Url2Vec:
Clustering di pagine in un grafo Web

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

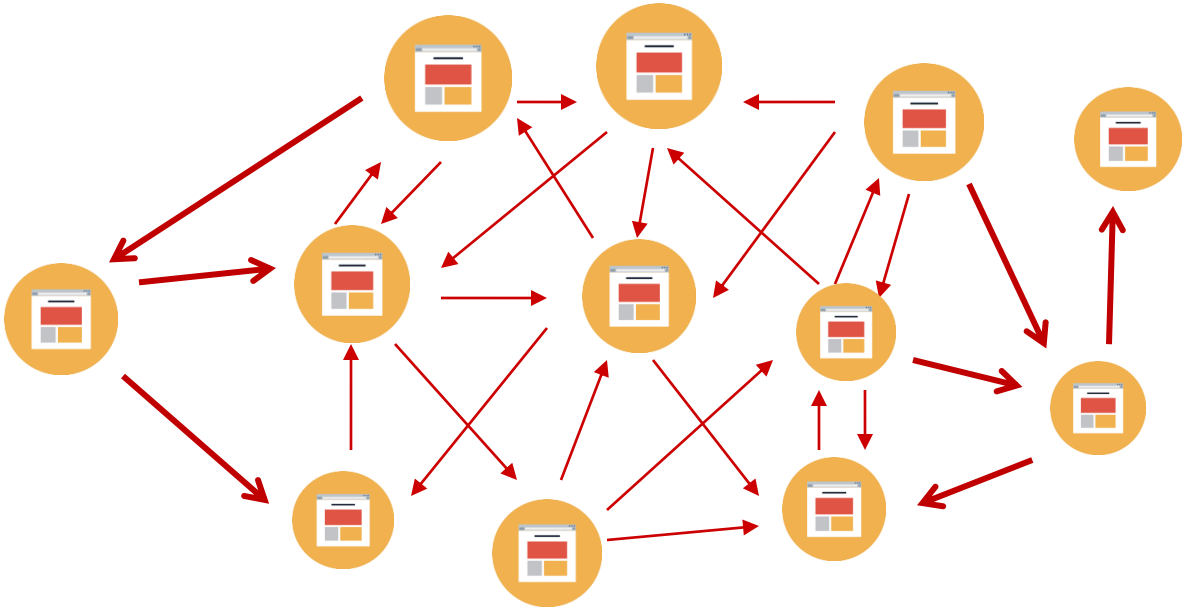
Relatore: Prof. Michelangelo *Ceci*

Correlatore: Dott.ssa *Pasqua Fabiana Lanotte*

Laureando: *Christopher Piemonte*

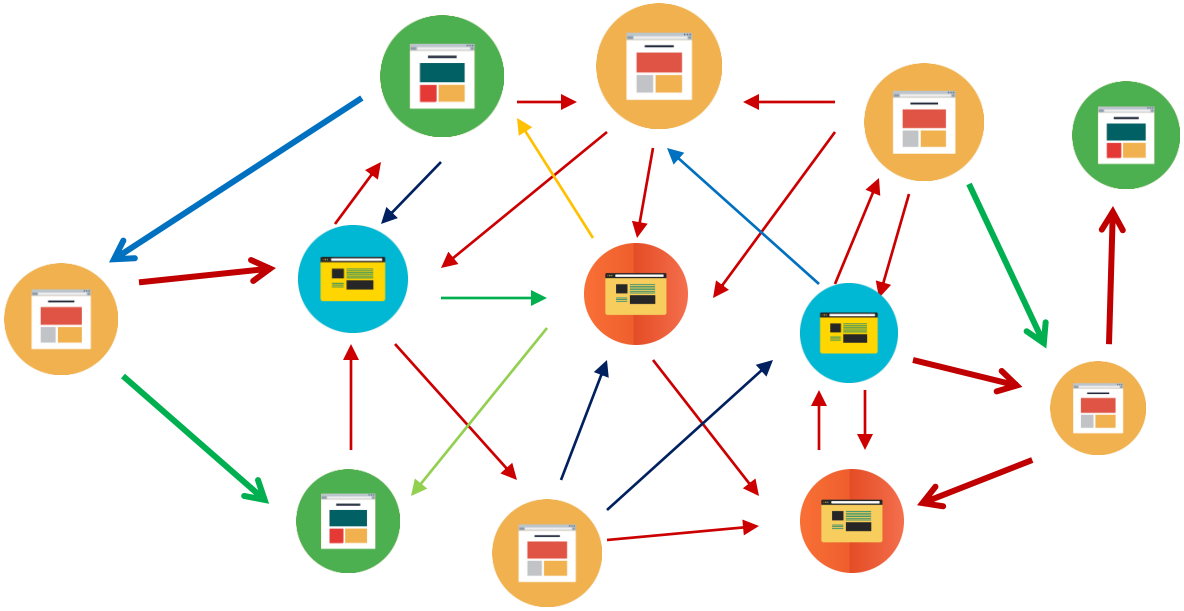
Il Web: da una rete omogenea . . .

