



UNIVERSITÀ  
DEGLI STUDI DI BARI  
ALDO MORO



DIPARTIMENTO  
DI INFORMATICA



# Applicazione di tecniche di Word Embedding e Text Mining per il Clustering di pagine in un grafo Web

Tesi di Laurea in Programmazione II  
Informatica e Tecnologie per la Produzione del Software

Relatore:

Prof. Michelangelo Ceci

Correlatore:

Dott.ssa Pasqua Fabiana  
Lanotte

Laureando:

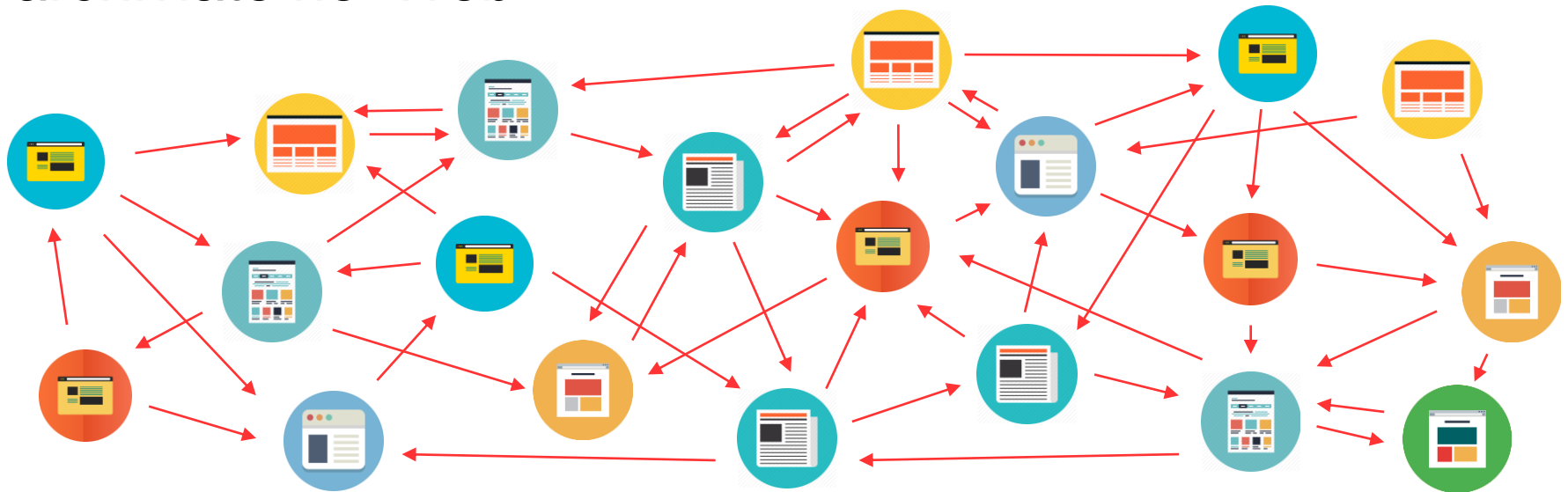
Andrea Del Fante



# Il problema nel Web

Il Web è la più grande, eterogenea e dinamica sorgente di informazione liberamente fruibile da chiunque.

**Problema:** organizzare ed accedere alle informazioni archiviate nel Web.



# Clustering

Processo di selezione e raggruppamento, a partire da una collezione di dati, di elementi omogenei, avendo come base la somiglianza tra gli stessi.

Il Clustering viene utilizzato nel Web per poter organizzare l'**enorme** mole di dati.

In letteratura, gli algoritmi di Clustering si classificano in quattro categorie, in base alle informazioni usate per raggruppare le pagine Web:

- Algoritmi di Clustering basati sul **contenuto testuale**
- Algoritmi di Clustering basati sui **Web log**
- Algoritmi di Clustering basati sulla struttura **HTML**
- Algoritmi di Clustering basati sulla struttura ad **hyperlink**

# Limitazioni

- Il **contenuto testuale** delle pagine Web, pur avendo lo stesso contenuto informativo, potrebbero essere contestualmente differenti.
- I **Web log** potrebbero produrre Cluster differenti di pagine Web, in base ad ogni profilo utente.
- La qualità dei Cluster, generati in base alla struttura **HTML** delle pagine Web, potrebbe abbassarsi se i tag sono differenti ma offrono una visualizzazione simile.
- Se non vi sono sufficienti relazioni tra nodi (**hyperlink**), allora la qualità dei Cluster sarà bassa.

