भारतीय सूचना प्रौद्योगिकी संस्थान धारवाड़
INDIAN INSTITUTE OF INFORMATION TECHNOLOGY DHARWAD

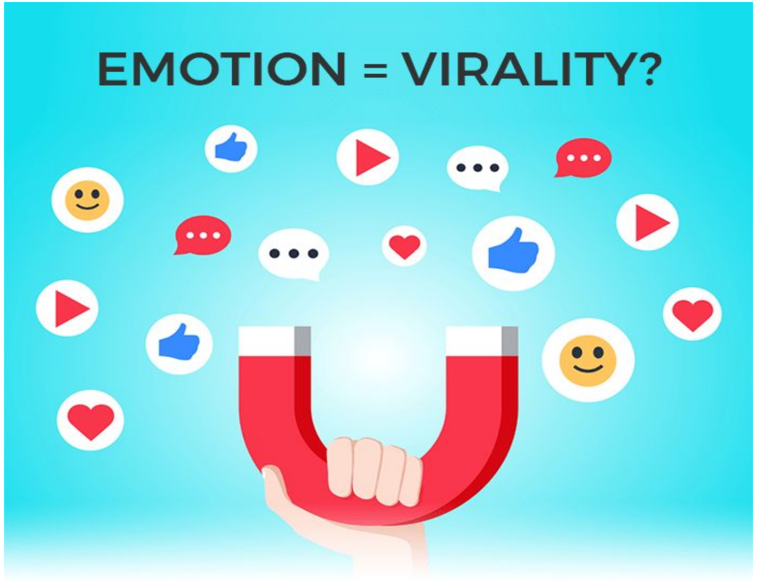
Pseudo-Cores

Name: Somyadeep Shrivastava (17BCS028)
Dheeraj Chaudhary (17BCS009)

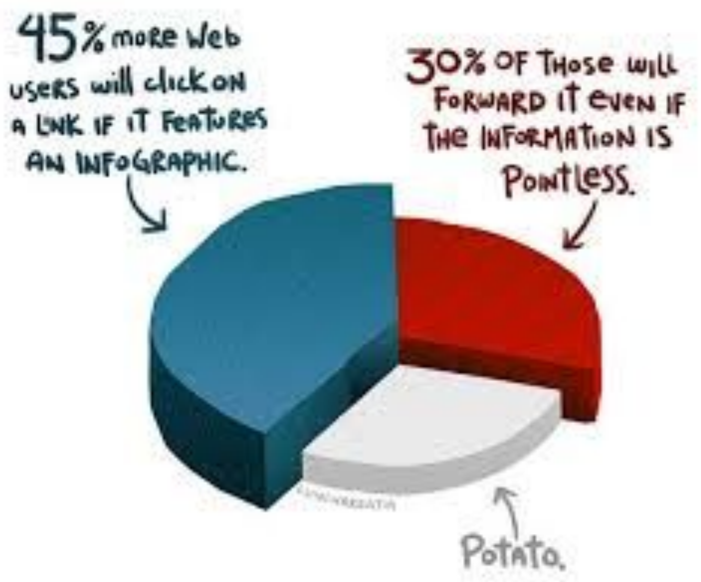
Under the Guidence of: Dr Yayati Gupta

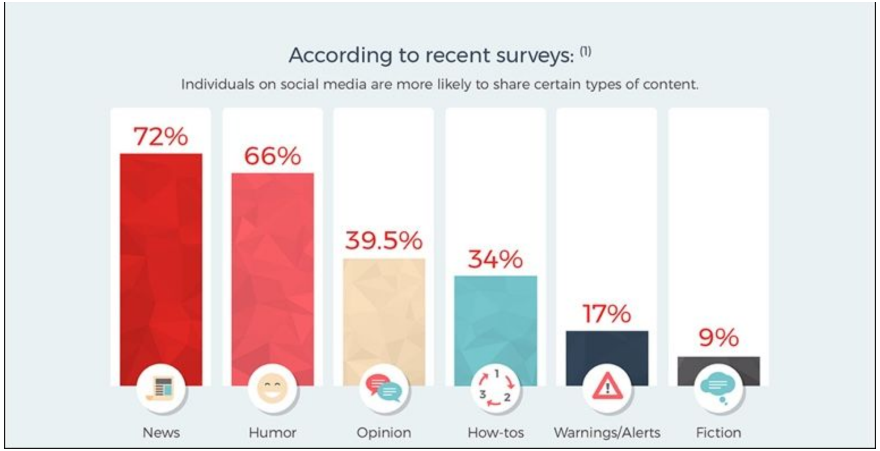
Virality

The tendency of an image, video, or piece of information to be circulated rapidly and widely from one Internet user to another



Virality


45% more Web users will click on a LINK IF IT FEATURES AN INFOGRAPHIC.
30% OF THOSE WILL FORWARD IT EVEN IF THE INFORMATION IS POINTLESS.
Potato.

