blog topic prediction

System Requirements:

- 16 GB RAM
- 12 GB RAM GPU 1080 Ti
- i7-8700 @ 3.20 GHz

Dependency:

- Custom NER, my bachelor year project NER
- Google word2vec
- gensim
- Mallet LDA
- spacy
- nltk

Grouping/classification of 114 instances (for 3 instances blog URL returned no data) of different blogs has to be done in the following way

Marketing	Branding	Growth marketing	Growth strategies	Product Management
Product discovery	Product Growth	Product Management Fundamentals	Agile principles	Company Culture
Company Growth	People Management	Startup Fundamentals	Interpersonal skills	Business Fundamentals
Business Growth	Sales Growth	Investment cycle		

Steps followed in the Machine learning pipeline

- Gather data in `raw_blog_content.csv` using gather_data.ipynb
- Clean data
- Build Feature

- Create Model
- Predict topics
- Map them on actual topics

In order to gather data/blog content, `requests` and `beautifulSoup4` and simple preprocessing was conducted in `gather_data.ipynb`. The preprocessing of data, with feature extraction and model creation is done in `lda_topic_modeling.py`. Three models were used and compared on Term Document frequency features those were

- lda
- ldamulticore
- lda mallet

The coherence and perplexity scores of each were checked and best model was picked to predict the topic of a given blog. In this case lda mallet showed best coherence of around 0.42. Due to the time constraint this metric could not be improved further.

Lastly, for each blog prominent topics were caluclated and were mapped to given topics, here I have used word2vec. I have calculated the vector of the predicted topic phrase and given topic and using Word Mover's Distance [https://github.com/mkusner/wmd/] calculated document distance. The result are written back to JSON file, `articles_topic.json`.