# Obiettivi della tesi

Lo scopo principale di questa tesi è applicare algoritmi di Clustering per raggruppare le pagine di un sito Web.

In dettaglio:

1. Capire se, combinando le informazioni della struttura del sito e quelle testuali delle pagine, vi è un miglioramento della qualità dei Cluster prodotti.
2. Cercare di aumentare la bontà dei Cluster prodotti dando più importanza a pagine vicino alla homepage.

# Metodologia

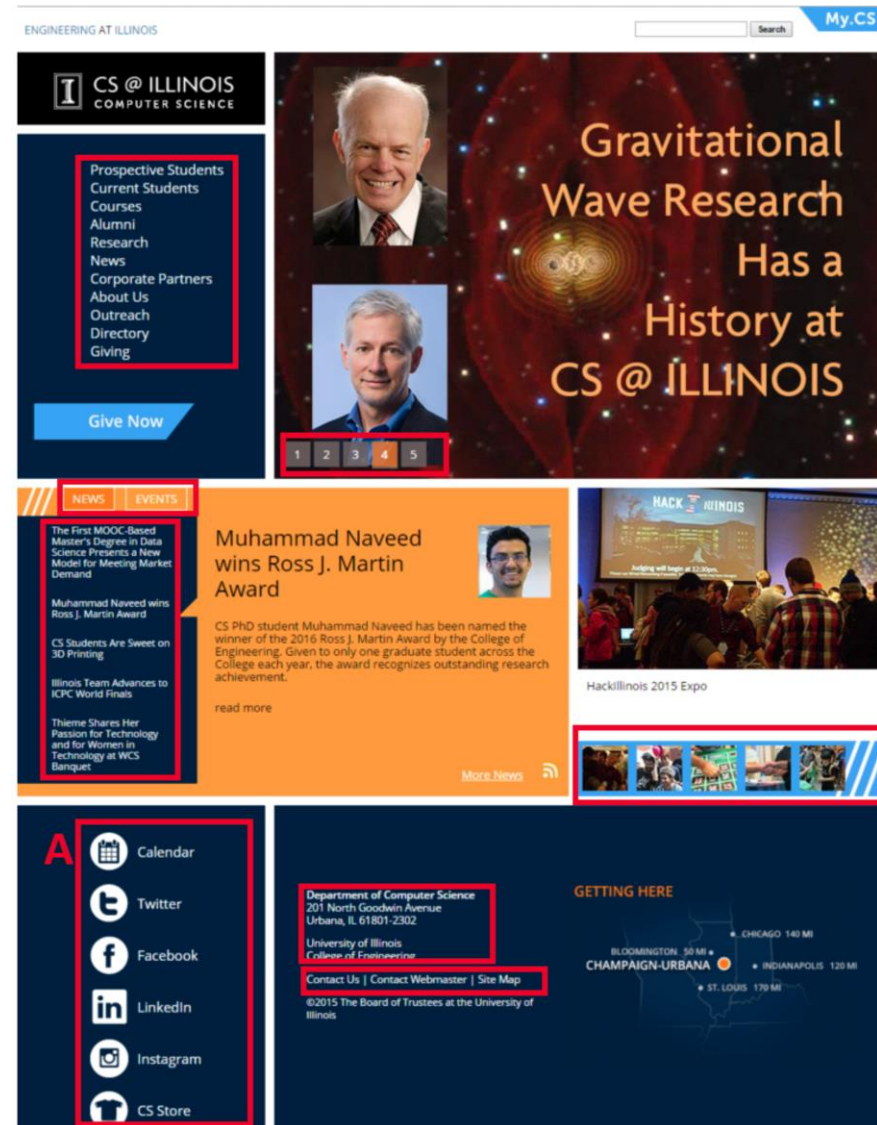
La metodologia definita in questa tesi è caratterizzata da 3 fasi principali:

