

# Document Clustering

# Clustering

- Automatically group related documents into *clusters*.
- Example
  - Medical documents
  - Legal documents
  - Financial documents

# Uses of Clustering

- If a collection is well clustered, we can search only the cluster that will contain relevant documents.
- Searching a smaller collection should improve effectiveness and efficiency.

# Clustering Algorithms

- Hierarchical Agglomerative Clustering
  - requires a pre-computed doc-doc similarity matrix.
- Clustering without a pre-computed doc-doc similarity matrix.

# Hierarchical Clustering Algorithms

- Single Link
- Complete Linkage
- Group Average
- Ward's Method

# Hierarchical Agglomerative

- Create  $N \times N$  doc-doc similarity matrix
- Each document starts as a cluster of size one
- Do Until there is only one cluster
  - combine the two clusters with the **greatest similarity**
  - update the doc-doc matrix
- Note: Various means of computing the *cluster similarity* results in different flavors of this algorithm.

# Example

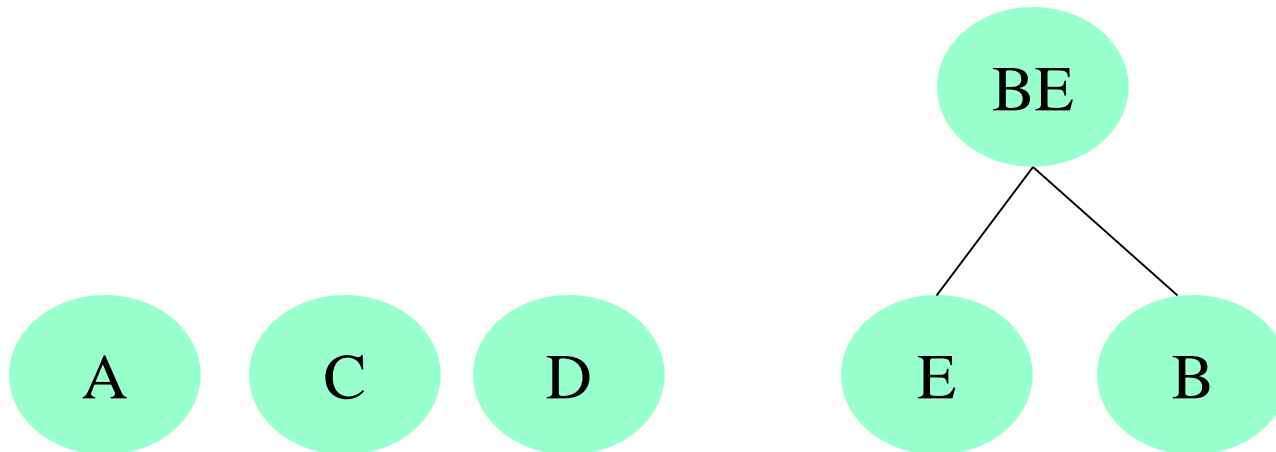
- Consider A, B, C, D, E as documents with the following similarities:

	<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>E</b>
<b>A</b>	-	2	7	9	4
<b>B</b>	2	-	9	11	14
<b>C</b>	7	9	-	4	8
<b>D</b>	9	11	4	-	2
<b>E</b>	4	14	8	2	-

Highest pair is: E-B = 14

# Example

- So lets cluster E and B. We now have the structure:



# Example

- Now we update the DOC-DOC matrix.

	A	BE	C	D
A	-	2	7	9
BE	2	-	8	2
C	7	8	-	4
D	9	2	4	-

Note: To compute BE --  $SC(A, B) = 2$   
 $SC(A, E) = 4$

$SC(A, BE) = 4$  if we are using **single link** (take max)