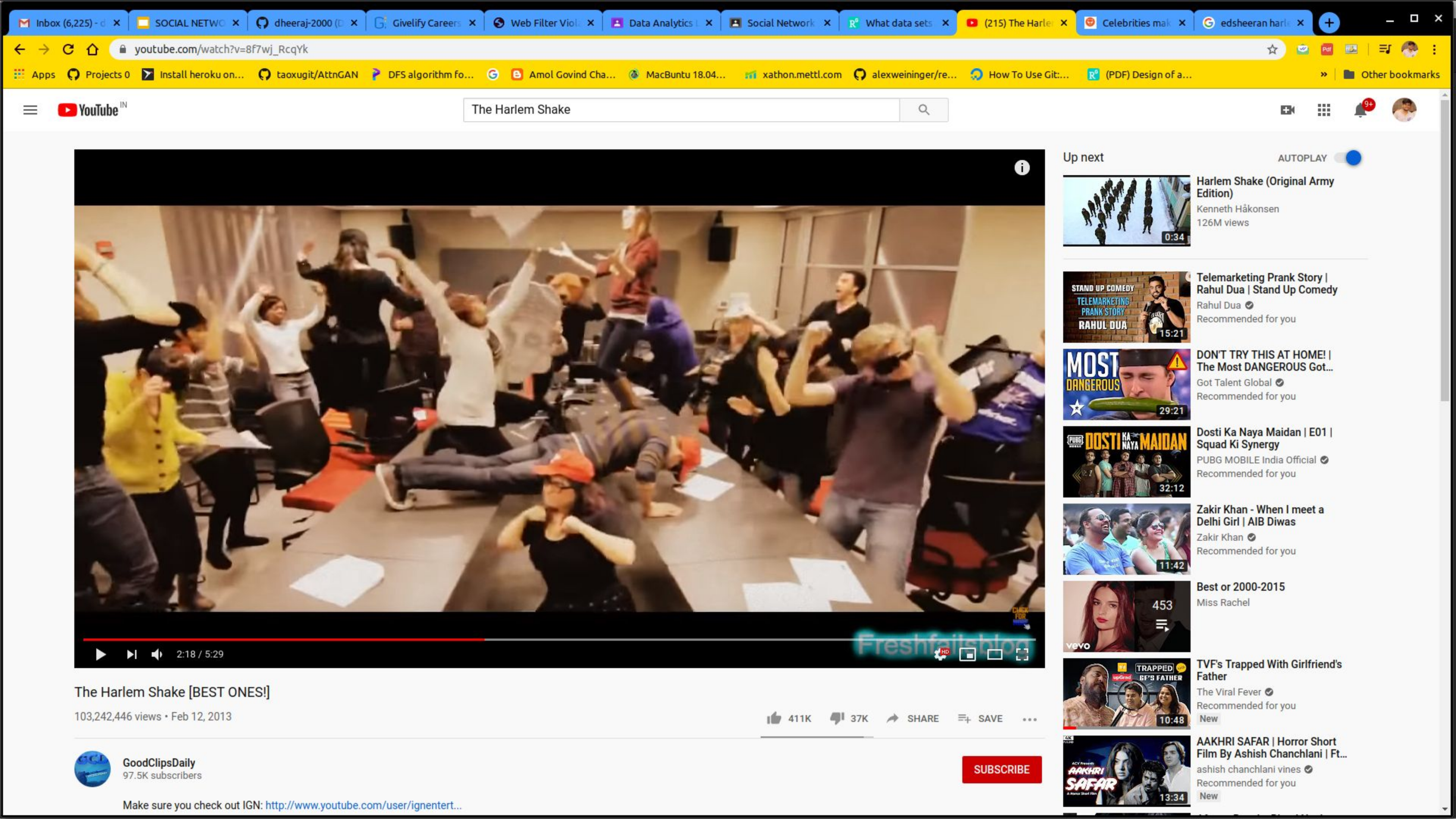

According to recent surveys: (1)
Individuals on social media are more likely to share certain types of content.
72%
66%
39.5%
34%
17%
9%
News
Humor
Opinion
How-tos
Warnings/Alerts
Fiction

Core : The Destination for a Viral Meme

- The application of core-periphery structure came to light as a consequence of the criticism of the Milgram's experiment by Judith Kleinfeld .
- Milgram's small world experiment aimed at passing a letter from a source person to the target person through a chain of acquaintances.
- The letter could only be forwarded to a person whom the participant knew on a first name basis.
- Milgram's experiment found the median of the chain lengths over multiple experiments to be six. Later on, Kleinfeld observed that the success rates of the recreations of Milgram's experiment were very low . Kleinberg clarified that the chain length was observed to be six only when the destination was a high status individual or "core", otherwise the chain length was higher .
- These two properties of **maximum spreading power** and **high reachability** make the core a suitable destination for a viral meme.

