Data-Challenge

Abderrahim Mama - Mohamed Dhouib
X2019

Kaggle team : Barsha barsha

Outline

Introduction

I - Text features

- Text preprocessing
- Text embeddings

II - Graph features

III - Models and results

- Using classifiers
- Other unexplored ideas

Conclusion

Introduction

I - Text features

Text preprocessing

He is playing football day and night to be the best !!!

He is playing Football day and night to be the best

he is playing football day and night to be the best

playing football day night best

play football day night good

['play', 'football', 'day', 'night', 'good']

[0, 2, 6, 12, 65]

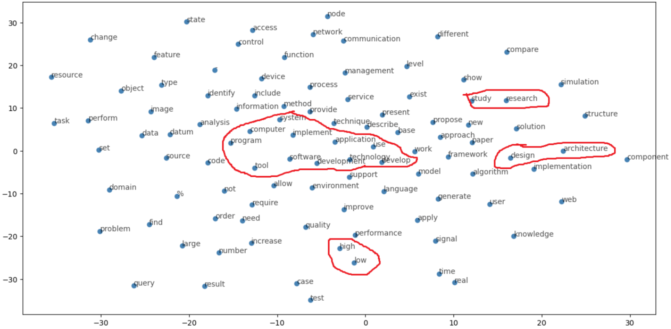
Text embeddings

Instead of $\begin{pmatrix} 0 \\ 1 \\ 0 \\ . \\ . \\ . \\ 0 \end{pmatrix} \in \mathbf{R}^{10000}$,

embeddings outputs of $\begin{pmatrix} 0.83 \\ 0.96 \\ 0 \\ . \\ . \\ 0.31 \end{pmatrix} \in \mathbf{R}^{l}$ where l is the latent space

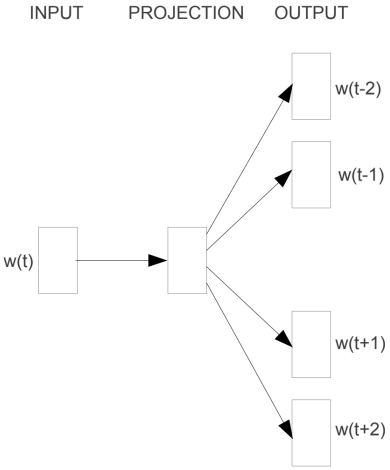
$l \in [10, 20, 60, 300]$

Text embeddings



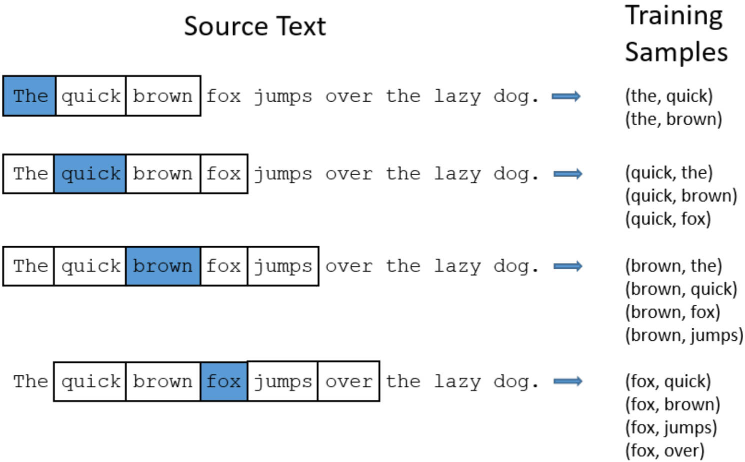
Text embeddings

The researcher won the nobel price



Skip-gram

Text embeddings



Source Text

Training Samples

The quick brown fox jumps over the lazy dog. →

(the, quick)
(the, brown)

The quick brown fox jumps over the lazy dog. →

(quick, the)
(quick, brown)
(quick, fox)