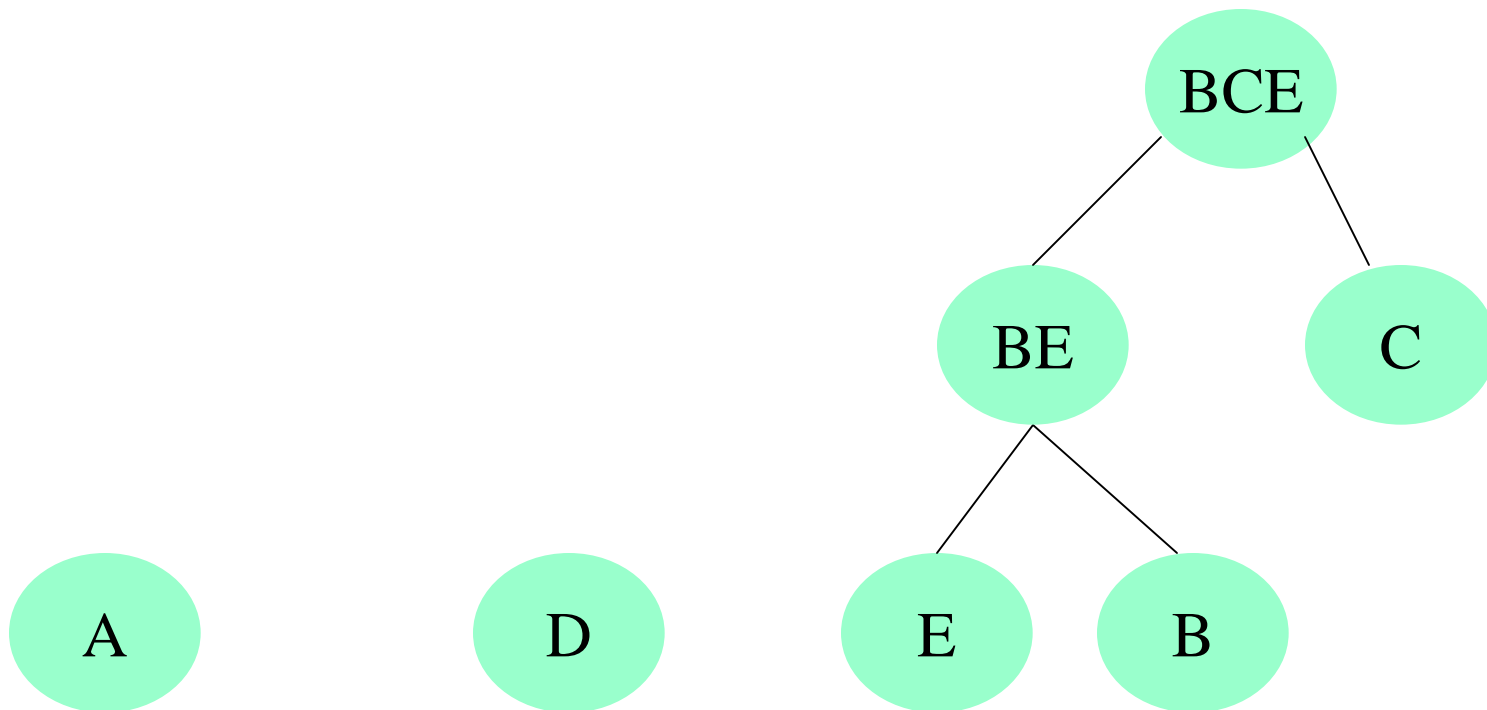
$SC(A, BE) = 2$  if we are using **complete linkage** (take min)

$SC(A, BE) = 3$  if we are using **group average** (take average)

Note: C - BE is now the highest link

# Example

- So lets cluster BE and C. We now have the structure:



# Example

- Now we update the DOC-DOC matrix.

	<b>A</b>	<b>BCE</b>	<b>D</b>
<b>A</b>	-	2	9
<b>BCE</b>	2	-	2
<b>D</b>	9	2	-

To compute  $SC(A, BCE)$ :

