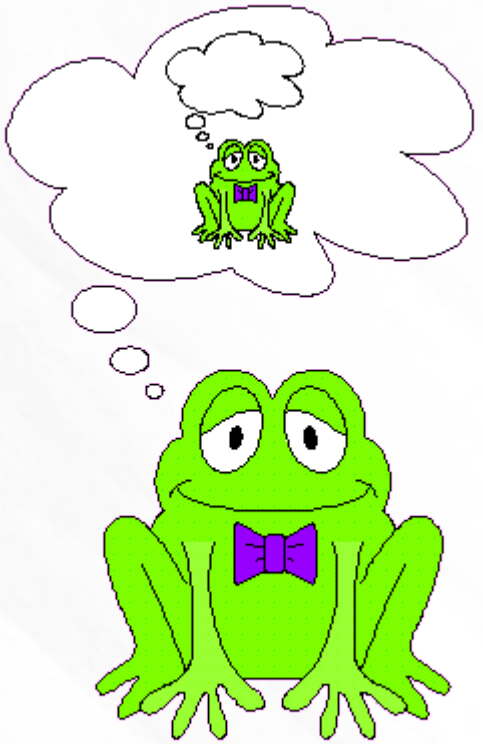
Meta-cognition evaluation

Maruseac Mihai
Neață Sofia
https://github.com/mihaimaruseac/nlp

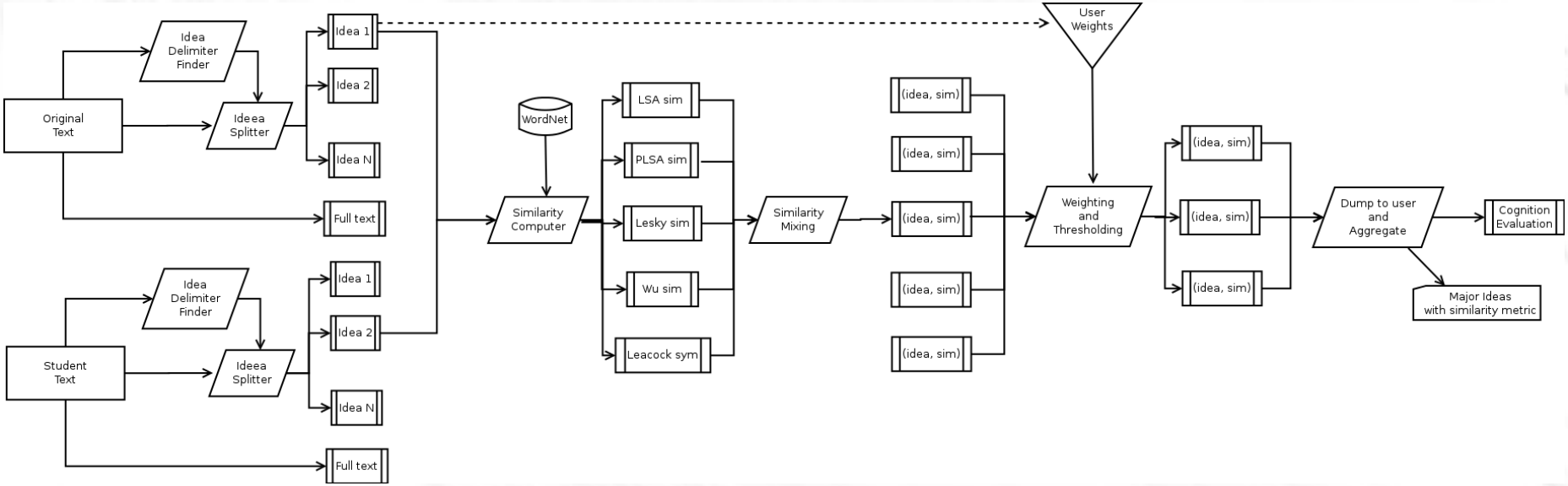
Metacognition

- Thinking about knowing
- Learning about thinking
- Control of learning



Overview

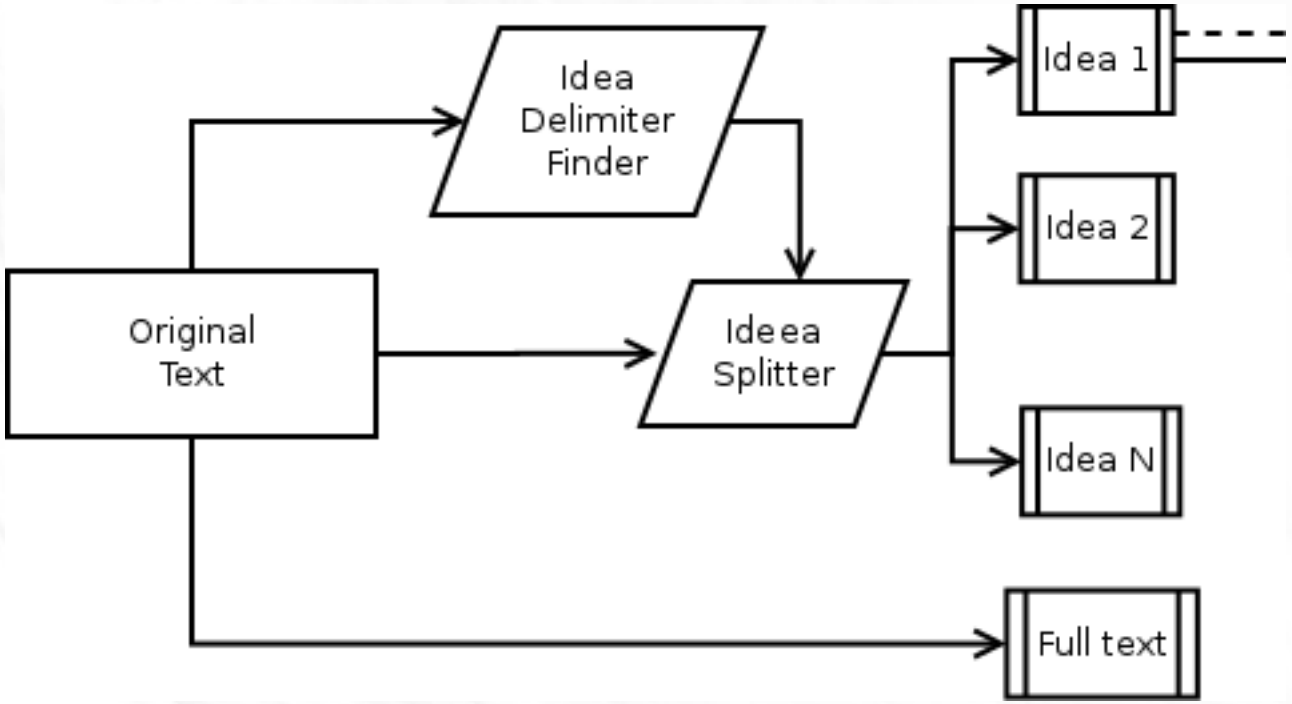
$$MC\ O\ S = presentUser\ \$\ filter\left(TF\ w_{ideeas}\right)\$\ map\ SC\ \$\ MI\left(getIdeas\ O\right)\left(getIdeas\ S\right)$$



Overview

- Pipeline architecture
- Easily parallelisable
- Map-Reduce

Stage 1


Idea
Delimiter
Finder
Original
Text
Ideea
Splitter
Idea 1
Idea 2
Idea N
Full text