Molti strumenti di Web Mining si basano sull'assunzione che il Web contiene pagine ed hyperlink dello stesso tipo formando una rete informativa omogenea.



. . . ad una rete eterogenea

In realtà il Web contiene diversi tipi di oggetti che interagiscono tra loro attraverso vari tipi di relazioni, propagate attraverso **dati strutturati** formando una rete informativa eterogenea.

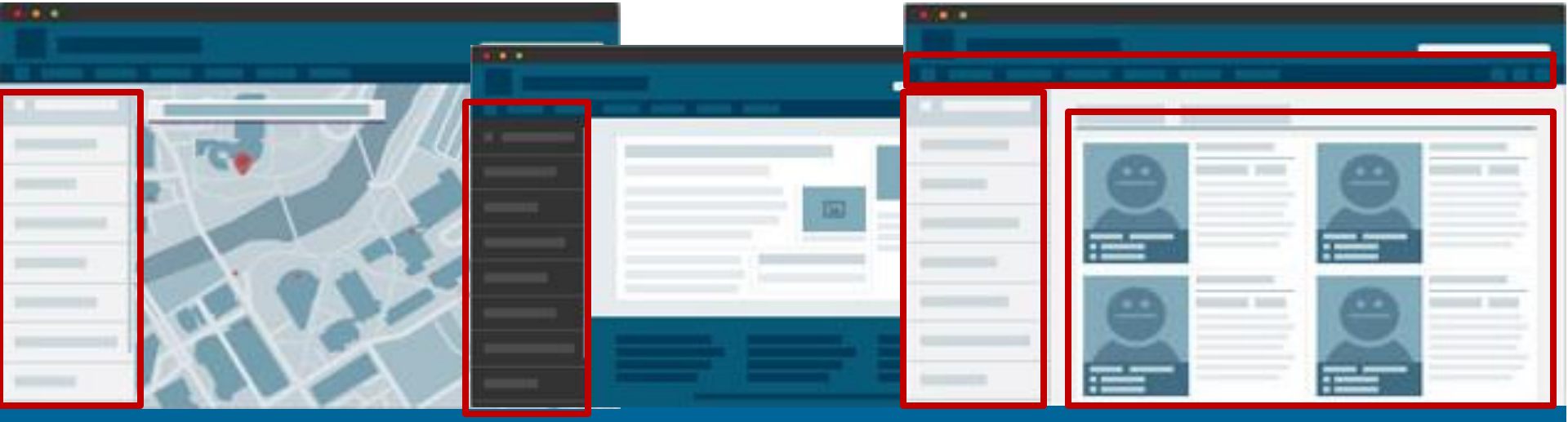


Dati strutturati nel Web

- Collezione di elementi web semanticamente simili, organizzati in collezioni aventi una struttura ed una presentazione uniforme (liste web);
- Grandi quantità di dati strutturati nel Web esistono in varie forme: liste *HTML, tabelle HTML* e Deep Web database.