$$SC(A, BE) = 2$$

$$SC(A, C) = 7 \text{ so } SC(A, BCE) = 2$$

To Compute:  $SC(D, BCE)$

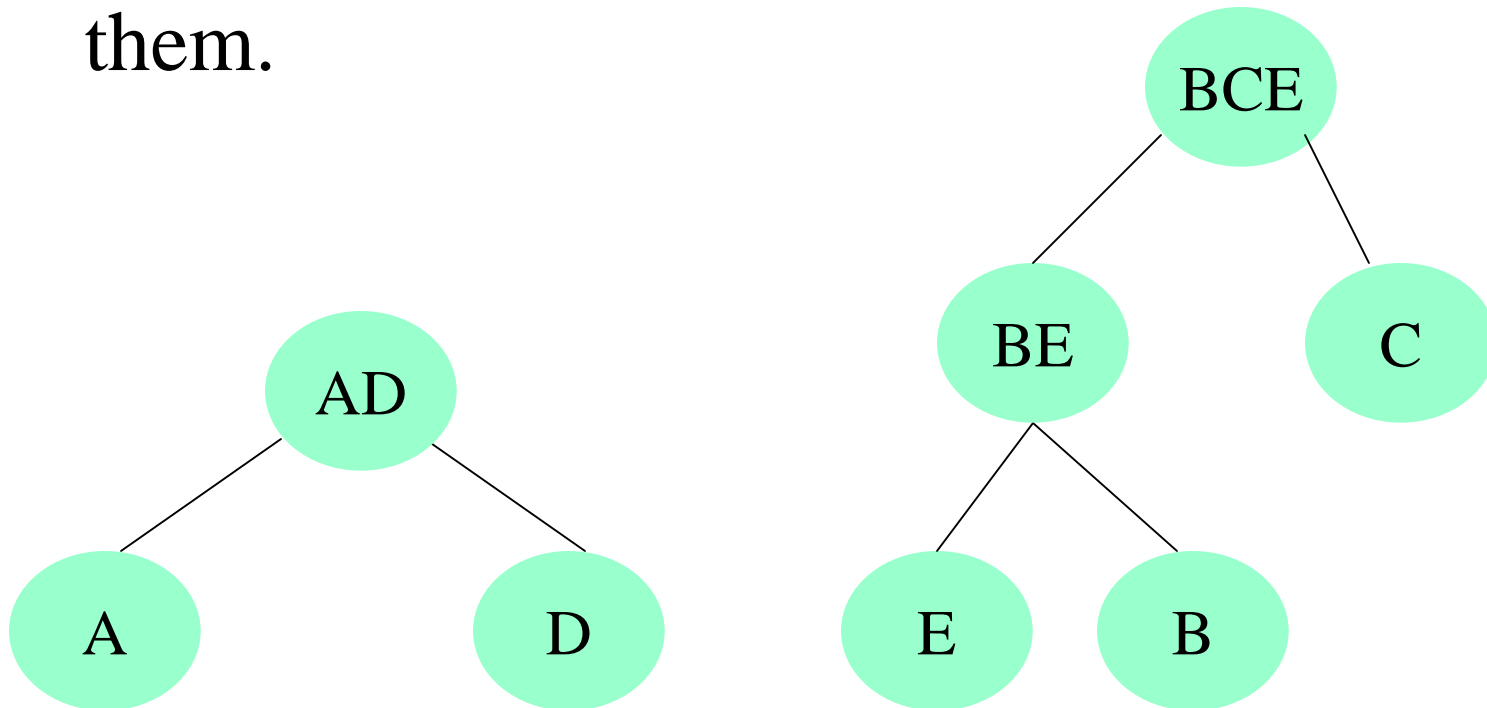
$$SC(D, BE) = 2$$

$$SC(D, C) = 4 \text{ so } SC(D, BCE) = 2$$

$SC(D, A) = 9$  which is greater than  $SC(A, BCE)$  or  $SC(D, BCE)$   
so we now cluster A and D.