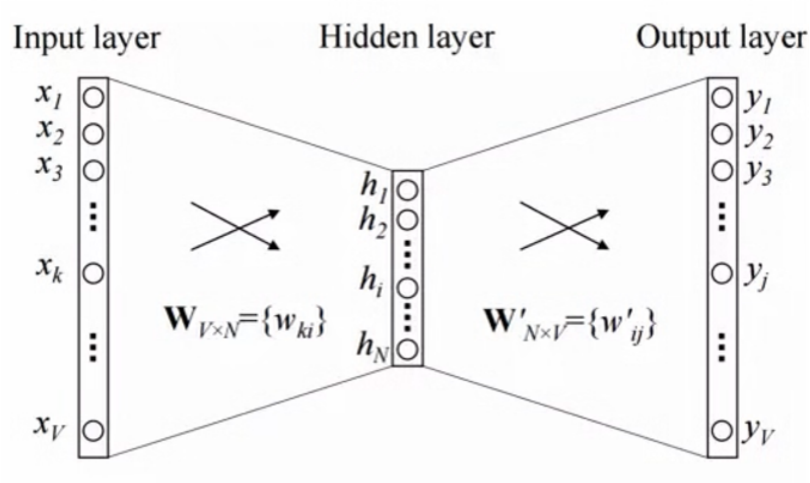
The quick brown fox jumps over the lazy dog. →

(brown, the)
(brown, quick)
(brown, fox)
(brown, jumps)

The quick brown fox jumps over the lazy dog. →

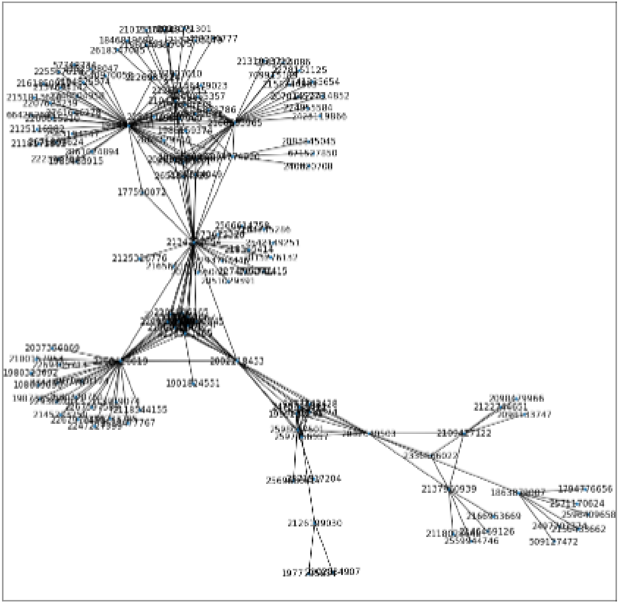
(fox, quick)
(fox, brown)
(fox, jumps)
(fox, over)

Text embeddings



II - Graph features

Graph features



Graph features

- Degree
- Neighbor's average degree : $AN(u) = \frac{\sum_{v \in N(u)} deg(v)}{|N(u)|}$
- Core number
- Onion layer number
- Pagerank : $PR(u) = \sum_{v \in N(u)} \frac{PR(v)}{deg(u)}$
- Papers number
- number of triangles
- Deep walk : walk length=10, dimensions=32, window size=8
- Eigenvector centrality : $Ax = \lambda x$ where λ : largest eigenvalue of the adjacency matrix A