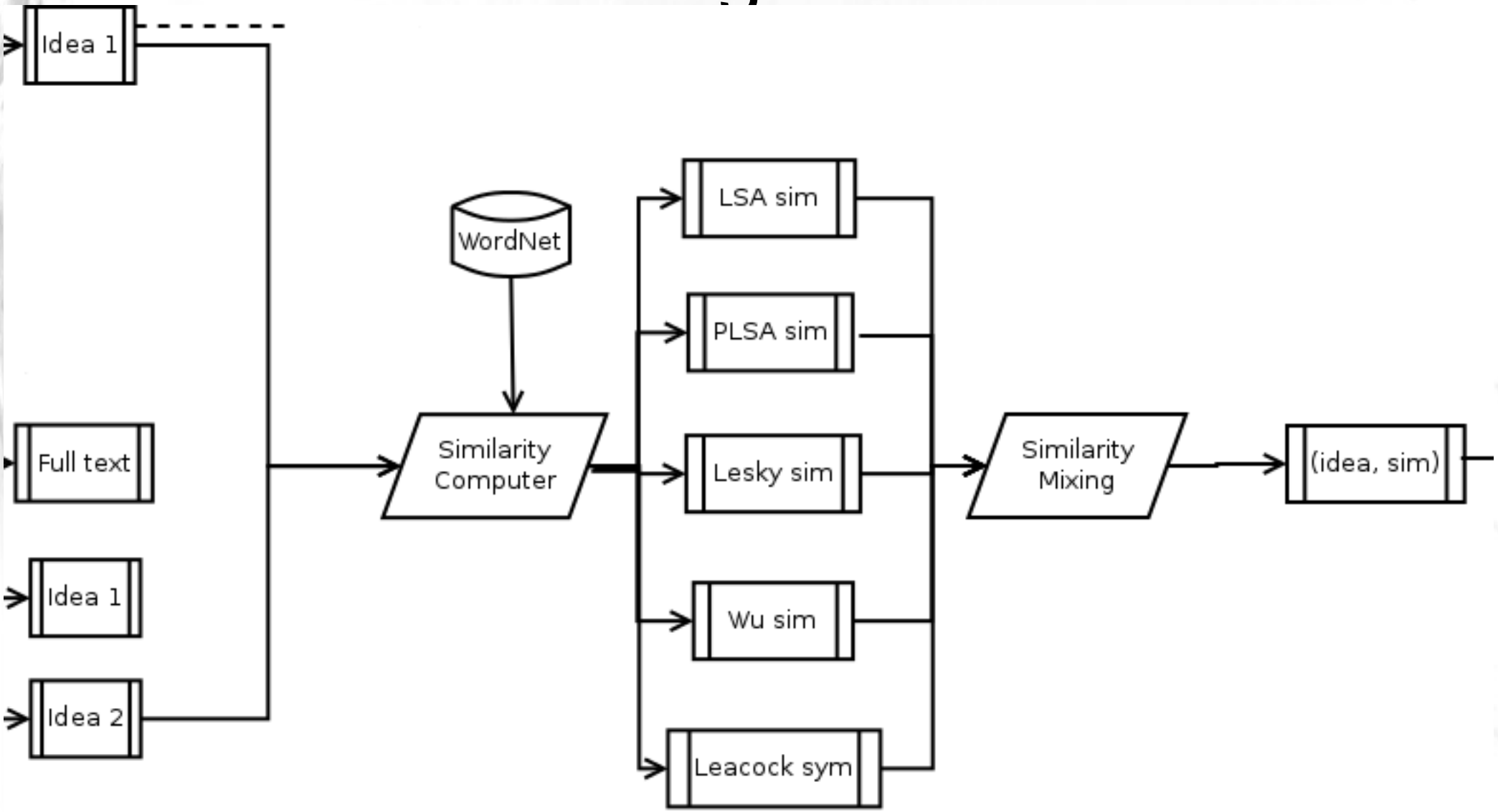
Stage 1

- Increase capabilities
- Weights to filter more important ideas
- Keep full text as well

- Split on paragraphs vs split on connectors

- Both for original and student's work

Stage 2


Idea 1
Full text
Idea 1
Idea 2
WordNet
Similarity Computer
LSA sim
PLSA sim
Lesky sim
Wu sim
Leacock sym
Similarity Mixing
(idea, sim)

Stage 2

- Pair ideas: one from original, one from student's work
- Use multiple similarities metrics
- Mix them before output