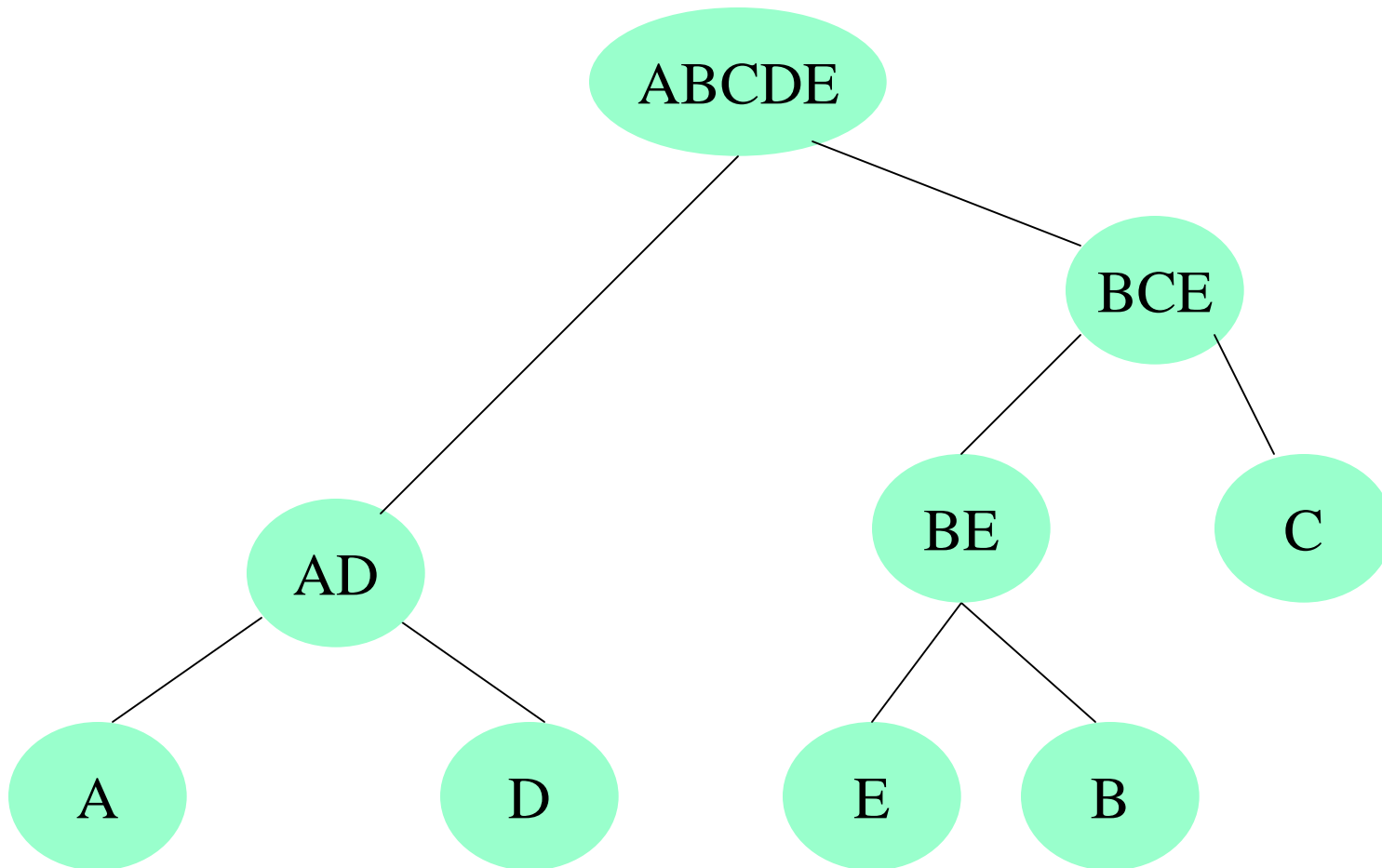
# Example

- So lets cluster A and D. At this point, there are only two nodes that have not been clustered, AD and BCE. We now cluster them.



# Example

- Now we have clustered everything.



# Analysis

- Hierarchical clustering requires:
  - $O(n^2)$  to compute the doc-doc similarity matrix
  - One node is added during each round of clustering so there are now  $O(n)$  steps
  - For each clustering step we must re-compute the DOC-DOC matrix. This requires  $O(n)$  steps.
  - So we have:
  - $n^2 + (n)(n) = O(n^2)$  -- so its very expensive.
  - For 500,000 documents  $n^2$  is 250,000,000,000!!<sub>14</sub>