Examples Of Role of cores in virality

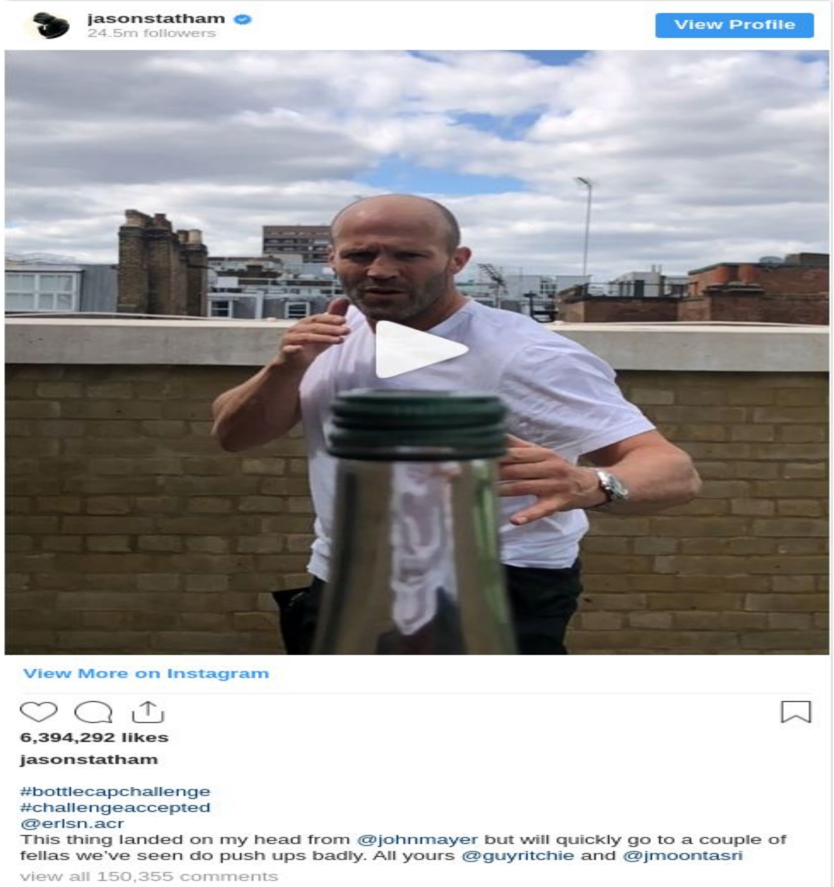
Examples Of Role of cores in virality



youtube.com/watch?v=8f7wj_RcqYk
The Harlem Shake
Up next
AUTOPLAY
Harlem Shake (Original Army Edition)
Kenneth Håkonsen
126M views
Telemarketing Prank Story | Rahul Dua | Stand Up Comedy
Rahul Dua
Recommended for you
DON'T TRY THIS AT HOME! | The Most DANGEROUS Got...
Got Talent Global
Recommended for you
Dosti Ka Naya Maidan | E01 | Squad Ki Synergy
PUBG MOBILE India Official
Recommended for you
Zakir Khan - When I meet a Delhi Girl | AIB Diwas
Zakir Khan
Recommended for you
Best or 2000-2015
Miss Rachel
TVF's Trapped With Girlfriend's Father
The Viral Fever
Recommended for you
New
AAKHRI SAFAR | Horror Short Film By Ashish Chanchlani | Ft...
ashish chanchlani vines
Recommended for you
New
2:18 / 5:29
The Harlem Shake [BEST ONES!]
103,242,446 views • Feb 12, 2013
411K
37K
SHARE
SAVE
GoodClipsDaily
97.5K subscribers
SUBSCRIBE
Make sure you check out IGN: http://www.youtube.com/user/ignentert...

Examples Of Role of cores in virality

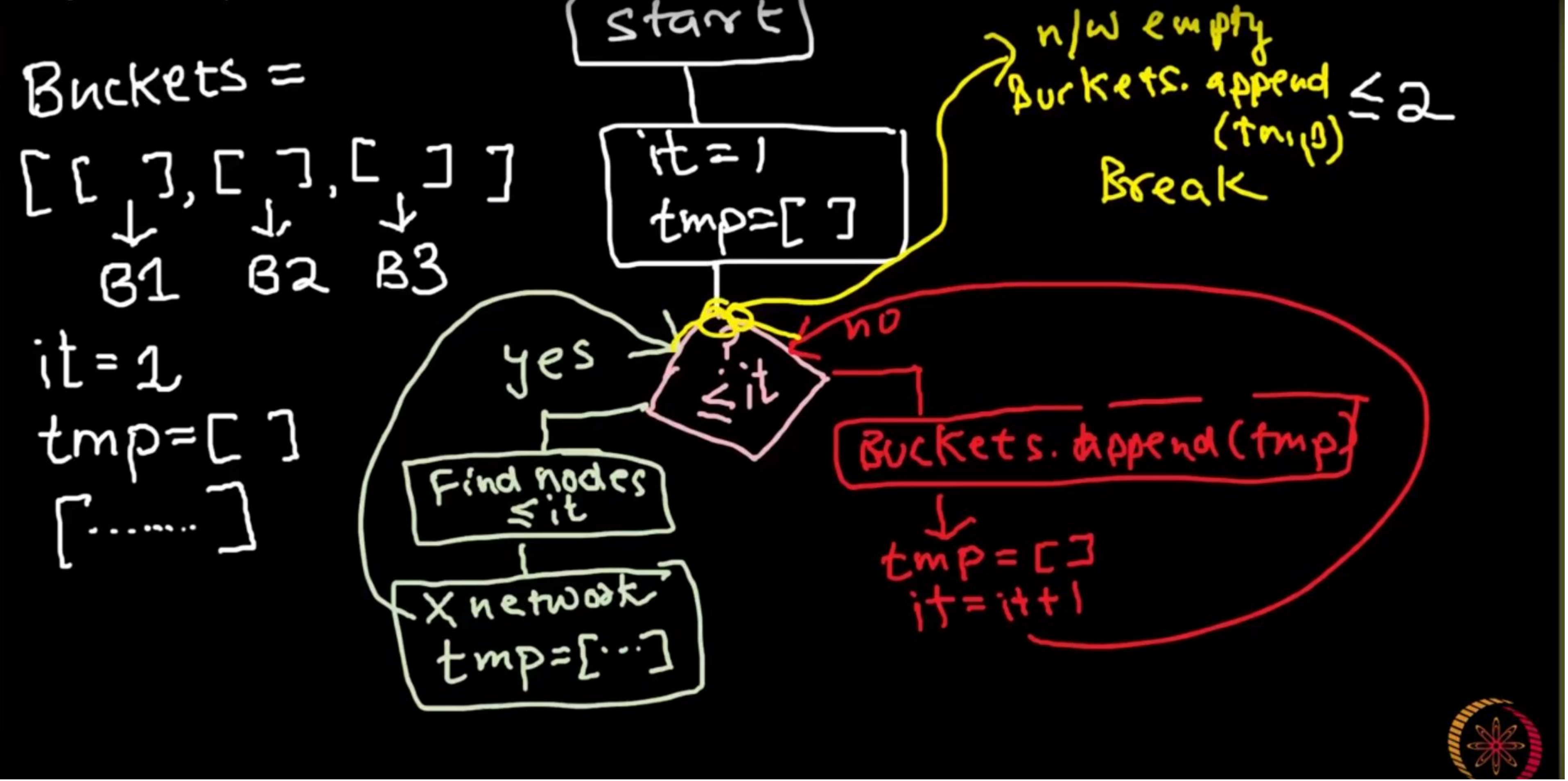
Hollywood star Jason Statham also attempted it:

