

Problem Statement

- Set of documents
 - Content
 - Metadata
 - Entities
 - Time
 - ...
- Query
- Answer:
 - Score for each metadata + Score for content
 - Combine Scores or Combined the ranked List

Pragmatics

- Need for knowing the task/ Purpose of a search or what the user has in mind.
- Vocabulary of Tasks

Task Vocabulary

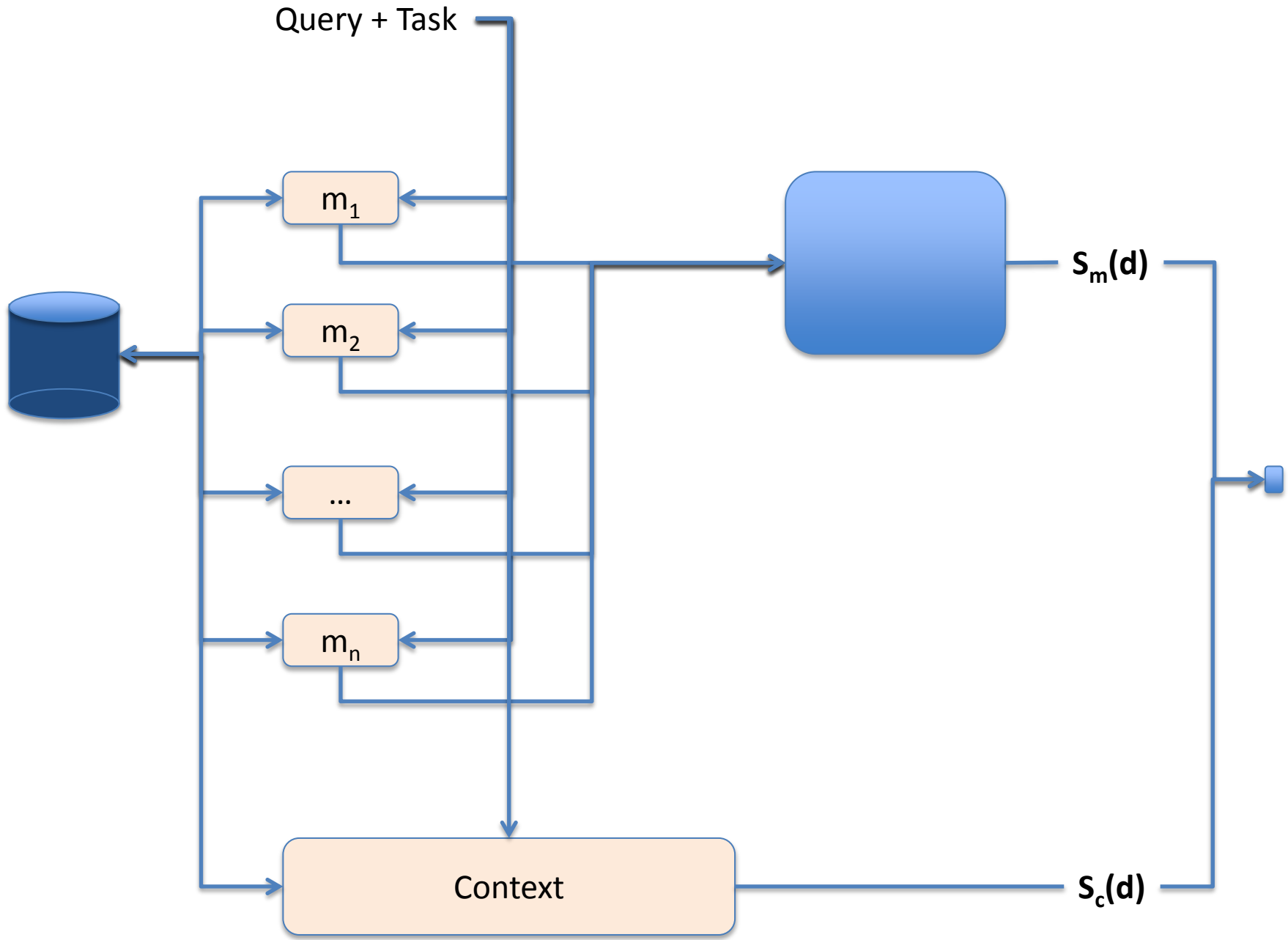
- Tree of tasks
- Defined:
 - Explicitly
 - Inferred
- Used:
 - User Query
 - Explicitly Stated (set of nodes, each defining a sub-tree)
 - Inferred from logs or other activities
 - Document
 - Explicitly stated by the author
 - Learned through ML techniques based on the content
 - Explicit rules

