



BITS Pilani
Pilani | Dubai | Goa | Hyderabad

PRESENTATION ON:

TOPIC CATEGORISATION

WORK REPORT | JULY 2021


OBJECTIVE



TOPIC CATEGORISATION

Topic Categorization is a Natural Language Processing (NLP) technique that allows us to extract meaning from text automatically by detecting recurring themes or topics. Categorizing news pieces into designated topics is our primary goal.

To test our final model, we will use data pre-processing techniques such as lemmatization and stemming, followed by Latent Dirichlet allocation (LDA).




WHAT'S NEW

We're aiming to determine the heading of a paragraph in this project and to do so, we've constructed a Python code that uses machine learning algorithms to generate a set of words that are closely related to the content of the paragraph. Output also provides statistical data to illustrate how closely the word is related to the paragraph, in addition to the collection of relatable words. The user can obtain a good notion about the topic by looking at the output.

TARGET AUDIENCE



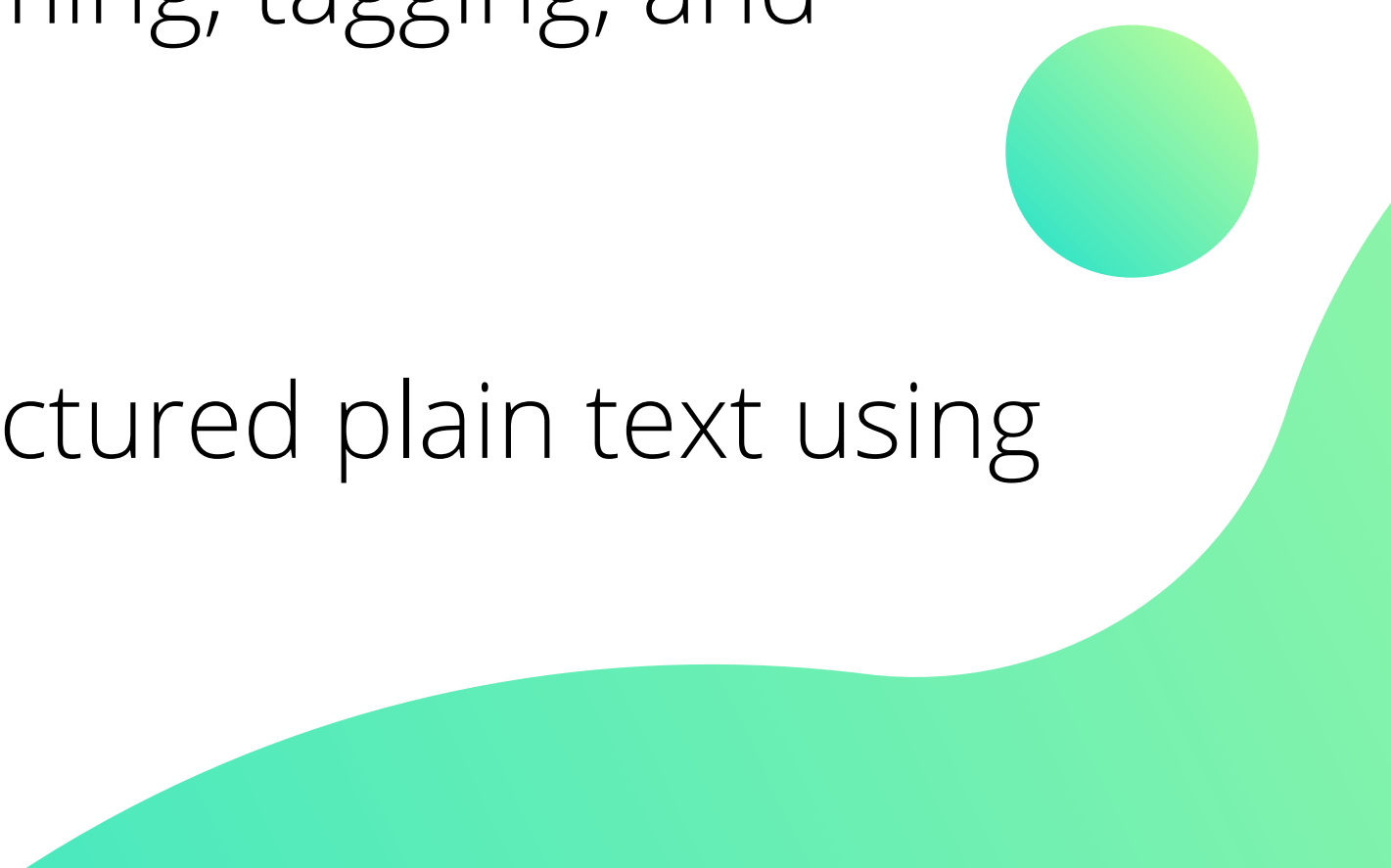
A Target Audience is a selected group of consumers who share common needs and characteristics. We have chosen our audience based on their behavioral characteristics

- We will be using topic categorization to understand the audience sentiments on social media like Twitter which carry emotion with itself for eg, racist, funny, etc. It can be concluded from the group of words we discovered using our code.
 - In today's world, people don't have time to read the full articles so creating an attractive heading for the paragraph might attract a large number of readers...
- 

TECHNOLOGIES USED

We have used Python 3 and its libraries to implement our project.

Libraries Used:

- nltk (Natural Language Toolkit): It contains text processing libraries for tokenization, parsing, classification, stemming, tagging, and semantic reasoning.
 - gensim: It is designed to process raw, unstructured plain text using unsupervised machine learning algorithms.
- 
- A decorative graphic in the bottom right corner consisting of a solid green circle and a larger, wavy green shape that resembles a stylized wave or a cloud.