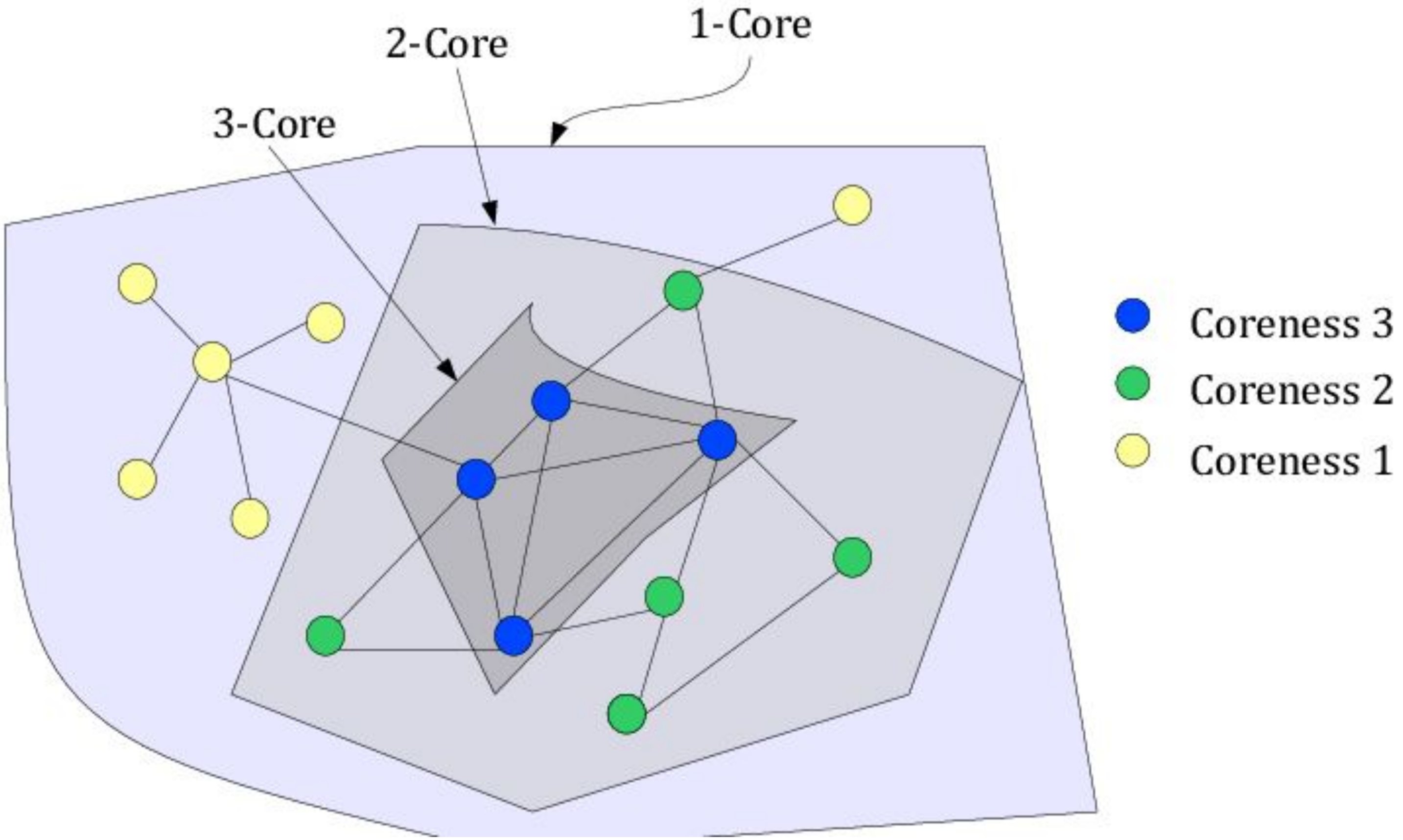


K-Shell Decomposition

- This algorithm was proposed by Kitsak et al. in order to determine the core in a network.
- The algorithm works by recursively pruning the lower degree nodes in a network. First the nodes having degree $d(u) \leq 1$ are pruned recursively till there are no nodes of degree 1 left in the network.
- Similarly, the process is repeated for the nodes k having degree $d(k)$ where $d(k)=2, 3, \ldots, n-1$ till the graph becomes empty.
- At every step, the pruned nodes can be visualised as being kept in a basket.
- **One of the key facts of the algorithm is that the nodes having high degrees lying at the periphery are pruned earlier because of recursive pruning.**
- Therefore, *it is not necessary that a person having high degree should also be a core node*.