- LSA, PLSA
- WordNet (3)

Stage 2 :: WordNet

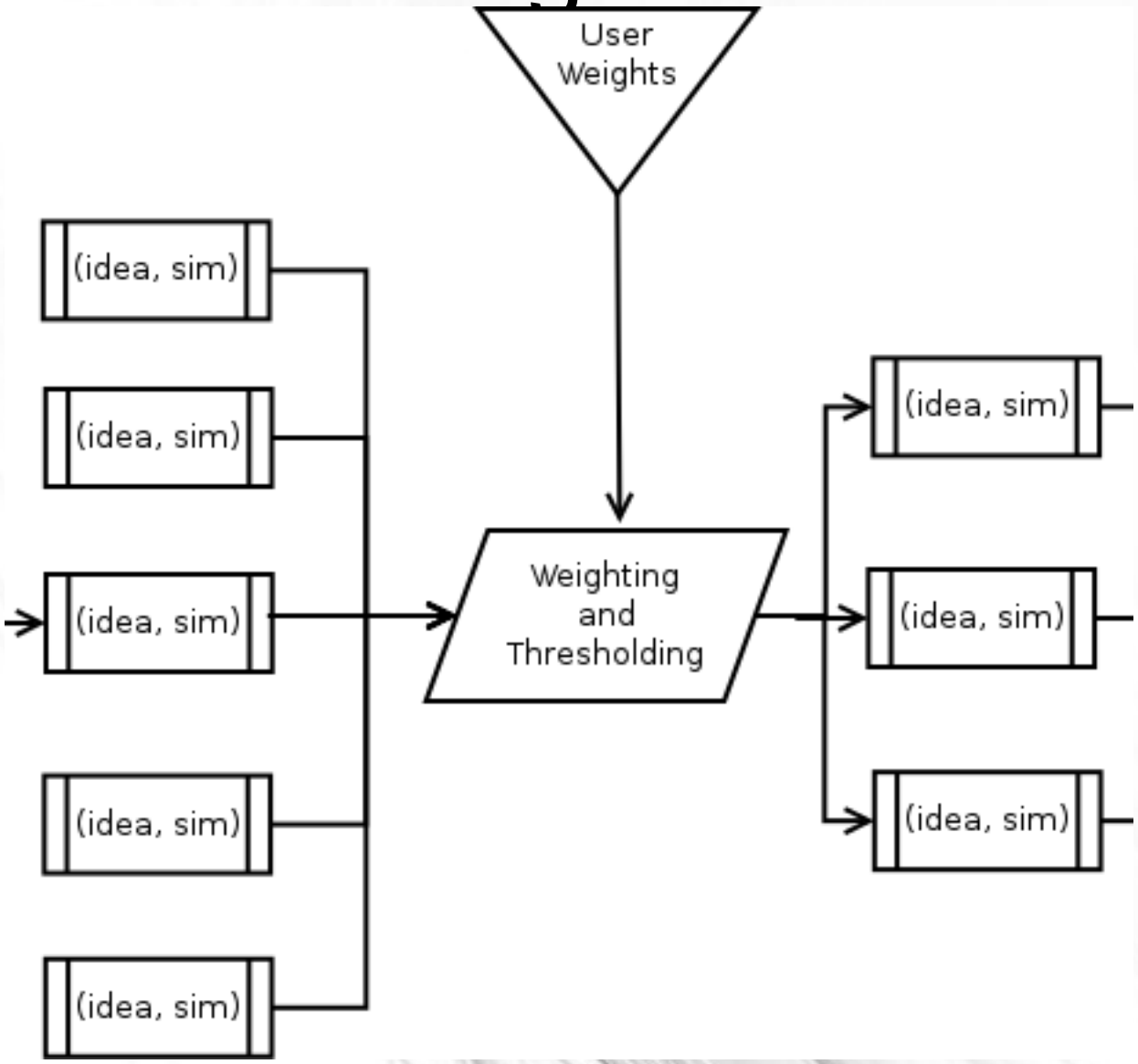
- Lesky, 1986: overlap between dictionary entries
- Wu, 1994:

