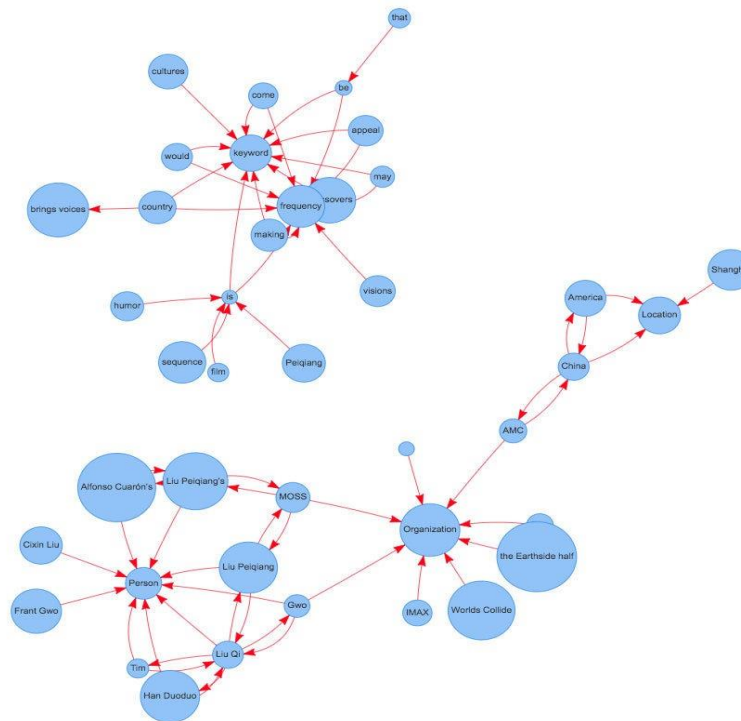
 <b>Marwadi University</b>	<b>Marwadi University</b> <b>Faculty of Technology</b> <b>Department of Information and Communication Technology</b>	
<b>Subject: Artificial Intelligence (01CT0616)</b>	<b>Aim:</b> Representation of a document with their keywords using TextRank	
<b>Experiment No: 09</b>	<b>Date:</b>	<b>Enrolment No:</b>

**Aim:** Representation of a document with their keywords using TextRank

**IDE:** Google Colab


## **Theory:**

TextRank is an algorithm based on PageRank, which often used in keyword extraction and text summarization. In this article, I will help you understand how TextRank works with a keyword extraction example and show the implementation by Python.



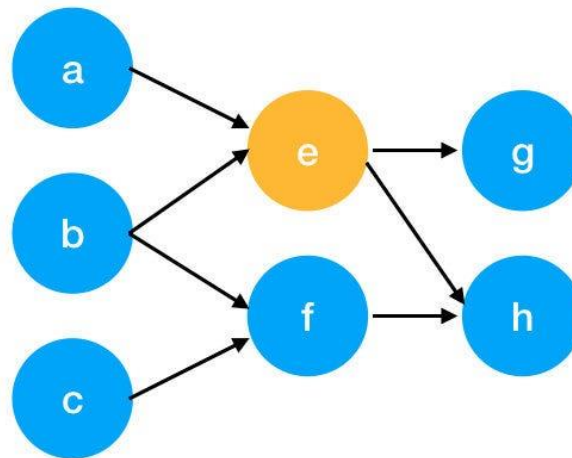
## **Understand PageRank**

There are tons of articles talking about PageRank, so I just give a brief introduction to PageRank. This will help us understand TextRank later because it is based on PageRank. PageRank (PR) is an algorithm used to calculate the weight for web pages. We can take all web pages as a big directed graph. In this graph, a node is a webpage. If webpage A has the link to web page B, it can be represented as a directed edge from A to B. After we construct the whole graph, we can assign weights for web pages by the following formula.

 <b>Marwadi University</b>	<b>Marwadi University</b> <b>Faculty of Technology</b> <b>Department of Information and Communication Technology</b>	
<b>Subject: Artificial Intelligence (01CT0616)</b>	<b>Aim:</b> Representation of a document with their keywords using TextRank	
<b>Experiment No: 09</b>	<b>Date:</b>	<b>Enrolment No:</b>

$$S(V_i) = (1 - d) + d * \sum_{j \in In(v_i)} \frac{1}{|Out(V_j)|} S(V_j)$$


- $S(V_i)$  - the weight of webpage i
- d - damping factor, in case of no outgoing links
- $In(V_i)$  - inbound links of i, which is a set
- $Out(V_j)$  - outgoing links of j, which is a set
- $|Out(V_j)|$  - the number of outbound links



Here is an example to better understand the notation above. We have a graph to represent how web pages link to each other. Each node represents a webpage, and the arrows represent edges. We want to get the weight of webpage e. We can rewrite the summation part in the above function to a simpler version.

$$\begin{aligned}
 In(v_e) &= \{a, b\}, j \in \{a, b\} \\
 \sum_{j \in \{a, b\}} \frac{1}{|Out(V_j)|} S(V_j) &= \frac{1}{|Out(V_a)|} S(V_a) + \frac{1}{|Out(V_b)|} S(V_b) \\
 &= \frac{1}{|\{e\}|} S(V_a) + \frac{1}{|\{e, f\}|} S(V_b) \\
 &= S(V_a) + \frac{1}{2} S(V_b)
 \end{aligned}$$

We can get the weight of webpage e by the following function.

 <b>Marwadi</b> University	<b>Marwadi University</b> <b>Faculty of Technology</b> <b>Department of Information and Communication Technology</b>	
<b>Subject: Artificial Intelligence (01CT0616)</b>	<b>Aim:</b> Representation of a document with their keywords using TextRank	
<b>Experiment No: 09</b>	<b>Date:</b>	<b>Enrolment No:</b>

$$S(V_e) = (1 - d) + d * \left( S(V_a) + \frac{1}{2} S(V_b) \right)$$

We can see the weight of the webpage e is dependent on the weights of its inbound pages. We need to run this iteration much time to get the final weight. In the initialization, the importance of each webpage is 1.

### **Pre Lab Exercise:**

1. What are the key components of TextRank?

---



---



---

2. What are some limitations of TextRank for keyword extraction?

---



---



---

3. What are the advantages of TextRank for keyword extraction?

---



---



---


### **Program (Code):**

To be attached with

### **Results:**

To be attached with

### **Observation:**

 <b>Marwadi</b> University	<b>Marwadi University</b> <b>Faculty of Technology</b> <b>Department of Information and Communication Technology</b>	
<b>Subject: Artificial Intelligence (01CT0616)</b>	<b>Aim:</b> Representation of a document with their keywords using TextRank	
<b>Experiment No: 09</b>	<b>Date:</b>	<b>Enrolment No:</b>

### **Post Lab Exercise:**

1. Write about “your ownself” (in not more than 500 words→ You know better about you, rather than ChatGPT!!). Extract top-25 keywords for your portfolio and analyze it in 3 categories: keyword can not give the real idea, keyword gives the real idea, keyword is irrelevant. Paste the code, your portfolio, output and your analysis.