Coding K-Shell Decomposition




1-Core
2-Core
3-Core
Coreness 3
Coreness 2
Coreness 1

Correlation of network properties with the shell number

Distribution of nodes across shells:

- Though it is expected that no. of nodes decrease with increase in shell number but according to plot it shows increase in no. of nodes in intermediary shell, but still diameter of core is least.

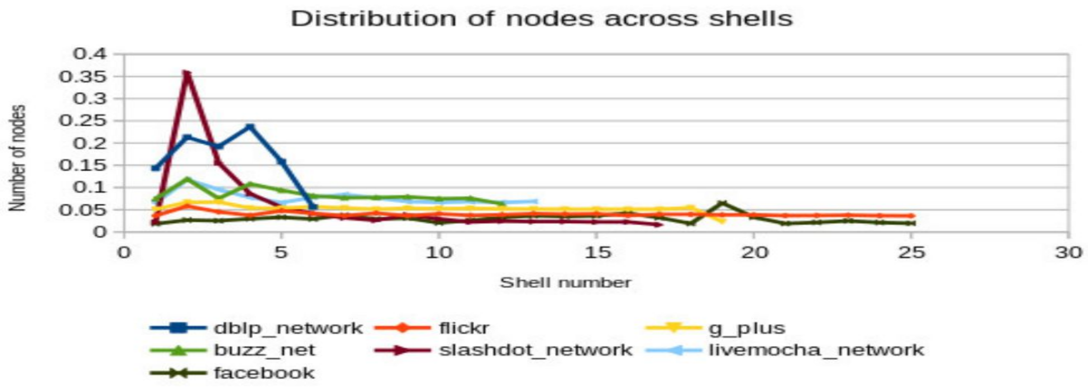
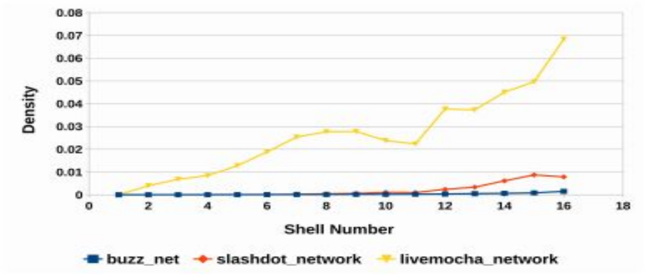


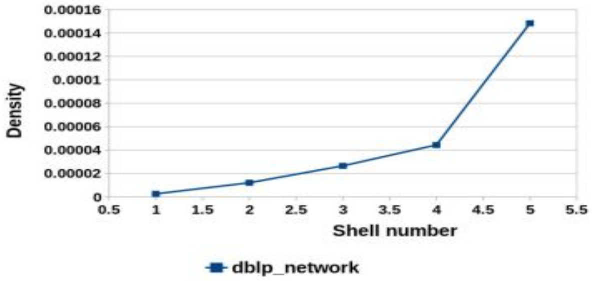
Fig. 1. Distribution of nodes across shells for various networks

Density of shells

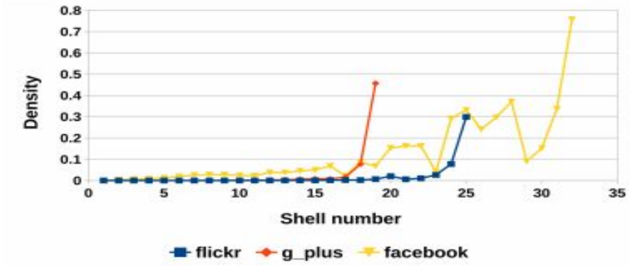
- The density of shell i is defined as the density of the induced subgraph Si on the vertices that lie in shell i of the network G. Let |V (Si)| be the number of nodes in the subgraph Si and |E(Si)| be the number of edges. Then, the density of shell i can be given as |E(Si)| (|V (Si)| 2)
- When the core shell is encountered in these networks, there is a sudden but powerful spike in density. High density of the core is one of the reason for the high spreading power of the core.
- However, in the case of social networks like Facebook, the curve does not accelerate only on reaching the core, there are quite a few spikes in between and the curve does not remain monotonically increasing.



