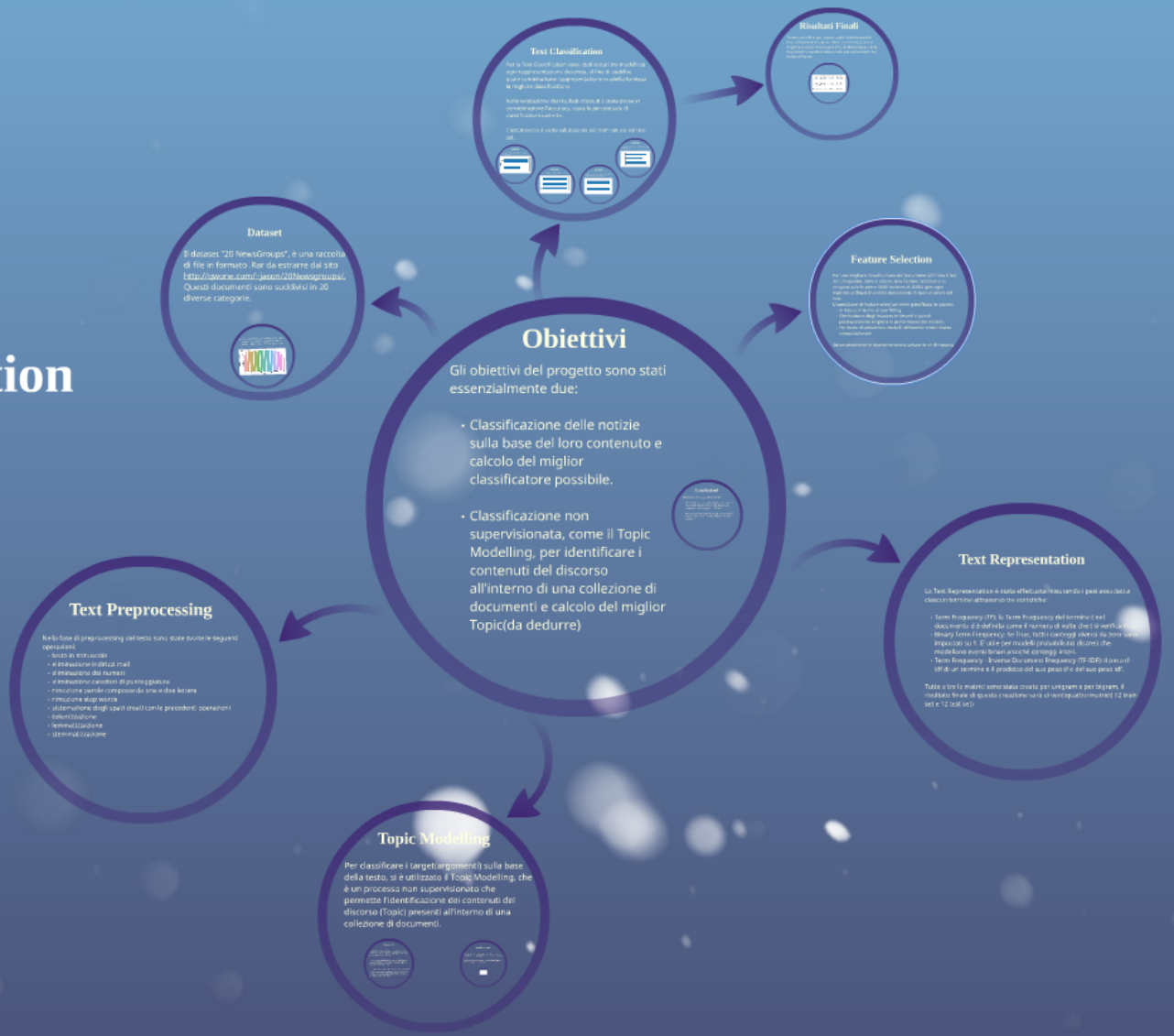
20 NewsGroups Classification

Corrado Montoro - 841489

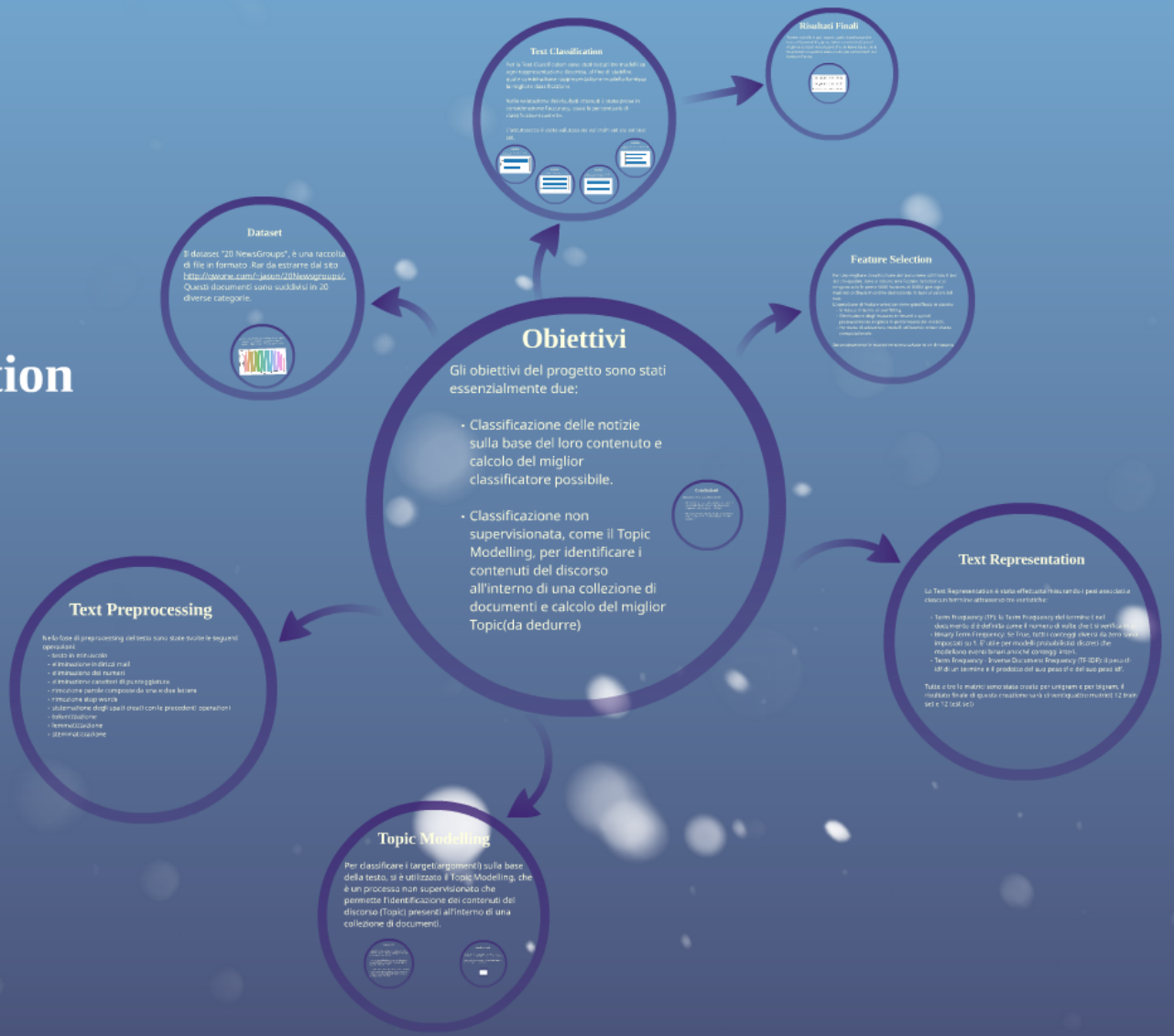
Luca Lazzati - 850334



20 NewsGroups Classification

Corrado Montoro - 841489

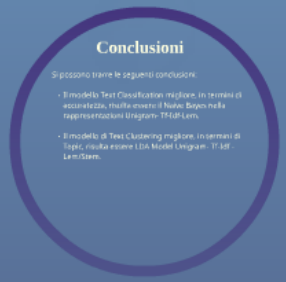
Luca Lazzati - 850334



Obiettivi

Gli obiettivi del progetto sono stati essenzialmente due:

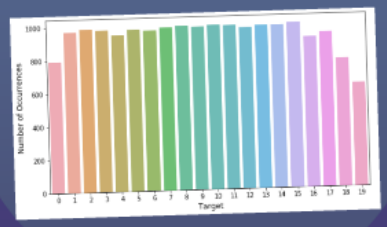
- Classificazione delle notizie sulla base del loro contenuto e calcolo del miglior classificatore possibile.
- Classificazione non supervisionata, come il Topic Modelling, per identificare i contenuti del discorso all'interno di una collezione di documenti e calcolo del miglior Topic(da dedurre)



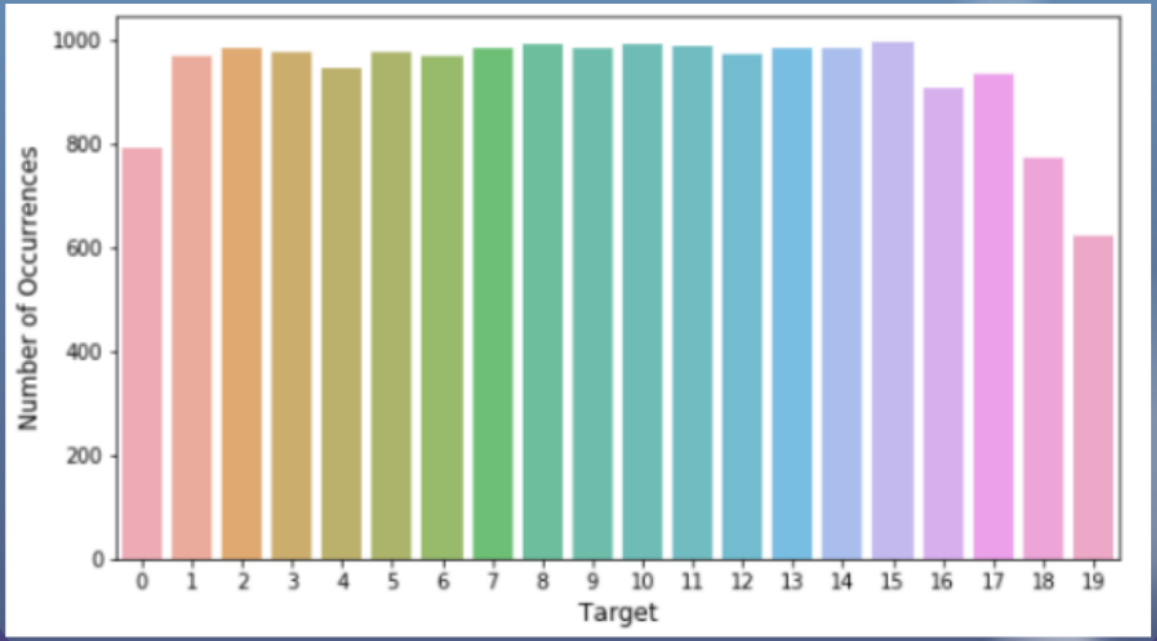
Dataset

Il dataset “20 NewsGroups", è una raccolta di file in formato .Rar da estrarre dal sito http://qwone.com/~jason/20Newsgroups/. Questi documenti sono suddivisi in 20 diverse categorie.

Ogni record presente è in realtà un file di testo in inglese, che presenta la seguente struttura: metadati - intestazione - testo del documento.



Ogni record presente è in realtà un file di testo in inglese, che presenta la seguente struttura: metadati - intestazione - testo del documento.



Text Preprocessing

Nella fase di preprocessing del testo sono state svolte le seguenti operazioni:

- testo in minuscolo
- eliminazione indirizzi mail
- eliminazione dei numeri
- eliminazione caratteri di punteggiatura
- rimozione parole composte da una e due lettere
- rimozione stop words
- sistemazione degli spazi creati con le precedenti operazioni
- tokenizzazione
- lemmatizzazione
- stemmatizzazione

Text Representation

La Text Representation è stata effettuata misurando i pesi associati a ciascun termine attraverso tre euristiche:

- Term Frequency (TF): la Term Frequency del termine t nel documento d è definita come il numero di volte che t si verifica in d.
- Binary Term Frequency: Se True, tutti i conteggi diversi da zero sono impostati su 1. E' utile per modelli probabilistici discreti che modellano eventi binari anziché conteggi interi.
- Term Frequency - Inverse Document Frequency (TF-IDF): il peso tf-idf di un termine e il prodotto del suo peso tf e del suo peso idf.