Dati strutturati nel Web

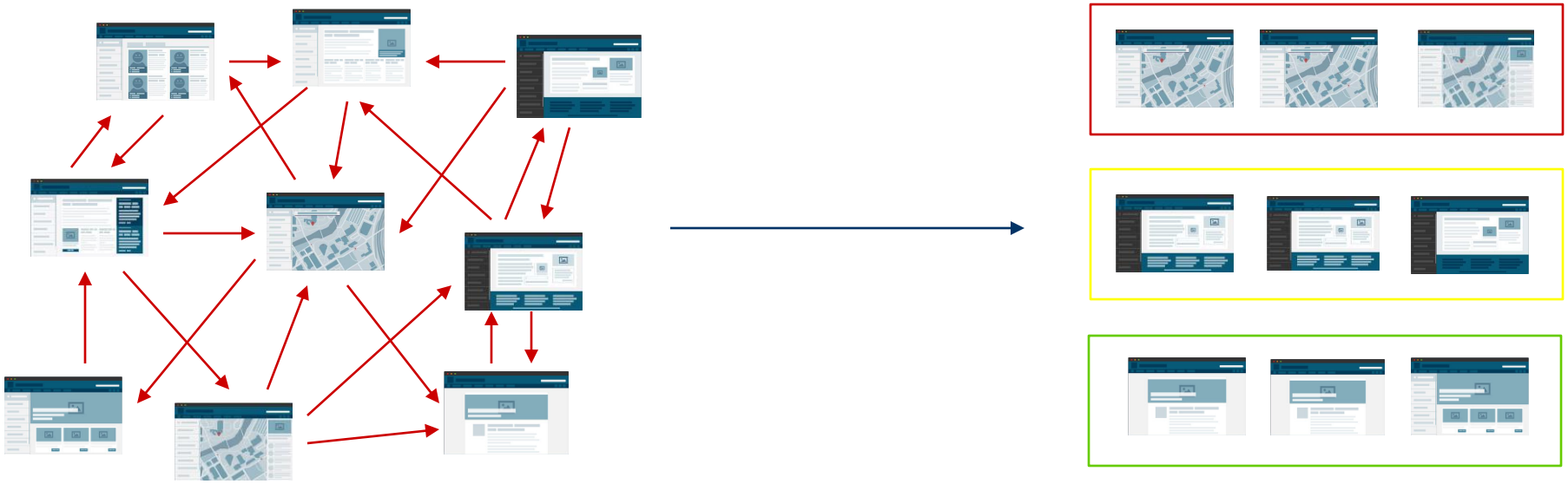
- Collezione di elementi web semanticamente simili, organizzati in collezioni aventi una struttura ed una presentazione uniforme (liste web);
- Grandi quantità di dati strutturati nel Web esistono in varie forme: liste *HTML, tabelle HTML* e Deep Web database.



Obiettivo

Raggruppare pagine web in cluster utilizzando dati strutturati (liste web, grafo web)

- Reperimento informazione
- Individuare pagine web dello stesso tipo semantico



Stato dell'arte

Il clustering delle pagine web può avvenire considerando diversi fattori:

- Il contenuto testuale *(Text Mining)*
- Web log *(Web Usage Mining)*
- Hyperlink *(Graph Theory)*
- Codice HTML *(Web Structure Mining*)

Limitazioni

- **Text Mining:** Assunzioni di indipendenza
- **Web Usage Mining:** Il clustering dipende dalla tipologia di utenti
- **Graph Theory:** Gli algoritmi sono computazionalmente costosi e considerano solo le relazioni tra i nodi
- **Web Structure Mining**: La qualità dei risultati dipende dalla struttura HTML delle pagine web

Soluzione

Creare un algoritmo capace di:

- Combinare informazioni testuali e informazioni strutturate del grafo web
- Rappresentare le informazioni estratte in uno spazio vettoriale

[0.4, 1.2, 3.1, . . .]

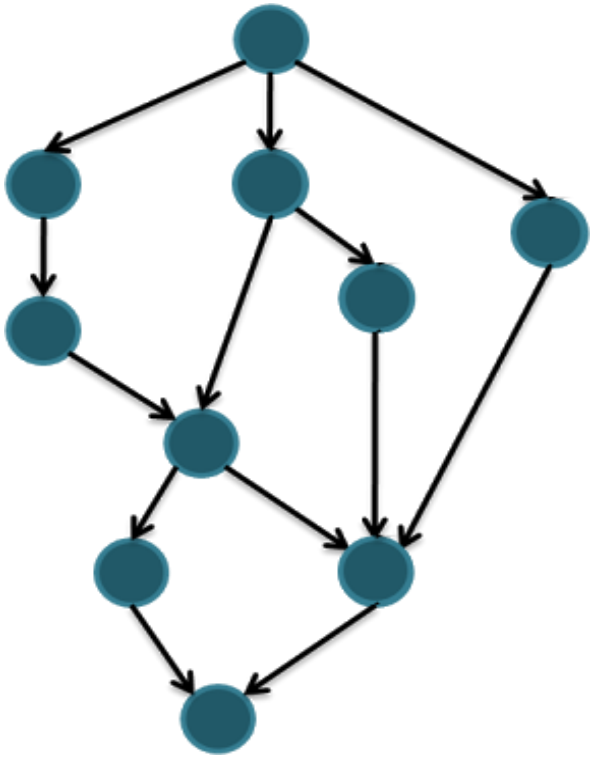
Metodologia

L'algoritmo è caratterizzato da tre fasi principali:

1. **Web Graph discovery**
2. **URL embedding**
   - Rappresentazione vettoriale del contenuto testuale
   - Rappresentazione vettoriale della struttura del grafo
3. **Web page clustering**

1. Web Graph Discovery

- Data l'homepage di un sito web, estrarre il grafo web
- Tecniche di crawling:
  - Senza vincolo sui dati strutturati
  - Con vincolo sui dati strutturati