(a) Density of shells in buzznet, slashdot and livemocha



(b) Density of shells in DBLP

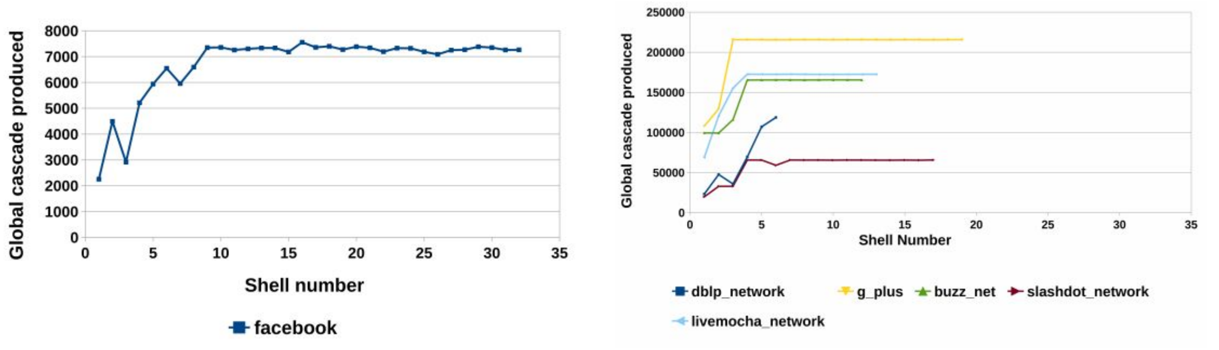


(c) Density of shells in Flickr, Google Plus and Facebook

Fig. 2. Shellwise Density distribution in Real world networks

Cascading Power of Different Shells in a Network:

- We choose nodes uniformly at random from a shell. Then we use the independent cascade model for meme propagation and initiate the process of infection over this network.
- Therefore at every iteration, each infected node infects its neighbours with a constant probability. We note that every node gets only one chance to infect its neighbours.
- The process stops as soon as it reaches an iteration where no new node is infected. The number of nodes infected by the end of the process is called the cascading power of a shell.
- One ideally expects the cascade to accelerate when core shell is encountered, but it is observed that the acceleration point is reached much before the point where the meme reaches the core. This intriguing fact led us to investigate the existence of a ***"pseudo- core shell"***



(a) Cascading power of facebook shells (b) Cascading power of other real world network's shells

Fig. 3. Shell wise cascading power distribution

Finding path to core and psuedocores

❖ Using traditional algorithms:-

Random walk algorithm

- Inspecting its neighbor and choose randomly one of the node amongst them.
- Algorithm terminates, if chosen node is core node
- Time Complexity - $O(n)$

Degree based hill climbing:-

- The algorithm uses a hill climbing approach
- At every step, a node looks at its neighbours and chooses the unexplored node having the highest degree.
- Algorithm terminates, if chosen node is core node
- Hill Climbing has a time complexity of max[O(n), O(m)] ~ O(n) in sparse graphs.

❖ Shell based Hill Climbing Approach (SH):

- Let G(V, E) represent the graph where V (G) is the set of vertices and E(G) is the set of edges.
- Let the number of vertices and edges in G be n and m respectively. *Shell(u)* represents the shell number of a node u as calculated by the k-shell decomposition algorithm.
- Start is the periphery node from where the meme starts spreading. *N G(u)* represents the set of neighbours of node u in the graph G.
- The proposed SH approach has a complexity of max[O(m + n), O(n)] ~ O(n) in sparse graphs.

Shell based Hill Climbing Approach (SH):

Algorithm 1 Shell Based Hill Climbing(SH)

procedure FINDNUMSTEPS
- **Input**:- Graph $G(V, E)$, Starting node $start$
- **Output**: Number of steps taken by the algorithm to terminate
- Apply k-shell decomposition and calculate $shell(u)\ \forall u \in V(G)$
- $visited[u] \leftarrow \text{`}false\text{'}\ \forall u \in V(G)$
- $numsteps \leftarrow 0$
- $current \leftarrow start$
- $visited[current] \leftarrow \text{`}true\text{'}$
- **while** $current$ is not a core node **do**
  - $v_1 \leftarrow argmax_{u \in N_G(current) \wedge visited[u]=\text{`}false\text{'}}\, shell(u)$
  - **if** $shell(v_1) \leq shell(current)$ **then**
    - $v_2 \leftarrow random\ node\ u\ \in N_G(current) \wedge visited[u] = \text{`}false\text{'}$
    - $current \leftarrow v_2$
  - **else**
    - $current \leftarrow v_1$
  - $numsteps \leftarrow numsteps + 1$
- return $numsteps$

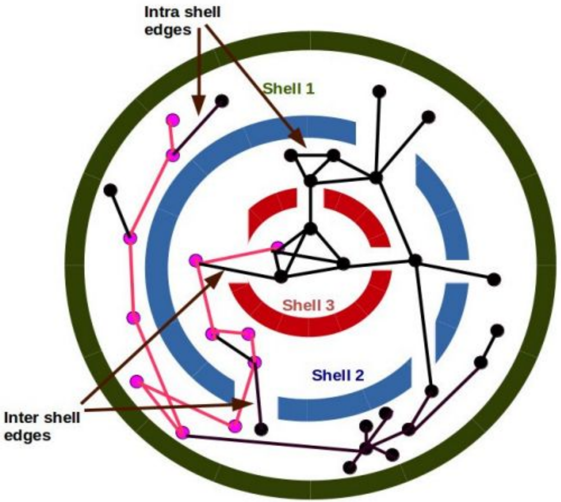
Intershell Hill Climbing with Intrashell Degree Based Approach(SA):

- Algorithm 2 is a modification of algorithm 1 and utilises the idea that a node with very high degree will cover most of the shell.
- If this node is chosen, it would greatly reduce the number of steps required to traverse a shell.
- Let the number of vertices and edges in G be n and m respectively. N G (u) represents the set of neighbours of node u in the graph G.
- The proposed SA approach has a complexity of max[O(m + n), O(n), O(m)] ~ O(n) in sparse graphs.