1. Crawling del sito Web
2. Feature construction
3. Clustering delle pagine Web



# Crawling del sito Web

- Un **Crawler** è un software automatizzato che permette l'indicizzazione delle pagine Web.
- Tecniche di Crawling:
  - Tradizionale
  - Vincolo sulle **Liste Web**
- Dato un grafo Web  $G = (V, E)$ , viene estratto un sottografo  $G' = (V', E')$  avente
 
$$V' \subseteq V, E' \subseteq E$$
- Gli URL vanno **normalizzati** per evitare che il Crawler li analizzi più di una volta. Esempio:  
<http://www.facebook.com/facebook.com/>



# Feature construction

Una volta concluso il processo di Crawling, viene prodotto il grafo del sito Web e il contenuto testuale di ogni singola pagina del sito.

Viene costruito un dataset basato sulle rappresentazioni vettoriali:

- Relative alle **informazioni della struttura**
  - Word2Vec Skip-Gram, applicato sulle sequenze di URL generate attraverso Random Walk standard
  - Word2Vec Skip-Gram modificato, applicato sulle sequenze di URL generate attraverso Random Walk a partenza fissa
  - LINE
- Relative alle **informazioni testuali**
  - Tf-Idf
  - Doc2Vec
- Combine
  - Relative sia alle **informazioni della struttura che testuali**



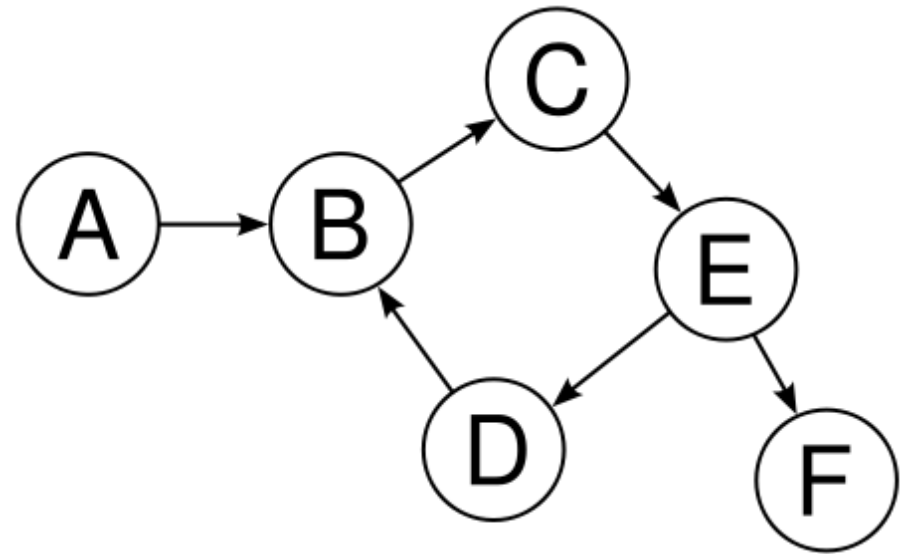
# Rappresentare la struttura: Random Walk

Il Random Walk è una **passeggiata aleatoria**, usata per l'esplorazione del grafo del sito Web.

Vengono generate sequenze di URL, partendo da un nodo del grafo e seguendo ricorsivamente un hyperlink casuale fino ad un limite prefissato.

Due tipologie:

- Random Walk **standard**
- Random Walk **a partenza fissa**



# Rappresentare la struttura: Word2Vec

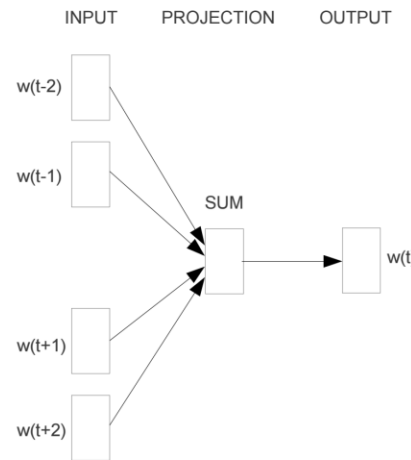
Word2Vec è un algoritmo di **Word Embedding** ed è una rete neurale che apprende le parole da un testo in input, che vengono trasformate in vettori.