Crawling con vincolo sui dati strutturati

- Utilizzo delle liste web per ridurre la dimensione del grafo *G (V, E)*
- A partire da *G* è estratto un sottografo *G' (V', E')* con

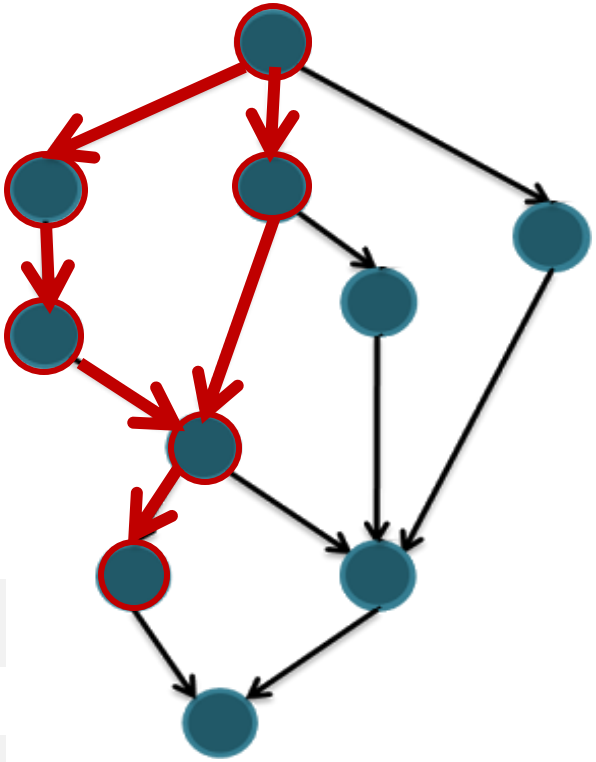
$$V' \subseteq V,\ E' \subseteq E$$

- *G'* è generato attraverso:
  I. Estrazione delle liste web

$$\forall\, a_i \in V, L_p = \{a_1, \dots, a_n \mid a_j \in webLists(a_i)\}$$

  II. Selezione di un url dalle liste estratte

$$\exists (a_i, a_j) \in E' \Leftrightarrow a_j \in L_{a_i}$$



Problemi affrontati per il crawling

- BFS con max depth
- Etica di crawling (tempi di attesa richieste http, robot.txt)
- Contenuto testuale (contenuto visibile, non visibile, etc.)
- Estrazione delle proprietà visuali (attraverso rendering)
- Normalizzazione URL (es. redirect, url differenti associati alla stessa pagina HTML, url relativi, etc.)
- Meccanismi di caching (e.g. Redis, MapDB, etc.)

2. Web Page Embedding

Apprendere una funzione che associa ad ogni URL un vettore multidimensionale

$$W : url \rightarrow R^n$$

$$W(\text{“http://www.uniba.it/”}) = (0.2,\ -0.4,\ 0.7,\ \ldots)$$

2. Web Page Embedding

Attraverso informazioni testuali:

2.1 TF-IDF

Attraverso informazioni strutturate:

2.2 Generazione di sequenze di URL attraverso Random Walk

2.3 Applicazione di algoritmi di Word Embedding (e.g. *Word2Vec*)

$$W : R^m \times R^n \rightarrow R^k, \quad k = m + n$$

2.1 TF-IDF

- **TF**: Contando la frequenza di occorrenza dei termini all'interno del documento costruendo la matrice *documenti-termini*:

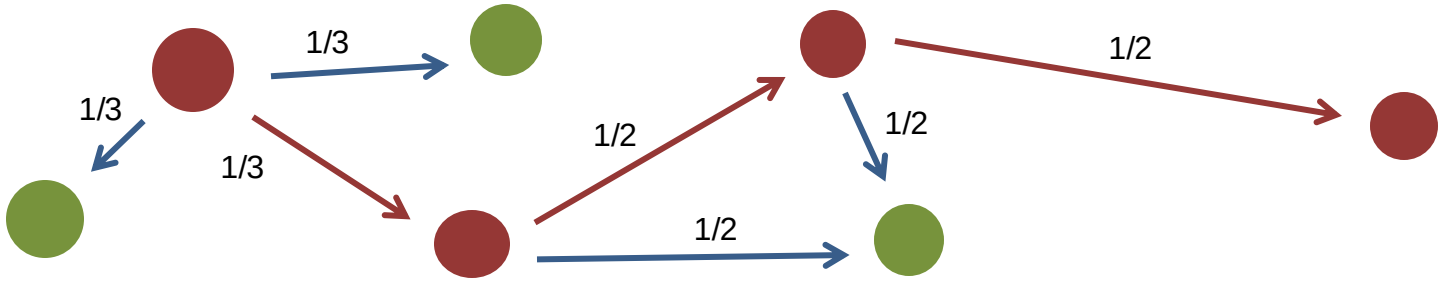
$$\mathrm{tf}_{i,j} = \frac{n_{i,j}}{|d_j|}$$