Tutte e tre le matrici sono stata create per unigram e per bigram, il risultato finale di questa creazione sarà di ventiquattro matrici(12 train set e 12 test set)

Feature Selection

Per una migliore classificazione del testo viene utilizzato il test del chi-quadro, dove si ottiene una Feature Selection e si tengono solo le prime 5000 features di 20000 (per ogni matrice) ordinate in ordine decrescente in base al valore del test.

L'operazione di feature selection viene giustificata in quanto:

- Si riduce il rischio di overfitting.
- Eliminazione degli features irrilevanti e quindi potenzialmente migliora le performance dei modelli.
- Permette di addestrare modelli utilizzando minor sforzo computazionale.

Successivamente le matrici verranno salvate in un dizionario

Text Classification

Per la Text Classification sono stati testati tre modelli su ogni rappresentazione descritta, al fine di stabilire quale combinazione rappresentazione-modello fornisca la migliore classificazione.

Nella valutazione dei risultati ottenuti è stata presa in considerazione l'accuracy, ossia la percentuale di classificazioni corrette.

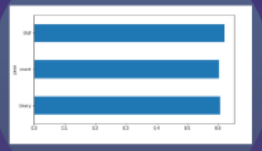
L'accuratezza è stata valutata sia sul train set sia sul test set.

Risultati

Il tipo di rappresentazione mediamente più performante:

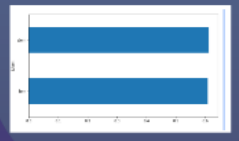
Risultati

Il peso mediamente più performante è il tf-idf:



Risultati

Mediamente l'operazione di stemming è il più performante(anche se di poco):

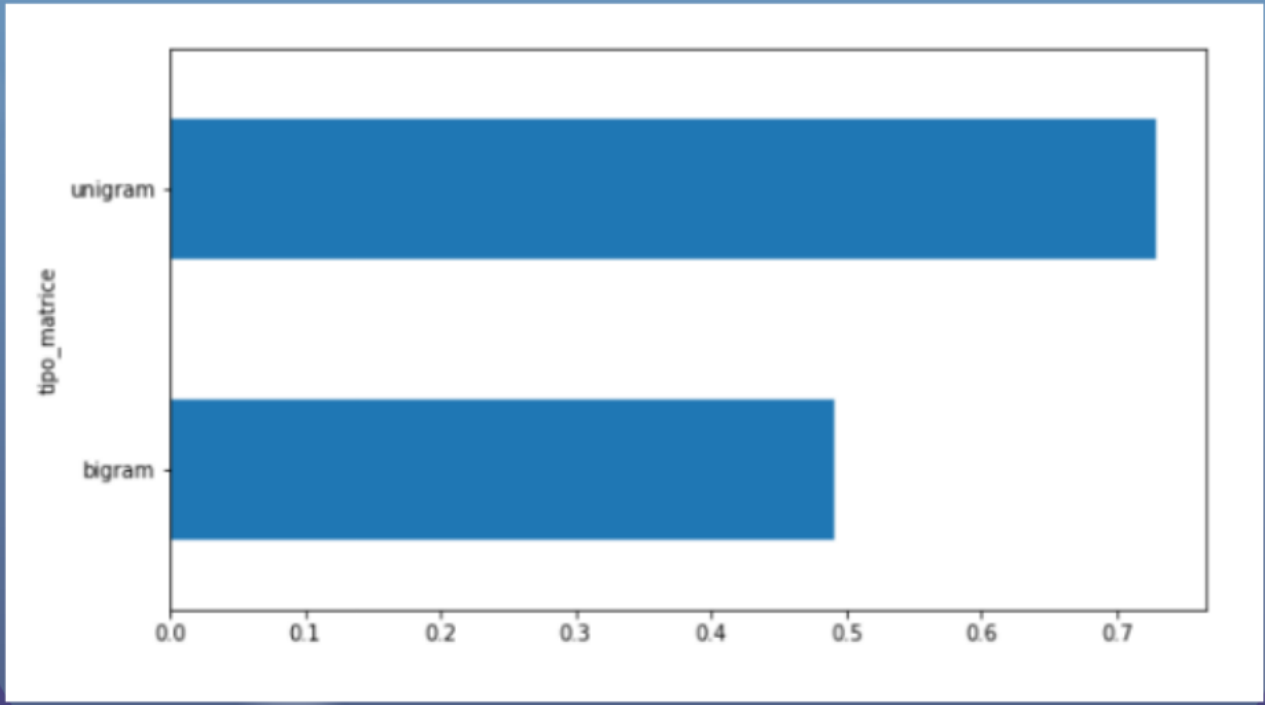


Risultati

Il classificatore mediamente più performante è il Naïve Bayes:

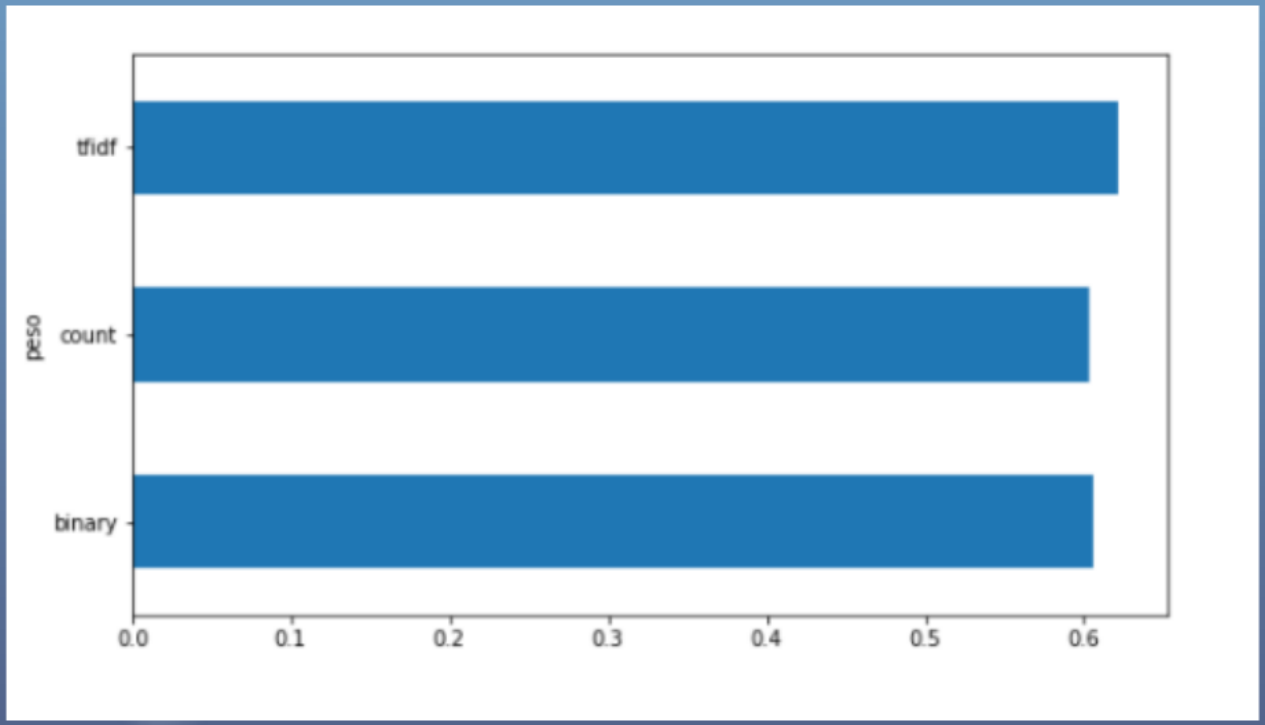
Risultati

Il tipo di rappresentazione mediamente più performante:



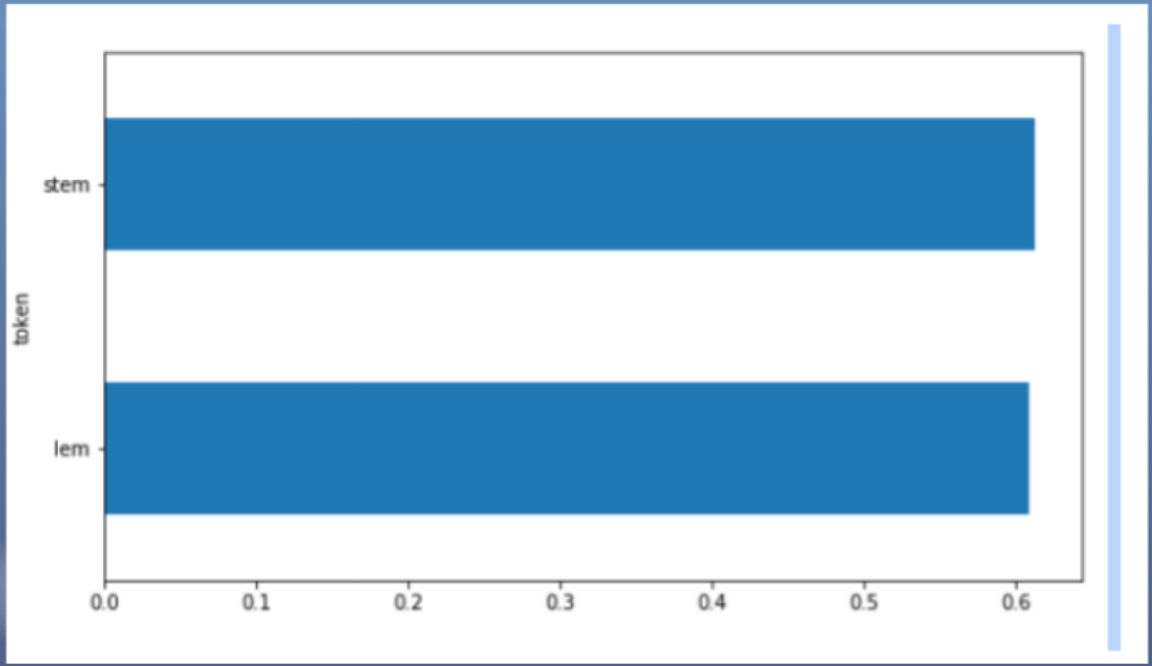
Risultati

Il peso mediamente più performante è il tf-idf:



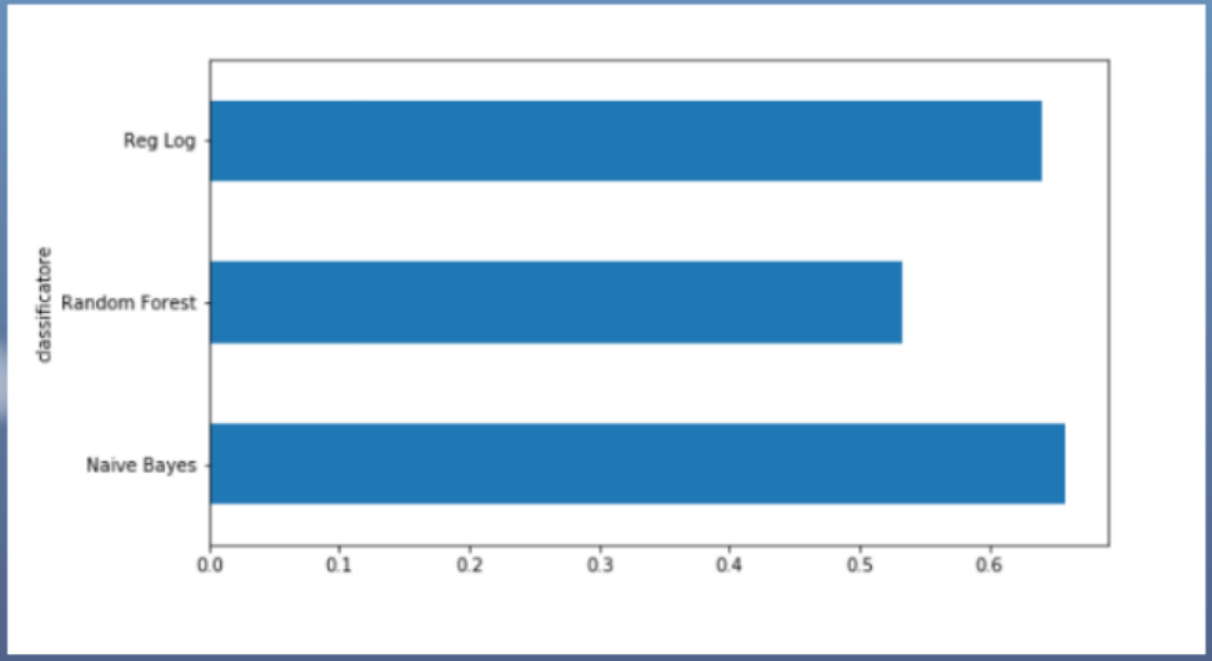
Risultati

Mediamente l'operazione di stemming è il più performante(anche se di poco):