Problem statement

- Documents Annotated with metadata + Tasks
- User Query + Set of User task nodes
- Task Vocabulary
 - Each node is associated with a vector of weights
 - One weight for each meta-data layer
- $S_m(d) = w_1 * s_1 + w_2 * s_2 + \dots + w_3 * s_3$
 - $W_i = \text{avg weights for } s_i \text{ in the user task nodes}$
- $\text{Score}(d) = w' * S_m(d) + w'' * S_c(d)$

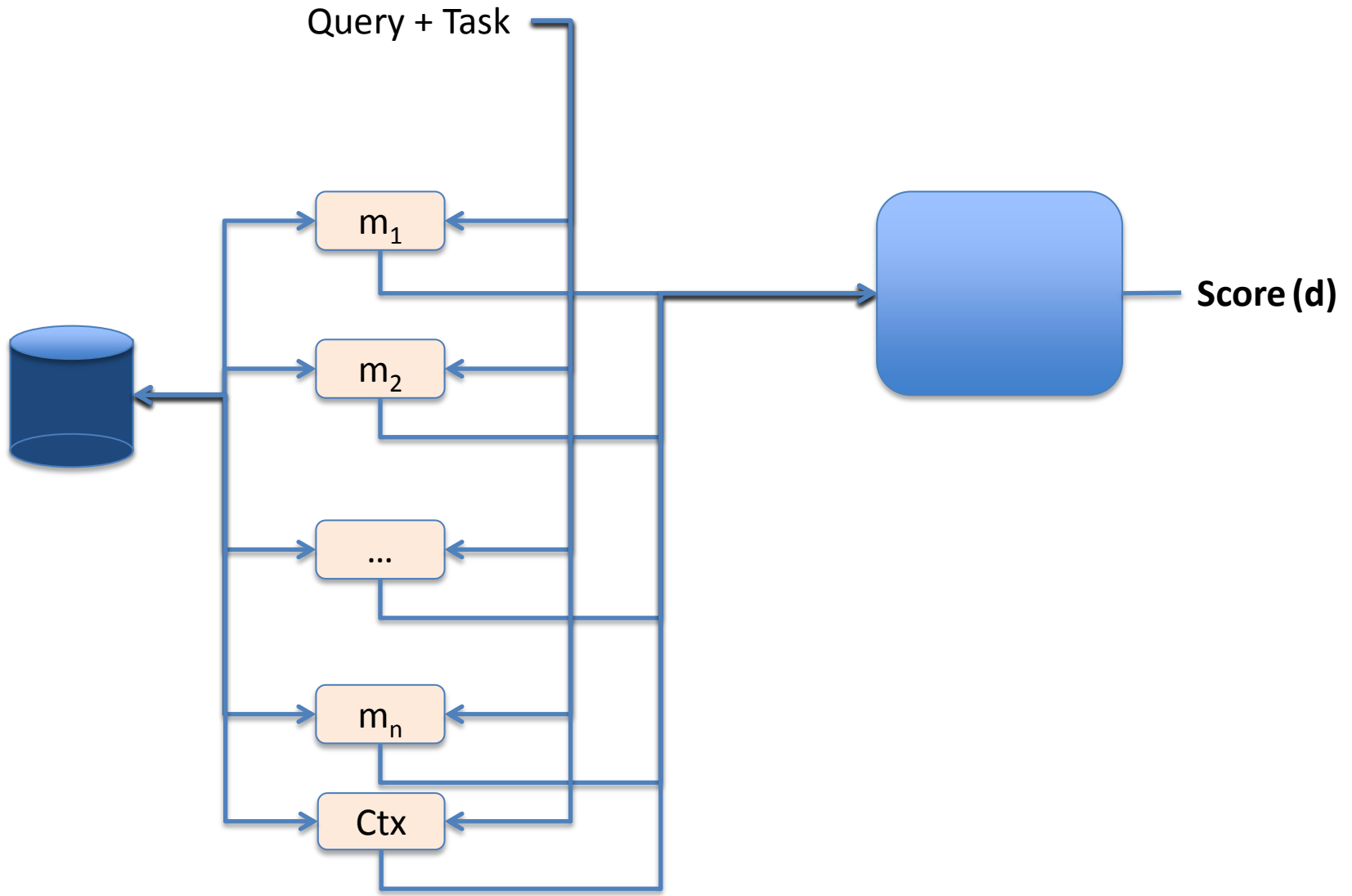
Problem statement

- Documents Annotated with metadata + Tasks
- User Query + Set of User task nodes
- Task Vocabulary
 - Each node is associated with a vector of weights
 - One weight for each meta-data layer
- $S_m(d) = w_1 * s_1 + w_2 * s_2 + \dots + w_3 * s_3$
 - $W_i = \text{avg weights for } s_i \text{ in the user task nodes}$
- $\text{Score}(d) = w' * S_m(d) + w'' * S_c(d)$



Problem statement

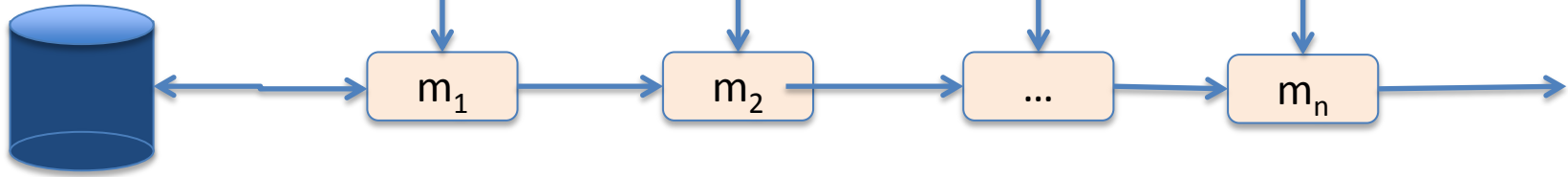
- Documents Annotated with metadata + Tasks
- User Query + Set of User task nodes
- Task Vocabulary
 - Each node is associated with a vector of weights
 - One weight for each meta-data layer
- **Score(d) = $w_1 * s_1 + w_2 * s_2 + \dots + w_3 * s_3 + w'' * S_c(d)$**
 - W_i = avg weights for s_i in the user task nodes
- ~~Score(d) = $w' * S_m(d) + w'' * S_e(d)$~~



Problem statement

- Documents Annotated with metadata + Tasks
- User Query + Set of User task nodes
- Task Vocabulary
 - Each node is associated with a vector of weights
 - One weight for each meta-data layer
- $S_m(d) = S_2 ((S_1 (w_1 * s_1)) * w_2 * s_2) \dots)$
 - $W_i = \text{avg weights for } s_i \text{ in the user task nodes}$
 - Pipeline

Query + Task



Multilingualism

- Common concept vocabulary
- Map keywords to concepts and then use the concept matching

Presentation

- Answer=(ranked) set of documents
- Many contain complementary or overlapping information
- Idea: Information packages
 - Clusters of documents
 - Minimize overlapping information
 - Maximize coverage

Thank you