- **IDF**: Pesare l'importanza dei termini che compaiono nel documento, ma che in generale sono poco frequenti

$$\mathrm{idf}_i = \log \frac{|D|}{|\{d : t_i \in d\}|}$$

2.2 Generazione delle sequenze

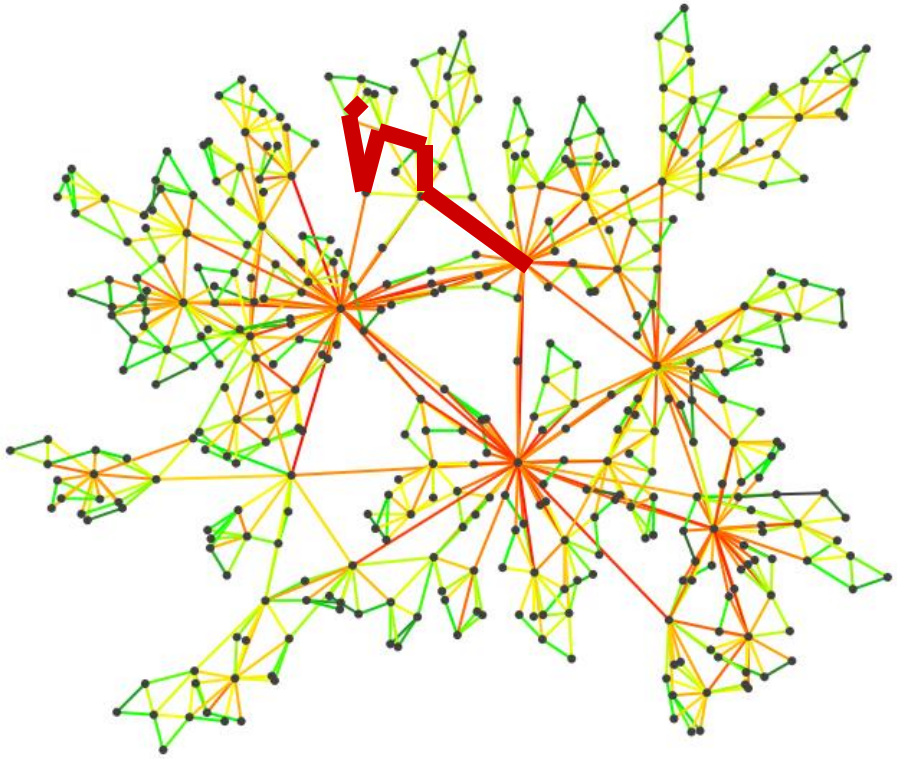
Vengono generate sequenze di URL partendo da un nodo nel grafo e seguendo ricorsivamente un hyperlink casuale fino ad un limite prefissato.



http://home.com/about -- http://home.com/about/awards -- http://home.com/ . . .

http://home.com -- http://home.com/courses -- http://home.com/courses/ml . . .

2.2 Generazione delle sequenze



2.1 Word2Vec

- Utilizza sequenze di parole per apprendere una rete neurale

$$\arg\max_{\theta} \prod_{(w,c)\in D} p(c|w;\theta)$$

- Il modello appreso è applicato ad ogni parola in input

2.1 Word2Vec

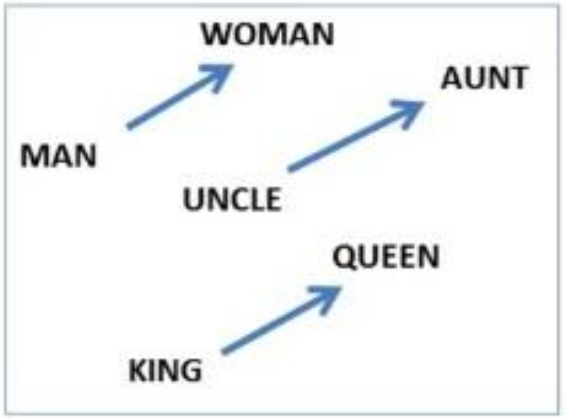
- I vettori appresi codificano regolarità linguistiche nella differenza tra i vettori