# Other Clustering Algorithms

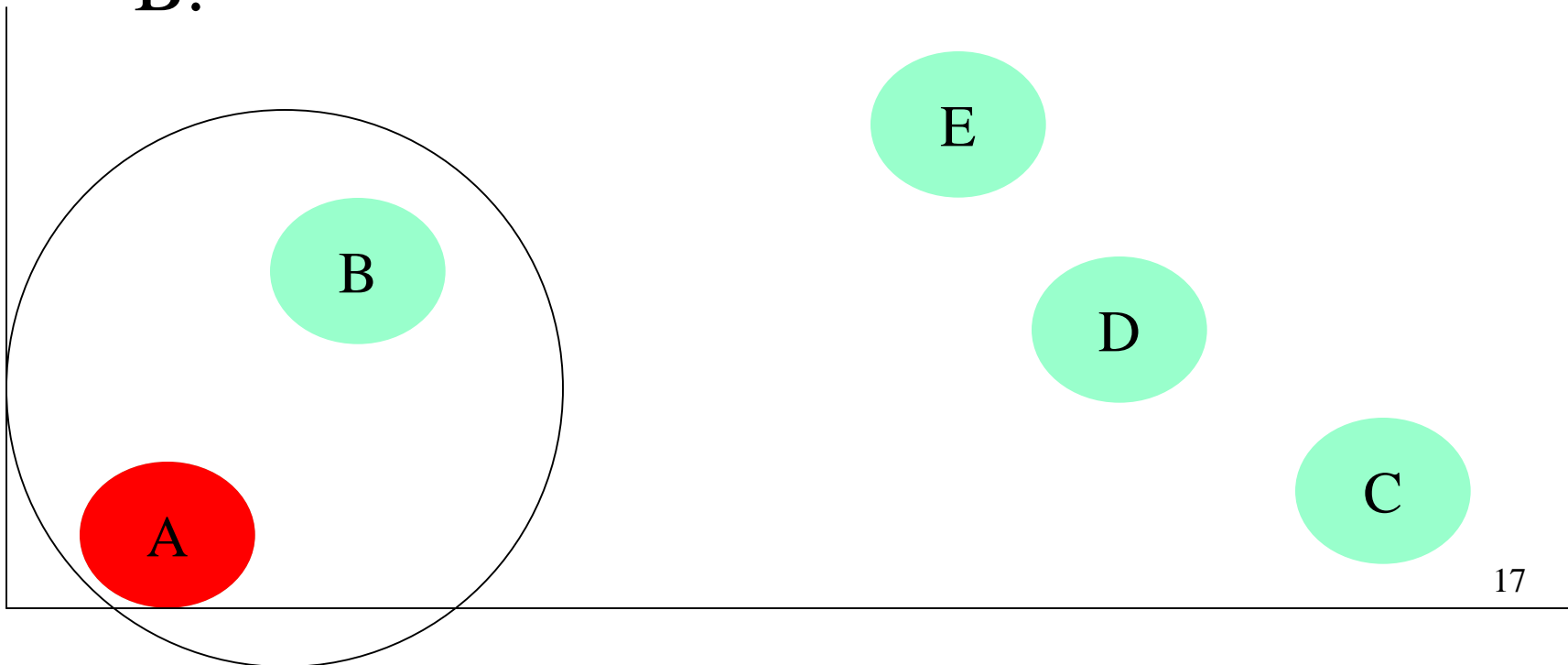
- One-pass
- Buckshot

# One pass Clustering

- Choose a document and declare it to be in a cluster of size one.
- Now compute distance from this cluster to all remaining nodes.
- Add “closest” node to the cluster. If no node is really close (within some threshold), start a new cluster between the two closest nodes.

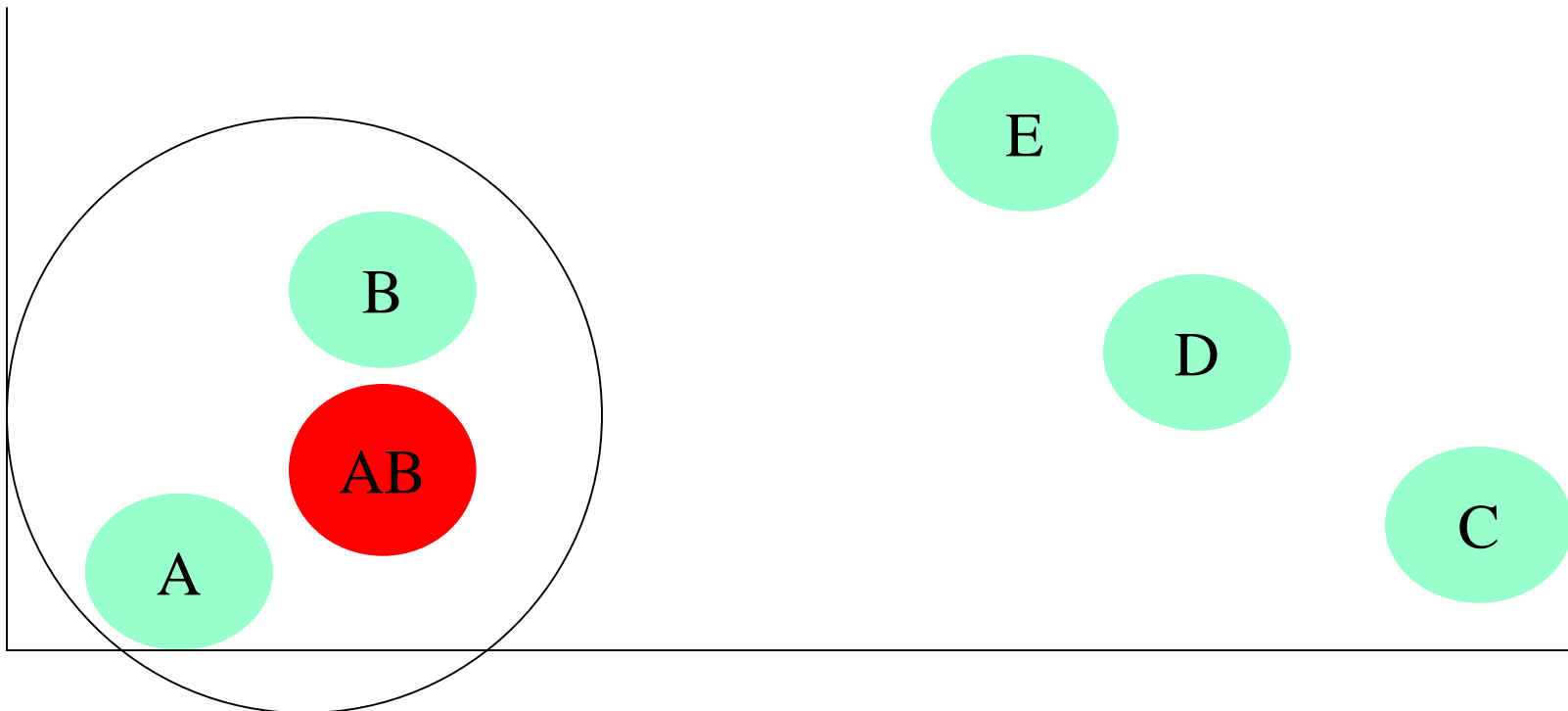
# Example

- Choose node A as the first cluster
- Now compute  $SC(A,B)$ ,  $SC(A,C)$  and  $SC(A,D)$ . B is the closest so we now cluster B.



