Project Name : Resume classification and Resume Parsing

Application : Classifying the inbound resumes into different categories

and retrieving the needed information like Name, Phone

number, E-mail, Education, Skills, Experience

Algorithm’s

Used : Latent Dirichlet Allocation for Resume classification

Spacy and Regex for

Metric Used : Perplexity and Coherence score

Model

Description : Firstly we built LDA topic modeling for classifying our

inbound resumes into different categories so that we can

retrieve specific category resumes from ton of resumes

then the required time for searching for specific resumes

decreases.

And then from the sorted resumes we retrieve the

Information like Name, Phone Number, E-mail

Education, Skills by using Regex for rule based search and

some pre-trained api’s like Spicy .

Algorithm’s Flow Chart :

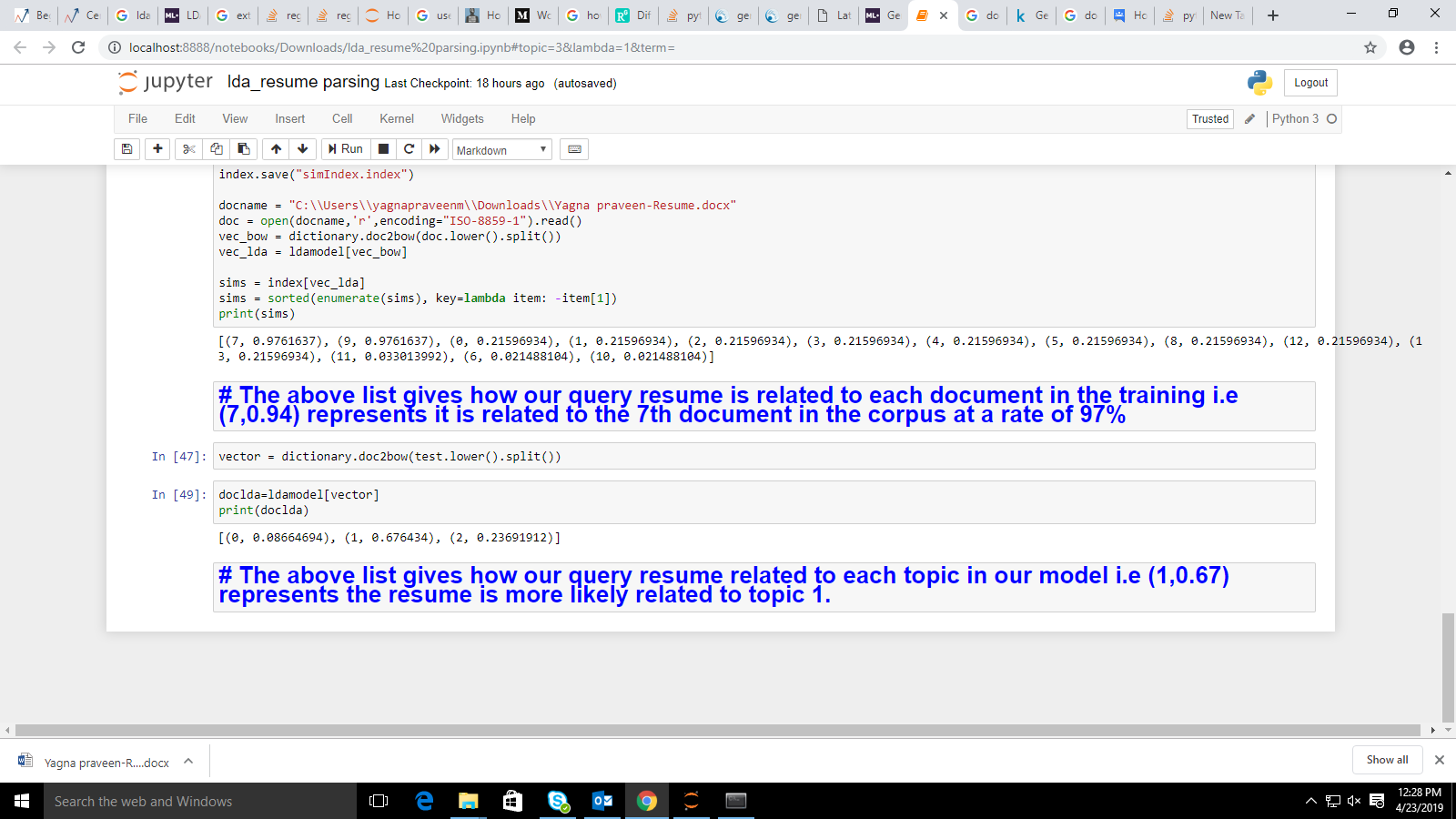
LDA topic modeling

Allocating Topics names

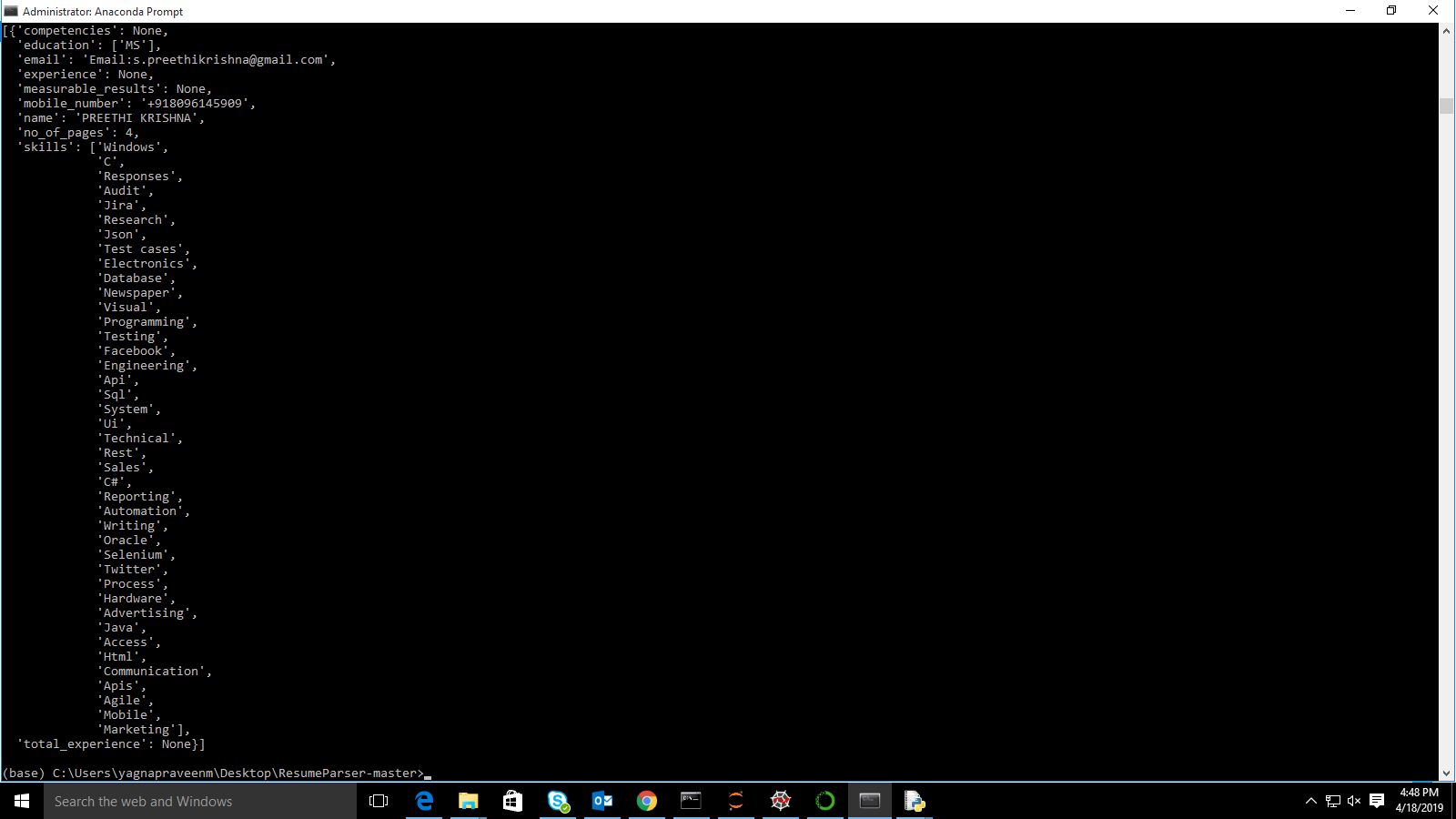
Model building for retrieving personal information using spicy

Rule based search for retrieving Education and skills

Retrieving Experience by RegEx



The above fig. shows as the sample output of the LDA model in which we are able to allocate specific topic to query resumes their by we can label our resume database into different categories like testing related resumes and Data science related and much more based on our requirement.



Above fig. shows the information retrieval from the classified resumes

After classifying the resumes into different categories we can retrieve information from classified resumes based our specific requirement

For example we have 100 resumes in our database can can’t read every resumes and for a specific requirement, by this approach we classify 100 resumes into 4 or 5 categories so we have 15 20 resumes to look over for our requirement,

In these 15 to 20 resumes we retrieve information and have a look over it with out reading 4 5 pages of resumes and finally we can select specific number of

candidates based on our requirement.

Coming to model performance we took Coherence score as metric for LDA i.e higher the score better will be the model here we trained our model with less data so we got Coherence score as 0.33,we can increase this score by feeding huge data to our model and fine tuning hyperparameters.

Pro’s of the model:

* Decrease time for searching specific resume
* After sorting specific resume some resumes have 10 15 pages of information ,reading all these pages is hectic job this model decreases time for selecting the resumes without reading all the pages by retrieving specified information into less than one page

Con’s of the model:

* Because we are using Regular Expression , information retravel in some cases like when the experience is not specified as separate heading model is not giving correct results in that case.