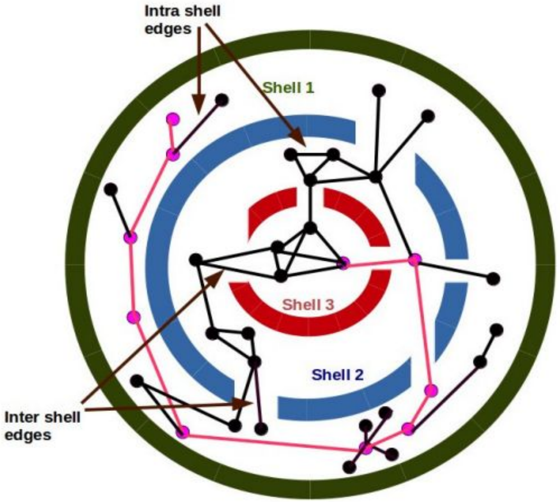
❖ Intershell Hill Climbing with Intrashell Degree Based Approach(SA):

Algorithm 2 Improved Shell Based Hill Climbing(SA)

procedure FINDNUMSTEPS
  Input:- Graph $G(V, E)$, Starting node *start*
  Output: Number of steps taken by the algorithm to terminate
  Apply k-shell decomposition and calculate $shell(u)\ \forall u \in V(G)$
  $visited[u] \leftarrow `false'\ \forall u \in V(G)$
  $numsteps \leftarrow 0$
  $current \leftarrow start$
  $visited[current] \leftarrow `true'$
  while *current* is not a core node **do**
    $v_1 \leftarrow argmax_{u \in N_G(current) \wedge visited[u]=`false'} shell(u)$
    if $shell(v_1) \leq shell(current)$ **then**
      $v_2 \leftarrow argmax_{u \in N_G(current) \wedge visited[u]=`false'} degree(u)$
      $current \leftarrow v_2$
    else
      $current \leftarrow v_1$
    $numsteps \leftarrow numsteps + 1$
  return *numsteps*



(a) Algorithm1- Shell Based Hill Climbing(SH)



(b) Algorithm2- Modified Shell Based Hill Climbing(SA)

Experimental Results:

- To evaluate the performance of the algorithms mentioned in the above section, we select periphery nodes from shell 1(periphery) in a network and for each of these nodes, we find the number of steps taken to reach the core.

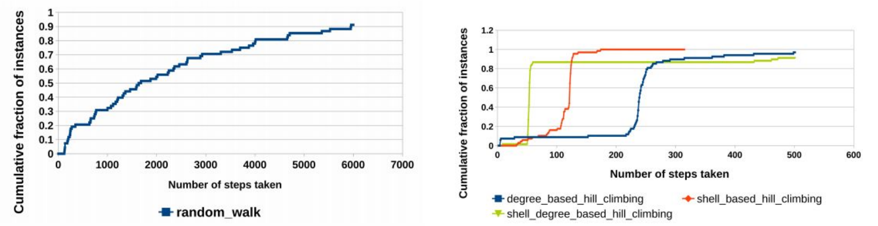
- We term each run from a periphery node as an instance of the problem. Therefore, we can say the number of instances is equal to the number of periphery nodes.

- It is observed that more than 80% of the walks conclude in a maximum of 15 steps in most of the datasets.

- Without the loss of generality, we have ignored the trivial case where source nodes are directly connected to the core as the path length in these cases is 1.

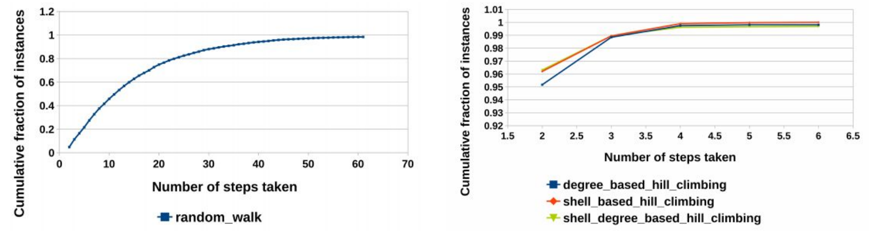
- Let R be a random variable whose value ranges from 2 to k.
- R depicts the number of steps taken by the algorithm to terminate. Let P (R = k) be the probability of value of R being k where k ranges from 2 to 15.
- We plot the cumulative probability distribution function of R. X axis indicates all possible values of R while Y axis shows the probability of R ≤ k.
- The plots given below validate that the proposed algorithms cover most of the instances in very less number of steps as compared to the existing path finding algorithms.
- ***The highest line in the curve represents the most efficient algorithm.***