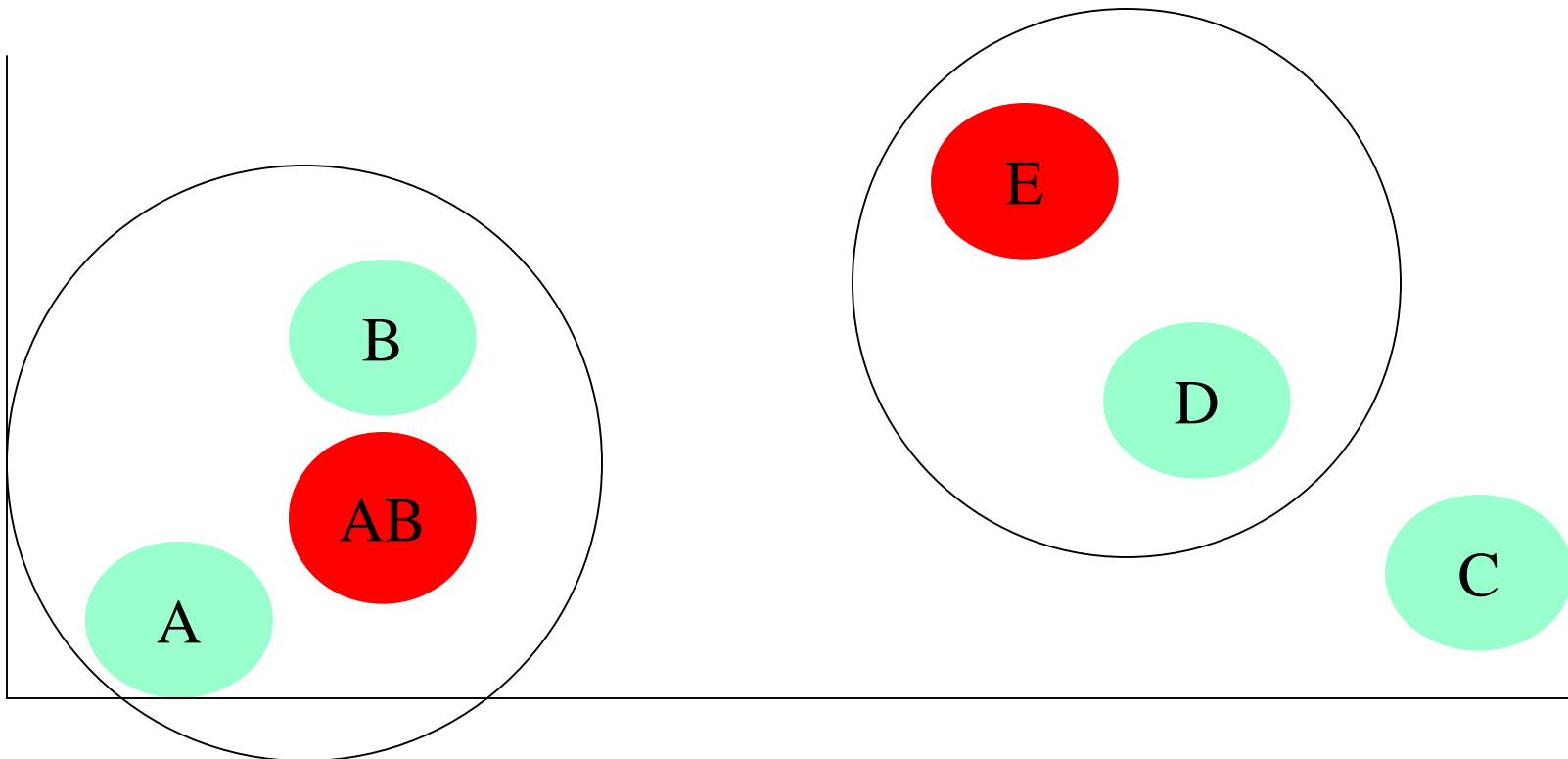
# Example

- Now we compute the centroid of the cluster just formed and use that to compute SC between the cluster and all remaining clusters. Lets assume its too far from AB to E, D, and C. So now we choose one of these non-clustered element and place it in a cluster. Lets choose E.