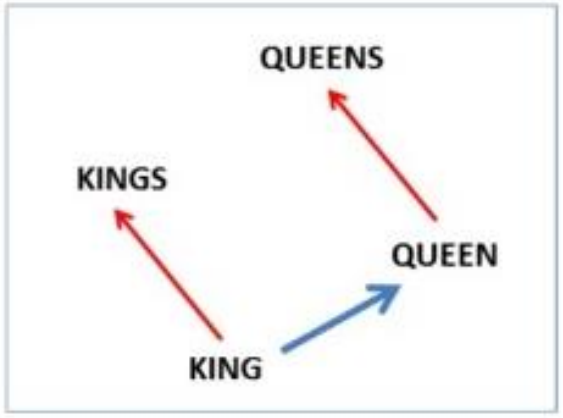
man is to **woman** as **king** is to **?**

2.1 Word2Vec

- I vettori appresi codificano regolarità linguistiche nella differenza tra i vettori

man is to **woman** as **king** is to **queen**





3. Clustering

Def: raggruppa una collezione di dati secondo una data funzione di similarità, in modo che oggetti dello stesso gruppo siano più simili tra di loro che ad oggetti in altri gruppi.

Sono stati utilizzati i seguenti algoritmi:

- K-means (`n_clusters`)
- DBSCAN (`eps, min_samples`)
- HDBSCAN (`min_cluster_size`)

Sperimentazione

Obiettivo: Verificare un eventuale miglioramento delle performance degli algoritmi di clustering, attraverso la combinazione di informazioni testuali e strutturate

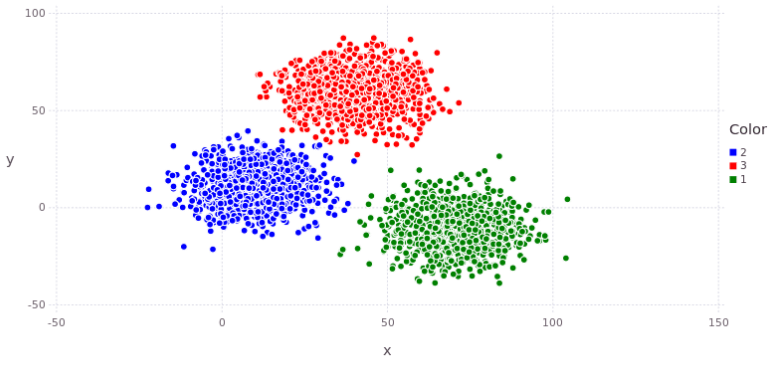
Dataset	n. pagine	n. hyperlink
cs.illinois.edu	728	16993
cs.stanford.edu	1458	99686
eecs.mit.edu	1745	63937
cs.princeton.edu	16378	206985
cs.ox.ac.uk	4183	27954

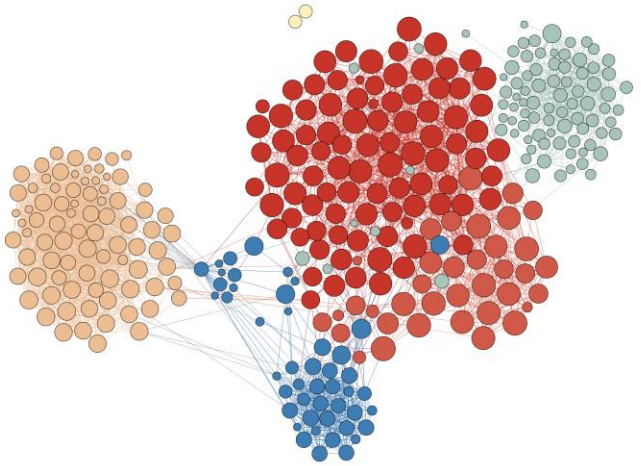
Sperimentazione

Sono state confrontate le performance di algoritmi basati su:

Rappresentazioni vettoriali
- K-Means
- DBSCAN
- HDBSCAN

Analisi del grafo
- WalkTrap (WT)
- Fastgreedy (FG)