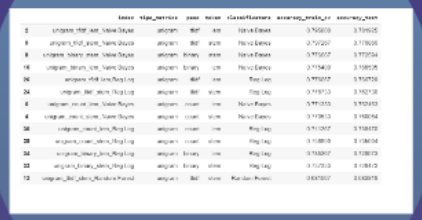
Risultati

Il classificatore mediamente più performante è il Naive Bayes:



Risultati Finali

Tramite tabella si può notare quale classificatore(in base al tipo matrice, peso, token e accuratezza) sia il migliore e si può visualizzare che sia Naive Bayes sia la Regressione Logistica siano molto più performanti del Random Forest.



	index	tipo_matrice	peso	token	classificatore	accuracy_train_cv	accuracy_test
2	unigram_tfidf_lem_Naive Bayes	unigram	tfidf	lem	Naive Bayes	0.795800	0.781925
0	unigram_tfidf_stem_Naive Bayes	unigram	tfidf	stem	Naive Bayes	0.797267	0.776860
8	unigram_binary_stem_Naive Bayes	unigram	binary	stem	Naive Bayes	0.776667	0.772594
10	unigram_binary_lem_Naive Bayes	unigram	binary	lem	Naive Bayes	0.775400	0.768595
26	unigram_tfidf_lem,Reg Log	unigram	tfidf	lem	Reg Log	0.776067	0.766729
24	unigram_tfidf_stem_Reg Log	unigram	tfidf	stem	Reg Log	0.778733	0.762730
6	unigram_count_lem_Naive Bayes	unigram	count	lem	Naive Bayes	0.771333	0.762463
4	unigram_count_stem_Naive Bayes	unigram	count	stem	Naive Bayes	0.770533	0.760064
30	unigram_count_lem_Reg Log	unigram	count	lem	Reg Log	0.741267	0.738470
28	unigram_count_stem_Reg Log	unigram	count	stem	Reg Log	0.736800	0.735004
34	unigram_binary_lem_Reg Log	unigram	binary	lem	Reg Log	0.735267	0.728073
32	unigram_binary_stem_Reg Log	unigram	binary	stem	Reg Log	0.737333	0.726473
12	unigram_tfidf_stem_Random Forest	unigram	tfidf	stem	Random Forest	0.681067	0.683818

Topic Modelling

Per classificare i target(argomenti) sulla base della testo, si è utilizzato il Topic Modelling, che è un processo non supervisionato che permette l'identificazione dei contenuti del discorso (Topic) presenti all'interno di una collezione di documenti.

Analisi LDA

L'algoritmo di Topic Modelling utilizzato è l'LDA, che determina la probabilità di appartenenza di un documento ad un Topic.

Permette quindi di determinare quali documenti trattano di uno stesso argomento (da dedurre) raggruppandoli in cluster.

La prima fase è stato riprendere tutte le 12 matrici create in precedenza con pesi e token differenti, stimare il numero ottimale di Topic(da 5 a 30) e poi visualizzare il risultato.

Risultati Finali

Per definire il numero ottimale di Topic si sono utilizzati diversi attributi della libreria "GridSearchCV"

Esempio di Visualizzazione di Topic Modelling(LDA Model Unigram - tf-idf - lem):

Analisi LDA

L’algoritmo di Topic Modelling utilizzato è l’LDA, che determina la probabilità di appartenenza di un documento ad un Topic.

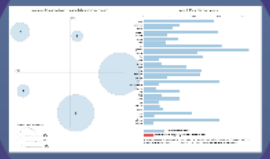
Permette quindi di determinare quali documenti trattano di uno stesso argomento (da dedurre) raggruppandoli in cluster.

