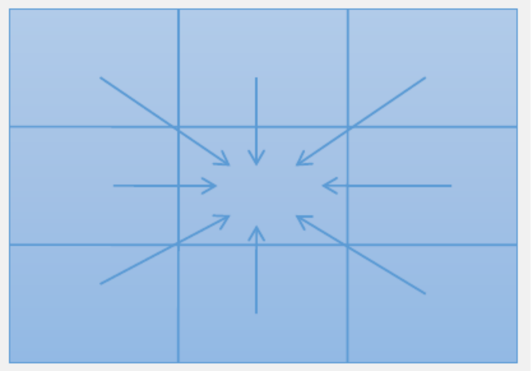
CLOUD COMPUTING
Cloud Applications

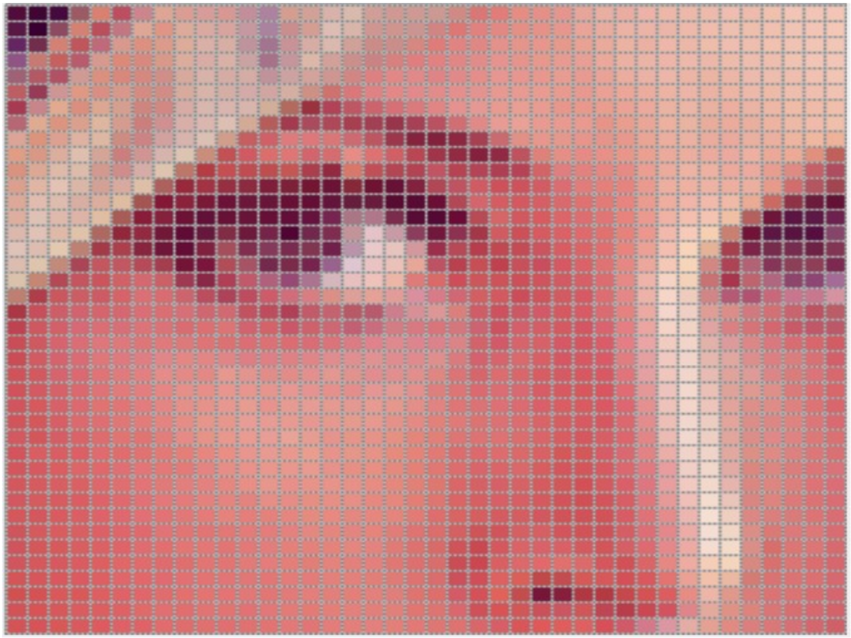
Zeinab Zali
Isfahan University Of Technology

MapReduce Examples

Image Smoothing

- To smooth an image, use a sliding mask and replace the value of each pixel



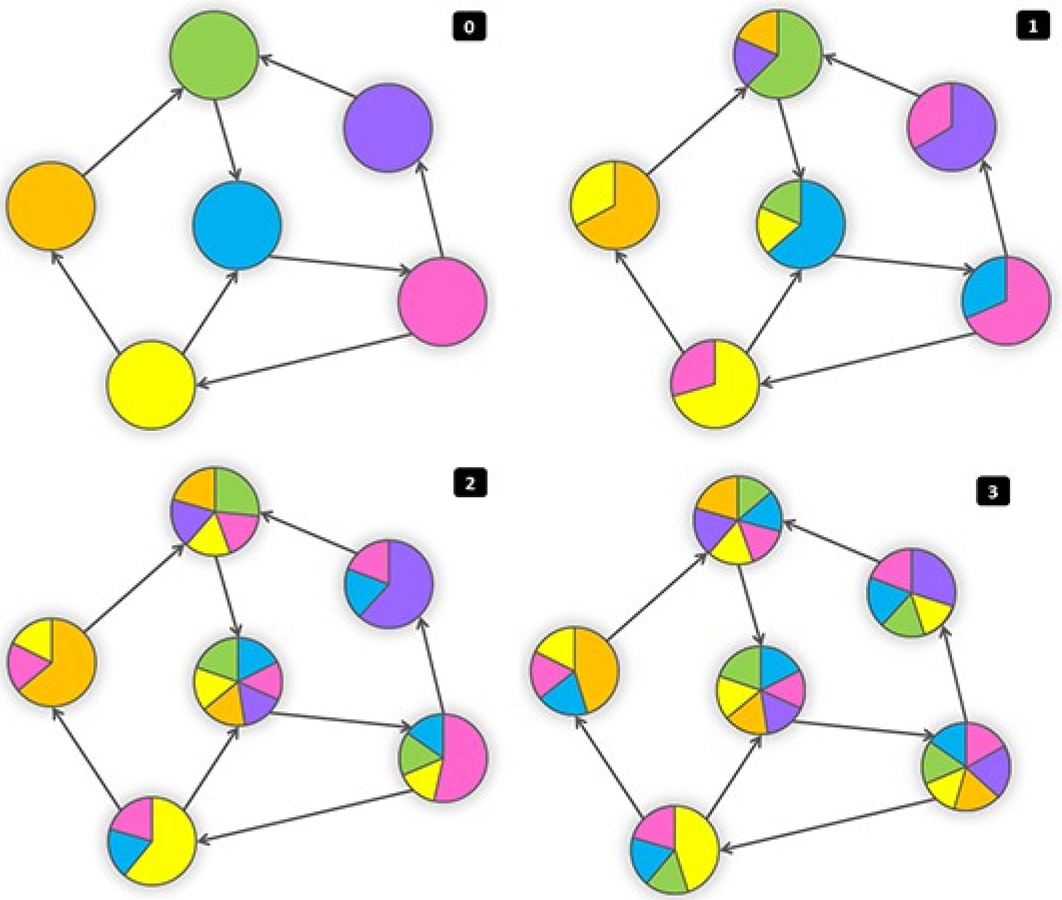


What are Mapper and Reducer?