$$W : words \rightarrow R$$

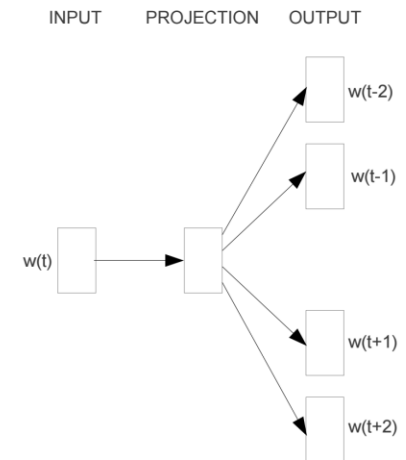
$$W(\text{mat}) = (0.0, 0.6, -0.1, \dots)$$

Due modelli di apprendimento:

- **CBOW**
- **Skip-Gram**



CBOW



Skip-gram

# Rappresentare la struttura: Word2Vec Skip-Gram



Trovare rappresentazioni vettoriali delle parole per predire quelle circostanti in una frase, massimizzando la probabilità media logaritmica.

$$\frac{1}{T} \sum_{t=1}^T \sum_{-c \leq j \leq c, j \neq 0} \log p(w_{t+j} | w_t)$$

Dove:

- $c$  è la dimensione della finestra di contesto
- $w_t$  è la parola in input
- $w_{t+j}$  è la parola in analisi del contesto

Esempio:

The quick brown fox jumped over the lazy dog.

([quick], the), ([the, brown], quick), ([quick, fox], brown), [...] →  
(the, quick), (quick, the), (quick, brown), (brown, quick), [...]

# Rappresentare la struttura: Word2Vec Skip-Gram modificato

Per raggiungere uno degli obiettivi della tesi, si è deciso di modificare Skip-gram, in maniera tale da:

- considerare solo il contesto sinistro, data una parola.
- dare più importanza alle parole più vicine a quella in analisi.

Esempio:

The quick brown fox jumped over the lazy dog.

([], the), ([the], quick), ([quick], brown), [...] →  
(the), (quick, the), (brown, quick), [...]

# Rappresentare la struttura: LINE

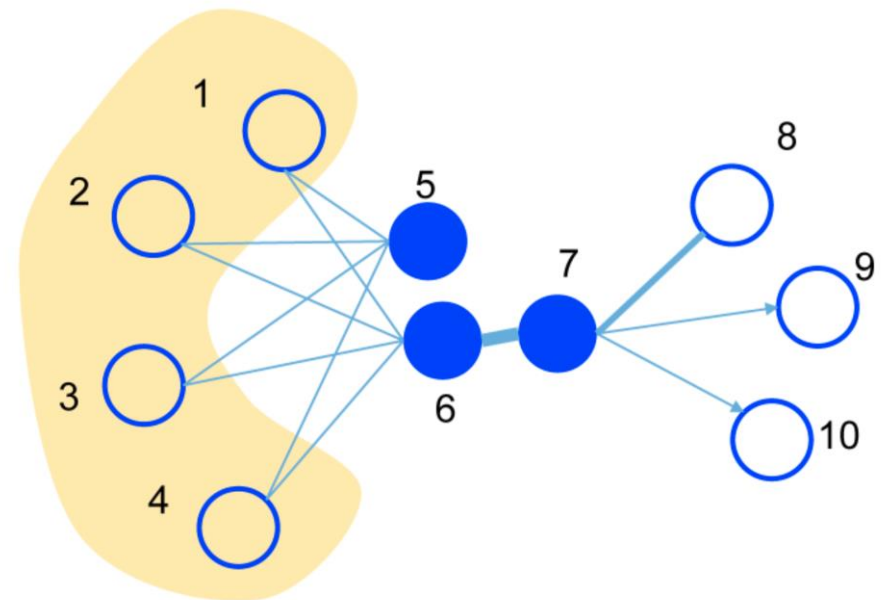
Dato un grafo  $G = (V, E)$ , l'obiettivo è quello di rappresentare ogni vertice  $v \in V$  in un vettore  $R^d$ , dove  $d \ll |V|$ .