Metriche

Omogeneità
- I cluster restituiti contengono solo vettori di una classe

Completezza
- Tutti i membri di una classe sono assegnati ad un cluster

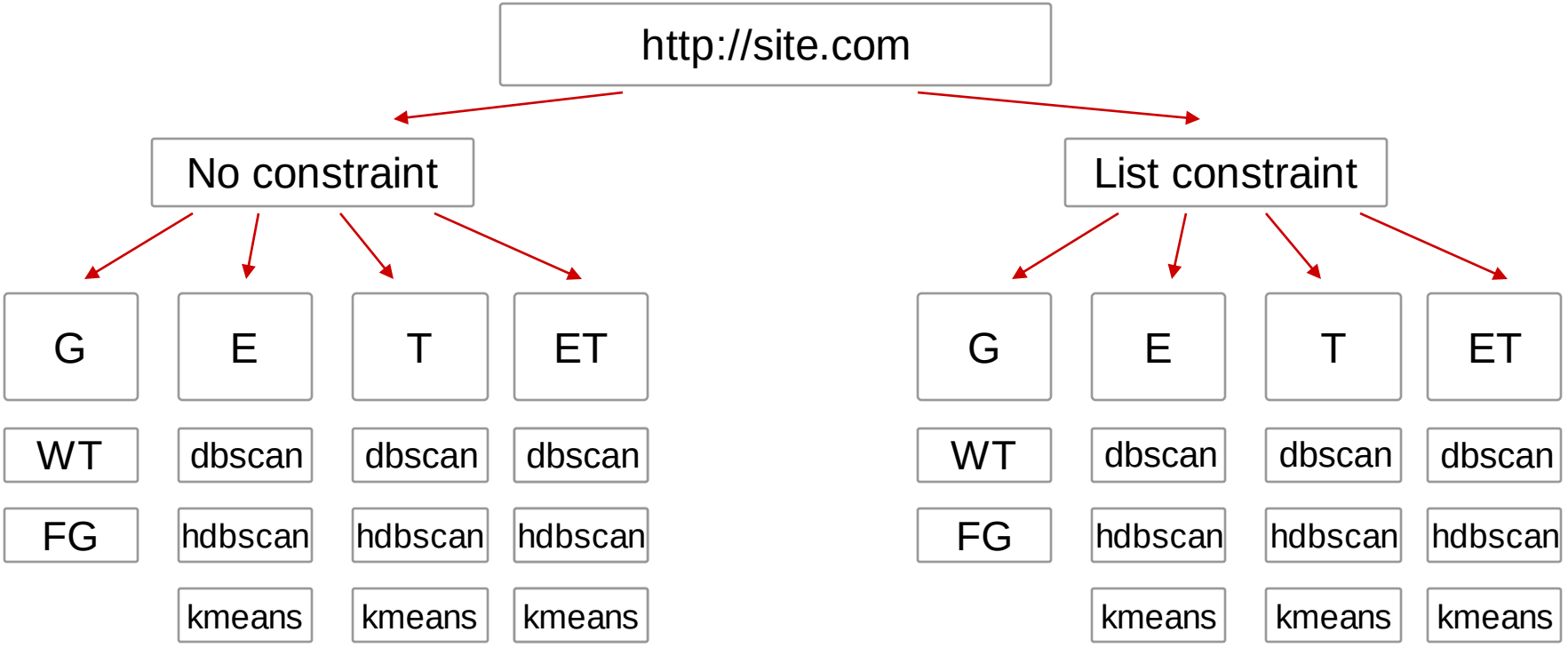
V-Measure
- Media armonica tra *omogeneità* e *completezza*

Adjusted Rand Index
- Percentuale di coppie concordanti nei due assegnamenti

Mutual Information
- Intesa come distanza tra i due assegnamenti

Silhouette
- Quanto sono definiti i cluster trovati

Sperimentazione



Illinois	Hom	Com	V-M	ARI	MI	Silh
G-nc WT	0.6471	0.6585	0.6527	0.4363	0.6281	//
G-nc FG	0.5518	**0.8563**	0.6711	0.5764	0.5354	//
G-lc WT	0.5093	0.4892	0.4991	0.2762	0.4722	//
G-lc FG	0.5522	0.6035	0.5767	0.3656	0.5382	//
E-nc dbscan	0.5553	0.6579	0.6023	0.4487	0.5234	0.2588
E-nc hdbscan	0.5759	0.6720	0.6203	0.5282	0.5525	0.2573
E-nc Kmeans	0.8238	0.7575	0.7892	0.7883	0.7423	0.3131
E-lc dbscan	0.4163	0.5922	0.4889	0.2250	0.3935	0.1320
E-lc hdbscan	0.4760	0.5067	0.4908	0.2275	0.4515	0.1054
E-lc Kmeans	0.8095	0.6593	0.7267	0.6189	0.6473	0.2281
T dbscan	0.5601	0.5962	0.5776	0.4078	0.5346	0.1242
T hdbscan	0.5152	0.6029	0.5556	0.3862	0.4858	0.0881
T Kmeans	0.7619	0.5814	0.6596	0.3184	0.5586	0.1767
ET-nc hdbscan	0.7327	0.7534	0.7429	0.7204	0.7186	0.2070
ET-nc Kmeans	**0.8812**	0.8069	**0.8424**	**0.8299**	**0.7949**	**0.3198**
ET-lc hdbscan	0.6541	0.6129	0.6328	0.3249	0.5992	0.1203
ET-lc Kmeans	0.8548	0.6885	0.7627	0.6488	0.6773	0.2573