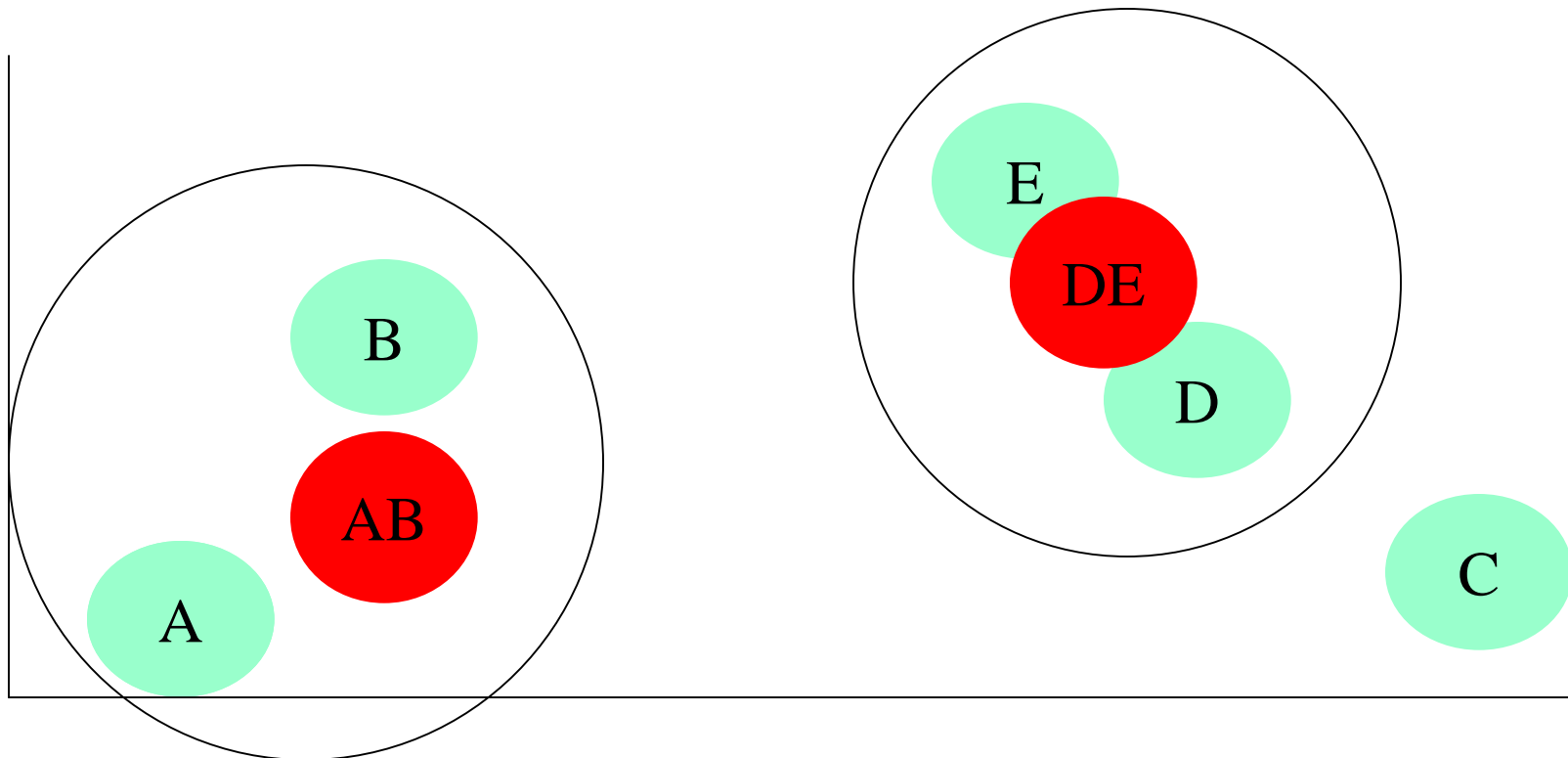
# Example

- Now we compute the distance from E to D and E to C. E to D is closer so we form a cluster of E and D.