- Prossimità di **primo** ordine:

$$- \sum_{(i,j) \in E} w_{ij} \log p_1(v_i, v_j)$$

- Prossimità di **secondo** ordine:

$$- \sum_{(i,j) \in E} w_{ij} \log p_2(v_j | v_i)$$



# Rappresentare il testo: Tf-Idf

Funzione che viene usata per misurare l'importanza di un termine rispetto ad un documento o ad una collezione di documenti.

- **Tf**: misura quante volte un termine appare in un documento.

$$tf_{i,j} = \frac{n_{i,j}}{|d_j|}$$

- **Idf**: misura quante volte un termine appare in tutta la collezione di documenti.

$$idf_i = \log \frac{|D|}{|\{d : t_i \in d\}|}$$

# Rappresentare il testo:

## Doc2Vec



**Estensione** di Word2Vec che aggrega tutte le parole di un paragrafo (documento) in un vettore.

Il processo di apprendimento è caratterizzato dallo spostamento delle parole di contesto attraverso ogni parola di ogni paragrafo, per ogni paragrafo.

L'idea di base è quella di apprendere rappresentazioni vettoriali di paragrafi in modo **simile** all'apprendimento delle rappresentazioni vettoriali delle parole.

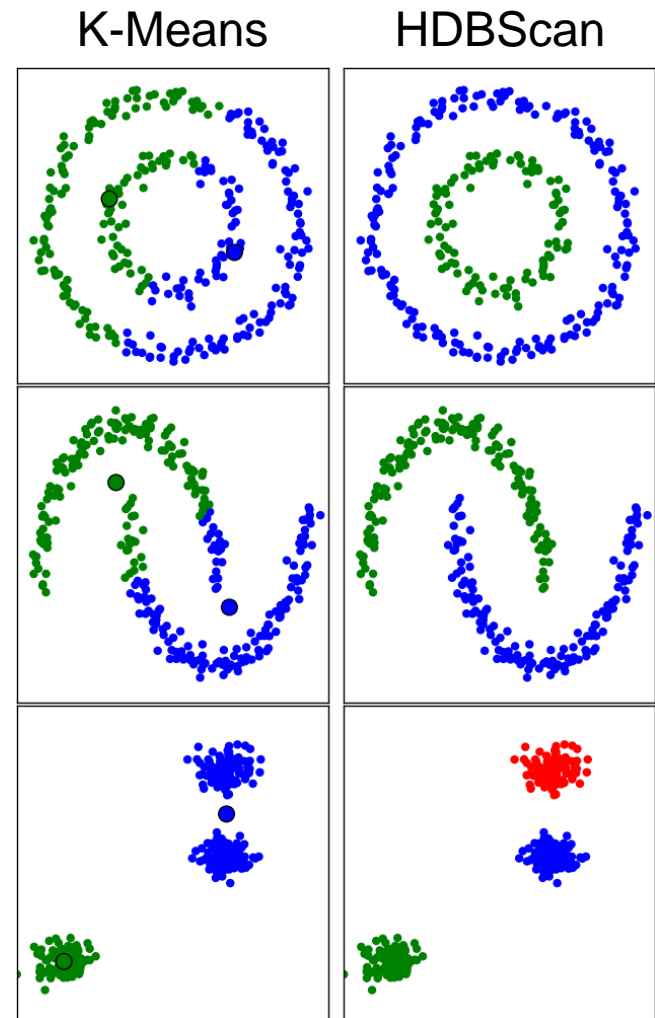
Vengono usati i vettori dei **paragrafi** appresi e quelli delle **parole**. I modelli di apprendimento sono gli stessi di Word2Vec.