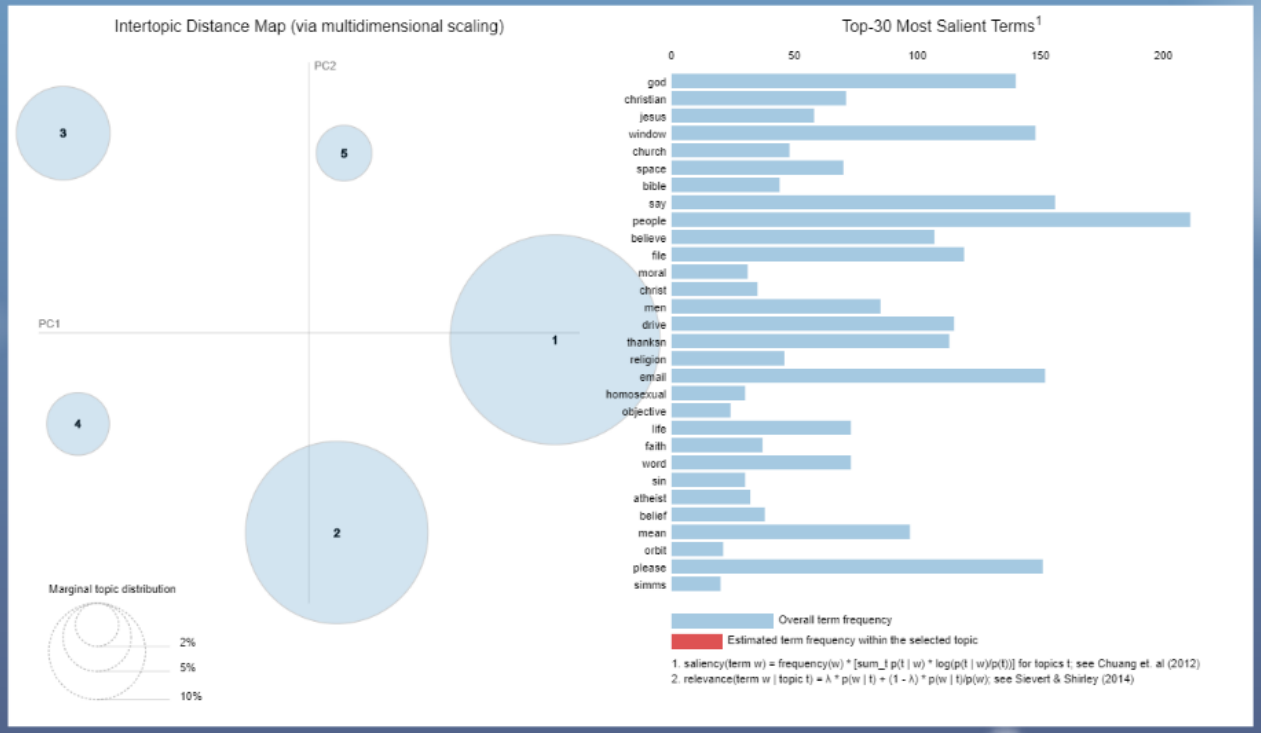
La prima fase è stato riprendere tutte le 12 matrici create in precedenza con pesi e token differenti, stimare il numero ottimale di Topic(da 5 a 30) e poi visualizzare il risultato.

Risultati Finali

Per definire il numero ottimale di Topic si sono utilizzati diversi attributi della libreria “GridSearchCV”

Esempio di Visualizzazione di Topic Modelling(LDA Model Unigram - tf-idf - lem):




Intertopic Distance Map (via multidimensional scaling)
PC2
PC1
1
2
3
4
5
Marginal topic distribution
2%
5%
10%
Top-30 Most Salient Terms[1]
0
50
100
150
200
god
christian
jesus
window
church
space
bible
say
people
believe
file
moral
christ
men
drive
thanksn
religion
email
homosexual
objective
life
faith
word
sin
atheist
belief
mean
orbit
please
simms
Overall term frequency
Estimated term frequency within the selected topic
1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

Conclusioni

Si possono trarre le seguenti conclusioni:

- Il modello Text Classification migliore, in termini di accuratezza, risulta essere il Naive Bayes nella rappresentazioni Unigram- Tf-Idf-Lem.
- Il modello di Text Clustering migliore, in termini di Topic, risulta essere LDA Model Unigram- Tf-Idf - Lem/Stem.

20 NewsGroups Classification

Corrado Montoro - 841489

Luca Lazzati - 850334