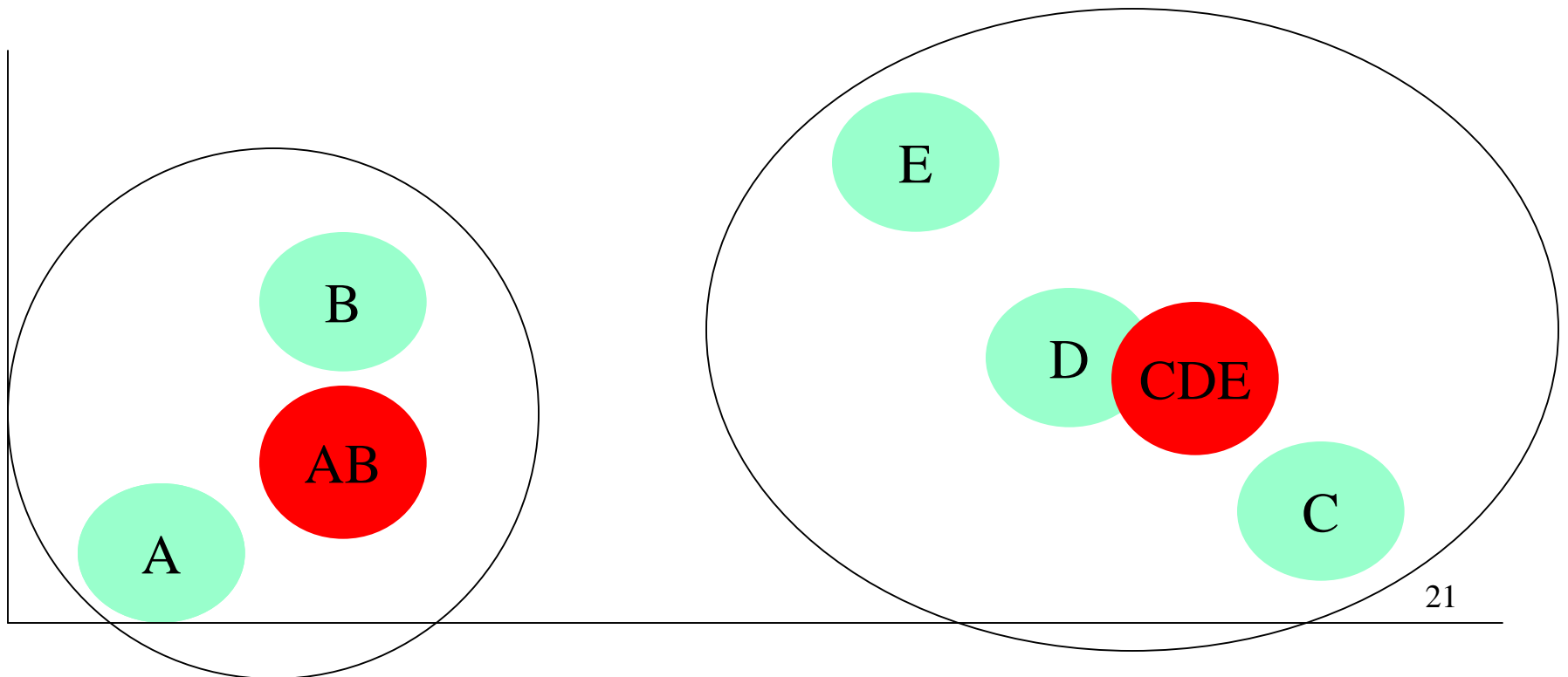
# Example

- Now we compute the centroid of D and E and use that to compute the  $SC(DE, C)$ . It is within the threshold so we now include C in this cluster.



# Example

- Now we compute the centroid of D and E and use that to compute the  $SC(DE, C)$ . It is within the threshold so we now include C in this cluster.



# Analysis of One Pass

- N passes as we add one node for each pass
- First pass requires  $n-1$
- Second pass requires  $n-2$
- Last pass is 1
- So we have  $1+2+3\dots+n = (n)(n-1) / 2$
- $O(n) + O(n^2)$
- Note: Constant is lower for one pass but we are still at  $O(n^2)$

# Effect of Document Order

- Hierarchical clustering we get the same clusters every time
- One pass clustering, we get different clusters based on the order we use to process the documents

# Buckshot Clustering

- Relatively recent technique (1992).
- Goal is to reduce run time to  $O(kn)$  instead of  $O(n^2)$  where  $k$  is the number of clusters.

# Buckshot Algorithm

- Randomly select  $d$  documents where  $d$  is  $\text{SQRT}(kn)$
- Cluster these using any hierarchical clustering algorithm. We now have  $kn$  clusters.
- Compute the centroid of each of the  $kn$  clusters
- Scan remaining documents and assign them to the closest of the  $kn$  clusters.
- This requires  $O(kn) + O(kn) + O(kn)$
- Note if  $k$  is small, this is close to  $O(n)$

# Summary

- Pro
  - Clustering can work to give users an overview of the contents of a document collection
  - Also can reduce the search space
- Con
  - Computationally expensive
  - Difficult to identify which cluster or clusters should be searched