# Clustering delle pagine Web

Sono stati usati algoritmi di Clustering che sfruttano le rappresentazioni vettoriali per raggruppare elementi:

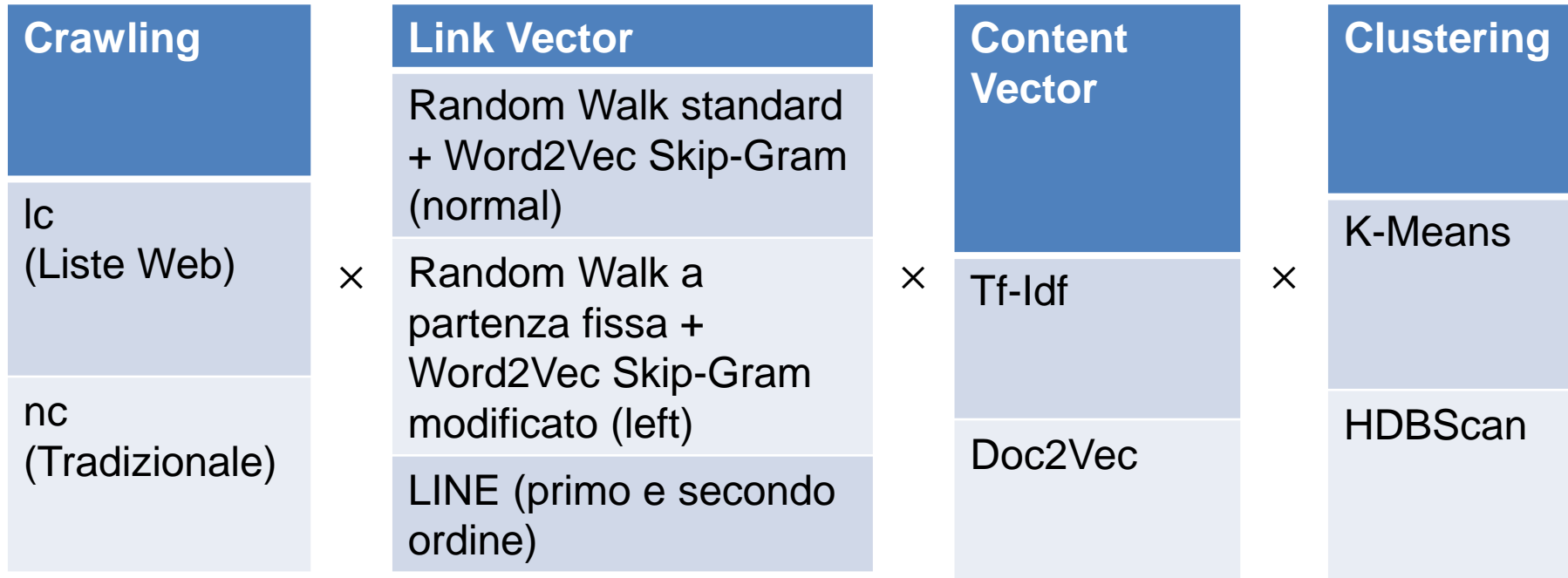
- K-Means (n\_clusters)
- HDBScan (min\_cluster\_size)

I vettori, prima di essere usati nel processo di Clustering, sono stati normalizzati usando L2.





# Impostazione sperimentale



Per il tuning dei parametri:

$$Clustering \times \{Crawling \times [Link Vector + (Link Vector + Content Vector)] + Content Vector\} = 316 \text{ combinazioni}$$

Si riportano le configurazioni con i valori migliori delle seguenti misure: omogeneità (**Hom**), completezza (**Com**), v-measure (**V-M**), adjusted mutual information (**AMI**), adjusted random index (**ARI**), silhouette (**Silh**).

