

Making Tasks Difficult

- ❑ We identified 3 components that make the tasks difficult:
 1. The specific terms and symbols used (i.e., the underlying dataset)
 2. The modality used (e.g., enforcing to use images at some point to describe a word)
 3. The task mechanic (e.g., free text vs. multiple choice)