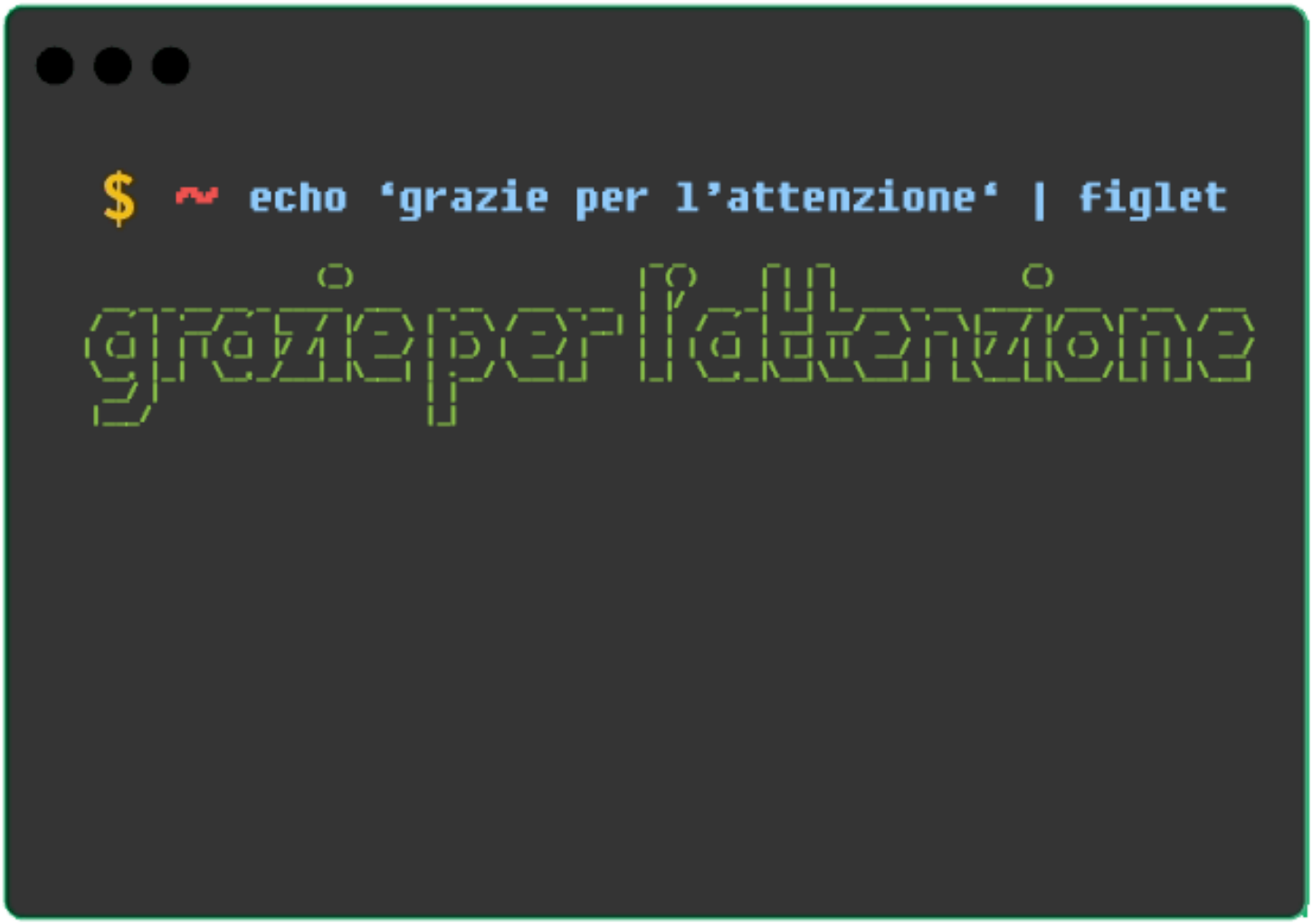
Conclusioni e Sviluppi Futuri

Conclusioni:

- Si sono riscontrati miglioramenti significativi unendo le informazioni testuali con le informazioni strutturate
- L'utilizzo delle liste non ha contribuito a migliorare le performance

Sviluppi futuri:

- Identificare la metodologia appropriata in base al contesto
- Utilizzare altri algoritmi di embedding (GloVe)
- Estendere l'analisi a più siti web


$ ~ echo 'grazie per l'attenzione' | figlet
grazieperl'attenzione