# Sperimentazione

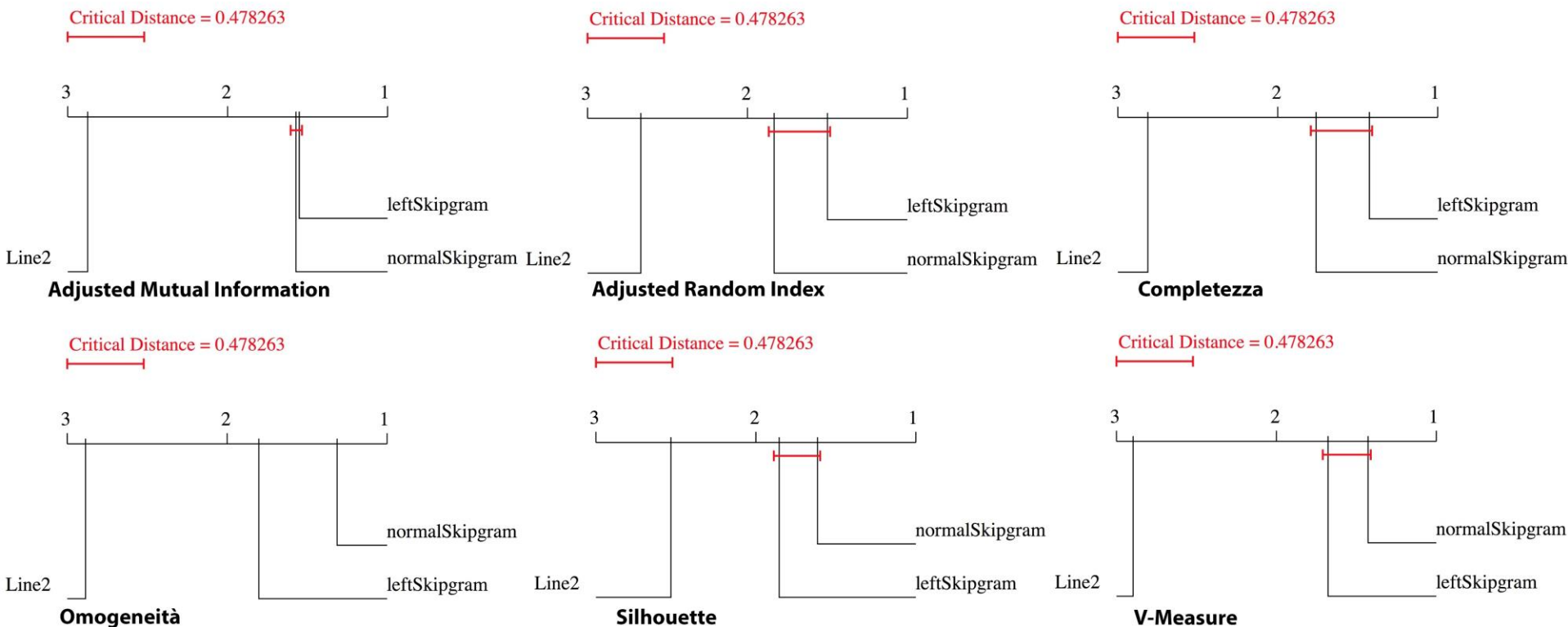


Sito	# pagine	# archi	# archi con Liste Web	# Cluster
Illinois	563	9415	5330	10
Oxford	3480	44526	35148	19
Priceton	3132	122493	104585	16
Stanford	167	30087	12372	10

# Sperimentazione

Configurazione	AMI	ARI	Com	Hom	Silh	V-M
Combinato vs Struttura	<b>0.0031</b> -	0.4105 -	<b>0.0117</b> -	<b>0.0153</b> -	0.4759 +	<b>0.0115</b> -
Combinato vs Testo	<b>0.0004</b> -	<b>0.0050</b> -	<b>0.0000</b> -	<b>0.0000</b> -	<b>0.0000</b> -	<b>0.0000</b> -
Struttura vs Testo	0.1443 +	<b>0.0389</b> -	<b>0.0065</b> -	<b>0.0019</b> -	<b>0.0000</b> -	<b>0.0067</b> -
LC vs NC	<b>0.0344</b> -	0.0511 -	<b>0.0034</b> +	0.4154 -	0.1467 +	0.0586 +
KMeans vs HDBScan	<b>0.0000</b> -	<b>0.0000</b> -	0.2322 -	<b>0.0000</b> -	<b>0.0000</b> -	<b>0.0000</b> -
TF-IDF vs Doc2Vec	0.2875 -	0.2643 -	0.2641 -	0.0707 -	<b>0.0150</b> -	0.1811 -

# Sperimentazione



# Conclusioni e sviluppi futuri

## Conclusioni:

- Il testo delle pagine e la struttura del sito Web forniscono informazioni diverse e complementari che possono migliorare la qualità dei Cluster.
- L'utilizzo delle liste Web non ha prodotto miglioramenti significativi.
- L'utilizzo di Skip-Gram modificato non ha prodotto miglioramenti significativi.

## Sviluppi futuri:

- Estendere la metodologia su più siti Web e meno strutturati.

# Grazie per l'attenzione