$$Similarity(w1,w2) = \frac{2 * depth(LCS)}{depth(w1) + depth(w2)}$$

- Leacock, 1998:

$$Similarity = -\log\left(\frac{length}{2 * D}\right)$$

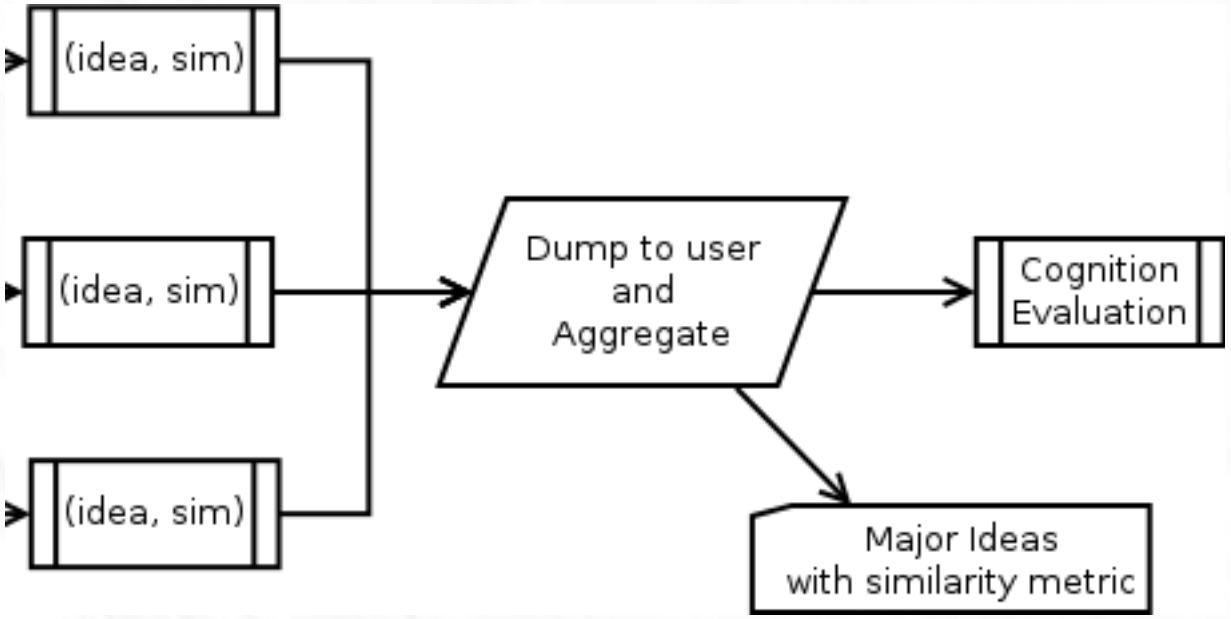
Stage 3


User
Weights
(idea, sim)
(idea, sim)
(idea, sim)
(idea, sim)
(idea, sim)
Weighting
and
Thresholding
(idea, sim)
(idea, sim)
(idea, sim)

Stage 3

- Keep only the relevant pairs
- Tweakable by user prefferences towards some ideas
- Use thresholds
- Filter pairs

Stage 4


(idea, sim)
(idea, sim)
(idea, sim)
Dump to user
and
Aggregate
Cognition
Evaluation
Major Ideas
with similarity metric

Stage 4

- Output each relevant pair
- Aggregate pairs to compute a single metric

- Let user decide what to use from them.

RapidMiner vs WEKA

- RapidMiner
- More text analysis tools
- Command line interface – usable from our application only when needed
- Multilayered-data-view concept mapping closely to our pipeline idea

OpenNLP vs MontiLingua vs GATE

- MontiLingua
- Python, a collection of libraries
- Use what you need
- No training required
- Enriched with common sense knowledge
- Less vulnerable to NLP errors

Thanks