III - Models and results

Models and results

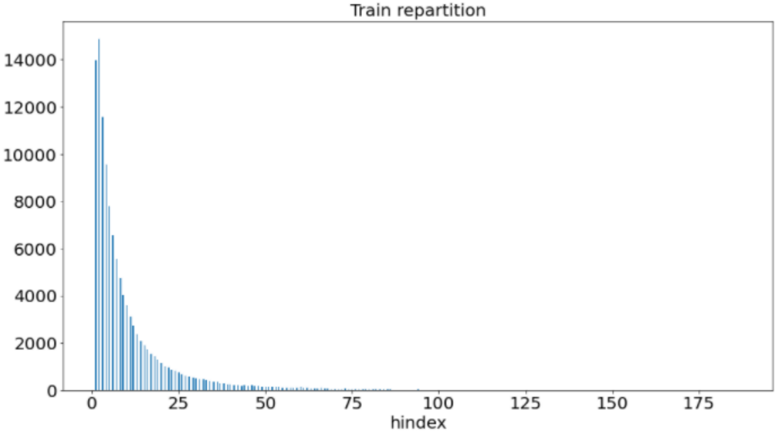


Figure: train dataset

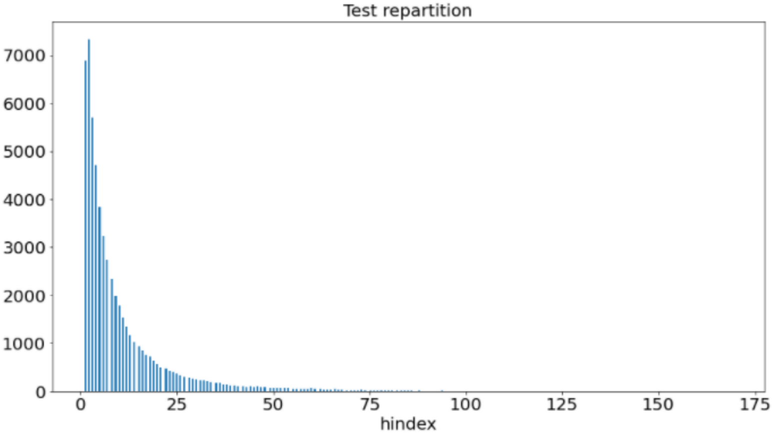


Figure: test dataset

Models and results

- XGboost
- LightGbm regressor
- Lasso :

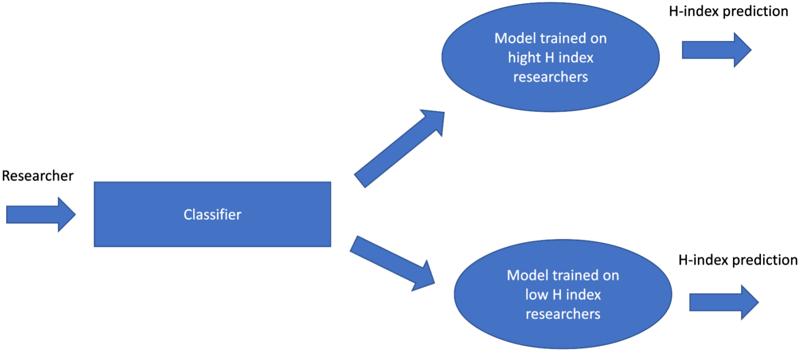
  $$\min_{w} ||y - Xw||_2^2 + \alpha ||w||_1$$

  $[\alpha = 1]$
- Multi Layer Preceptron : 2 hidden layers with [32,64,128] neurons.
- Knn regressor : hindex (author)= $\frac{\sum_{u \in N_k} hindex(u)}{k}$ where N_k contains the k nearest neighbours of the author : $[k \in$ range (5,20,2)]

Models and results

Regressor	Graph	Embeddings	Graph + embeddings
XGboost	98.7	95.1	64.2
LightGbm	**93.5**	**88.3**	**59.8 (59.3)**
Lasso	125.1	103.3	82.8
MLP	97.1	97.8	65.1
Knn regressor	130.2	120.5	88.3

Using classifiers


H-index prediction
Model trained on hight H index researchers
Researcher
Classifier
H-index prediction
Model trained on low H index researchers

Other unexplored ideas

Using KNN for the graph to generate missing abstracts

Training the embedding part with the actual model while prediction the H-index

Conclusion