- Map: input key = x, y input value = R, G, B
  - Emit 9 points
    - (x-1, y-1, R, G, B)
    - (x, y-1, R, G, B)
    - (x+1, y-1, R, G, B)
    - Etc.
- Reduce: input key = x, y input value: list of R, G, B
  - Compute average R, G, B
  - Emit key = x, y value = average R, G, B

Iterative Message Passing (Graph Processing)

- In network of entities and relationships between them, It is required to calculate a state of each entity on the basis of properties of the other entities in its neighborhood
  - Ex: Distance to other nodes



Iterative Message Passing

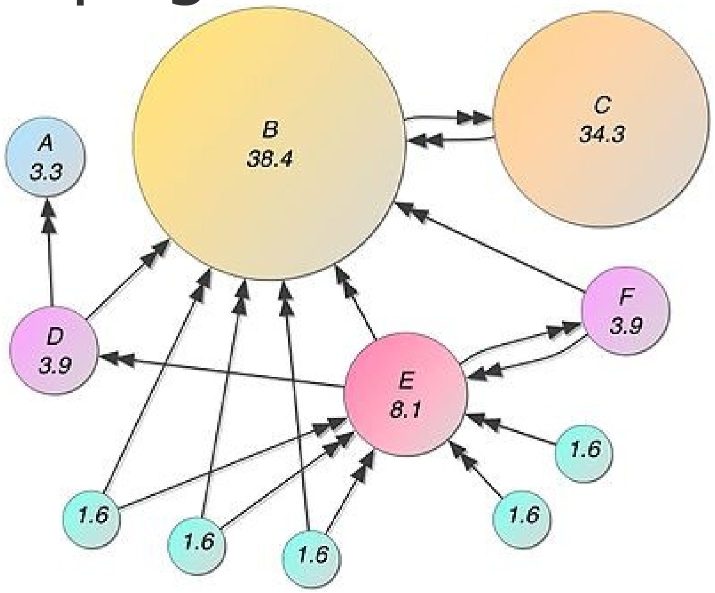
- A network is stored as a set of nodes and each node contains a list of adjacent node IDs
- MapReduce jobs are performed in iterative way
  - at each iteration each node sends messages to its neighbors.
  - Each neighbor updates its state on the basis of the received messages.
- Iterations are terminated by some condition
  - fixed maximal number of iterations (say, network diameter)
  - negligible changes in states between two consecutive iterations

Iterative Message Passing

- Map: for each node, n, in the graph
  - Emit key=n_id, val=n_obj (containing its values)
  - For all the n’s neighbors (outgoing links), m, Emit key=m_id, val=get message(n)
- Reduce: inputs of are records with the same key and all their values, key=n_id,V=[v1, v2, …]
  - new_v=[]
  - For all v in V:
    - If n is object, old_v=v
    - If n is a message new_v.add(v)
  - Emit (n_id, calculate_state(old_v, new_v)

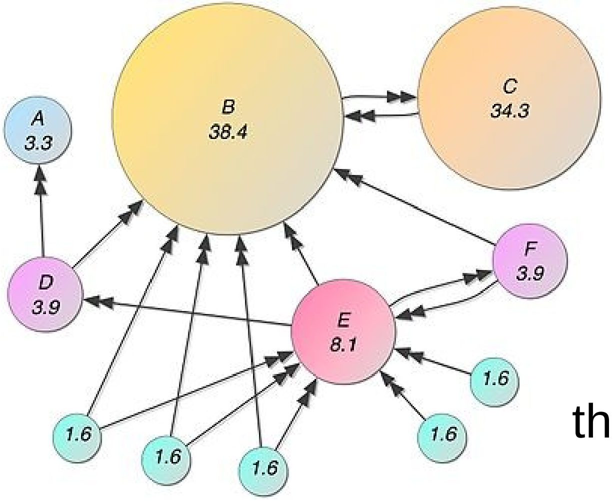
Page Rank

- PageRank is an algorithm used by Google Search to rank web pages in their search engine results
  - It is a way of measuring the importance of website pages



Page Rank Algorithm(I)

- PageRank works by counting the number and quality of links to a page to determine a rough estimate of how important the website is.
  - The underlying assumption is that more important websites are likely to receive more links from other websites



Number of outbound links of v

$$PR(u) = \sum_{v \in B_u} \frac{PR(v)}{L(v)}$$

the set containing all pages linking to page u

Page Rank Algorithm(II)

- Phase 1: Propagation
- Phase 2: Aggregation
- Input: a pool of objects including both vertices and edges
  - Vertex: the web page
  - Edge: the link to another page

Propagation: What are Mapper and Reducer?

- Map: for each object
  - If it is a vertex, emit key=URL, val=obj
  - If it is an edge, emit key=source URL, val=obj
- Reduce:input is a web page and all the outgoing links
  - Find the number of edge objects → outgoing links
  - Read the pageRank value from the vertex obj
  - Assign PR(edges) = PR(vertex)/num_outgoing

Aggregation: What are Mapper and Reducer?

- Map: for each object
  - If it is a vertex: emit key=URL, val=obj
  - If it is an edge: emit key=destination URL, val = obj
- Reduce: input is a web page and all the incoming links
  - Add PR value of all the incoming links
  - Assign PR(vertex)=ΣPR(incoming links)