Core as a destination



(a) Random Walk (b) Shell Based Hill Climbing Algorithms

Fig. 5. Comparison of algorithms for Facebook



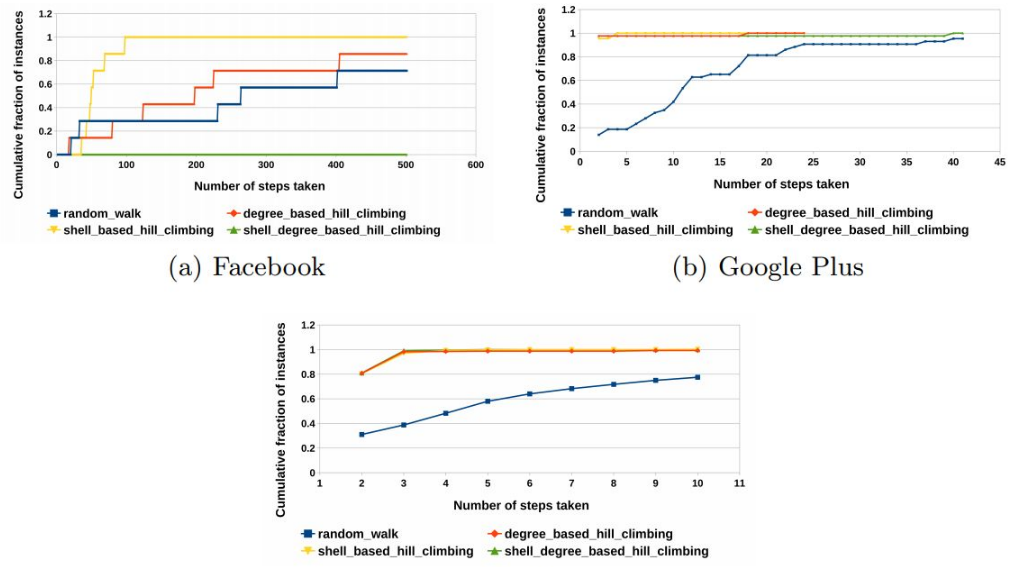
(a) Random Walk (b) Shell Based Hill Climbing Algorithms

- Next, we modify the destination to be the pseudo-core shells and observe the cumulative frequency distribution of R as given in the next slide.

- In this case also, our proposed algorithms perform better than the other algorithms.

- **Interestingly, the performance of even the random walk algorithm increases drastically when the target is changed to be the pseudo-cores.**

Psuedocore as a destination



(a) Facebook
(b) Google Plus
(c) Slashdot

Fig. 7. Comparison of algorithms for infecting pseudo-cores

Extension to existing research

- Leakage power denotes a shell’s potential to take quick and long jumps to the higher numbered shells.
- There were indications of high leakage power in intermediary shells as well apart from core
- This **led the idea of “teleportation shells” and “barricade shells”**. The shells having higher leakage powers act as the teleportation shells and can trigger a meme to take longer jumps on its path to the core.
- On the other hand, the shells having low leakage powers may tend to block a meme inside it, hence suggesting why some memes are non viral.
- To determine coreness of a shell locally

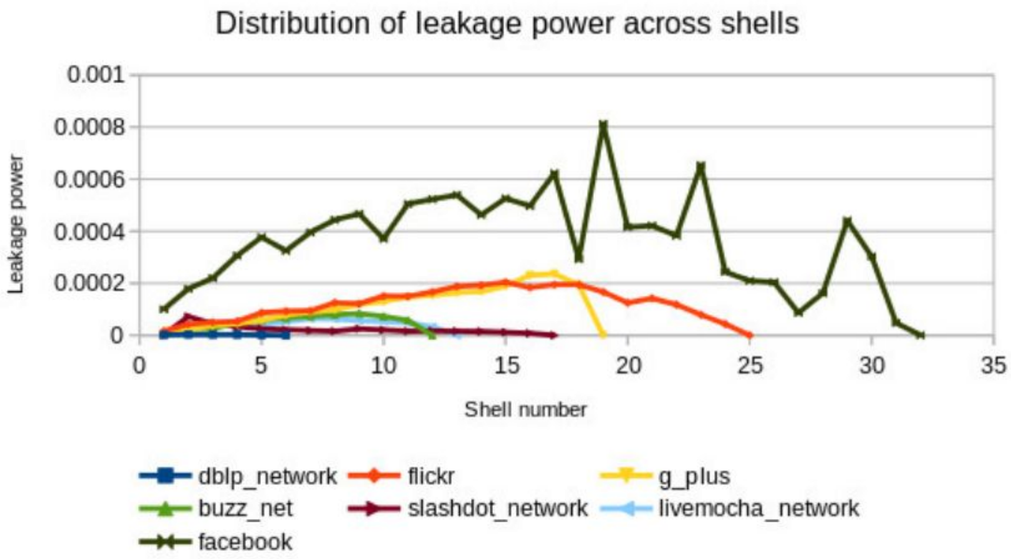


Fig. 8. Distribution of leakage power across shells for various networks

Dataset for the project

Datasets used in the Paper

Dataset	Description
Facebook	Facebook is the most popular Social Networking Site today. This dataset consists of anonymized friendship relations from Facebook. [23]. The network contains 4,039 nodes and 88,234 edges.
Google plus	Google plus is a social layer for Google Services.[23]. The network contains 107,614 nodes and 13,673,453 edges.
Slashdot	Slashdot is a website where the users can submit and evaluate the news stories on science and technology. It is famous for its specific user community. [24]. This dataset contains 82,168 nodes and 948,464 edges.
Flickr	This is an image and video hosting site. It is mainly used for sharing and embedding personal photographs. [5]. This dataset has 80513 nodes and 5899882 edges.
Livemocha	Livemocha is the world's largest online language learning community.[5]. This dataset has 104438 nodes and 2196188 edges.
DBLP	The DBLP computer science bibliography is a collaboration network. It provides a detailed list of research papers in computer science [25]. The network contains 317,080 nodes and 1,049,866 edges.
Buzznet	Buzznet is social media network used for sharing photos, journals, and videos. It has 101168 nodes and 4284534 edges.

Other Datasets

		Nodes	edges	
ego-Twitter	Directed	81,306	1,768,149	Social circles from Twitter
musae-github	Undirected	37,700	289,003	Social network of Github developers.

Thank You