Algorithm Used: LDA

Latent: This refers to everything that we don't know theoretically and are hidden in the data. Here, the themes or topics that document consists of are unknown, but they are believed to be present as the text is generated based on those topics.

Dirichlet: In the context of topic modeling, the Dirichlet is the distribution of topics in documents and distribution of words in the topic.

Allocation: This means that once we have Dirichlet, we will allocate topics to the documents and words of the document to topics.

DEMONSTRATION

Our project is divided into following steps:

- 1. Tokenization:** It is the process of breaking text into pieces, called tokens, and ignoring characters like punctuation marks and spaces. There are a couple of different ways we can approach this:
 - **sentence tokenization :-** In this, the tokenizer looks for specific characters that fall between sentences, like exclamation points, and newline characters.
 - **word tokenization :-** breaking up the text into individual words. This is a critical step for many language processing applications.

DEMONSTRATION

Our project is divided into following steps:

2. Cleaning of document: In this part the paragraph will be converted into the list of words (bag of words) and then the words will be lemmatized, first letter of each word will be lowercased and our set of words will be cleaned by removing the unnecessary stopwords like punctuations and full stop.

DEMONSTRATION

After text-cleaning: ['text', 'classification', 'machine', 'learning', 'one', 'commonly', 'used', 'nlp', 'task', 'article', 'saw', 'simple', 'example', 'performed', 'python', 'sentimental', 'analysis', 'movie', 'review', 'loaded', 'trained', 'model', 'stored', 'variable', 'let', 'predict', 'sentiment', 'test', 'set', 'using']

DEMONSTRATION

Our project is divided into following steps:

3. Comparing the cleaned document with datasets and Scoring the set of 5 weighted words generated in order to check their relevance for being the topic: In this part each word from our dataset will be assigned weights and then we will compare cleaned set of words of our input text with our dataset and try to assign the score based on their similarities with the document present in our dataset.



Topic 1

How much is Topic 1 in Doc 1?

2



Topic 2

How much is Topic 2 in Doc 1?

0



Topic 3

How much is Topic 3 in Doc 1?

2





Topic 1

Topic 2

Topic 3

How much is Topic 1 in Doc 1?

2

How much is Topic 2 in Doc 1?

0

How much is Topic 3 in Doc 1?

2

How much is 'ball' in Topic 1?

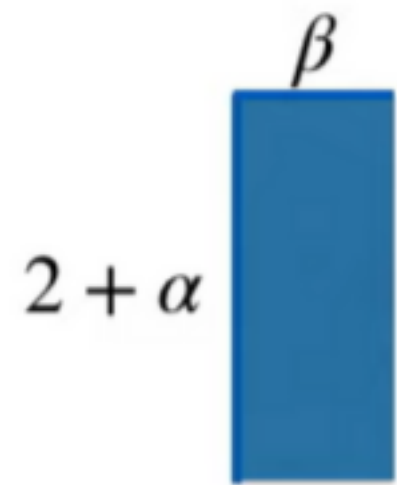
0

How much is 'ball' in Topic 2?

1

How much is 'ball' in Topic 3?

3



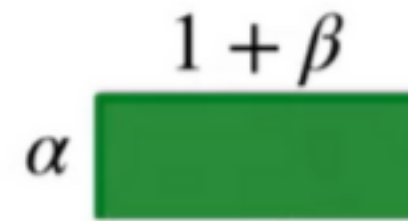
Topic 1

How much is Topic 1 in Doc 1?

$$2 + \alpha$$

How much is 'ball' in Topic 1?

$$0 + \beta$$



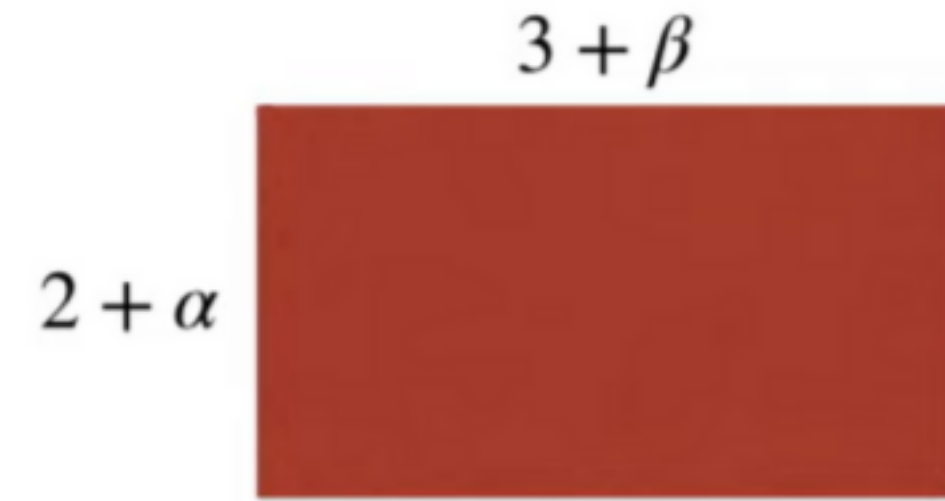
Topic 2

How much is Topic 2 in Doc 1?

$$0 + \alpha$$

How much is 'ball' in Topic 2?

$$1 + \beta$$



Topic 3

How much is Topic 3 in Doc 1?

$$2 + \alpha$$

How much is 'ball' in Topic 3?

$$3 + \beta$$

DEMONSTRATION

Sample Input

Text classification machine learning is one of the most commonly used NLP tasks. In this article, we saw a machine learning simple example of machine learning how text classification machine learning can be performed machine learning in Python. We performed the sentimental analysis of movie reviews. We loaded machine learning machine learning machine learning our trained model and stored it in the model variable. Let's predict the sentiment for the test set using o machine learning machine"

Sample Output

Score: 0.9918103218078613 Topic: 0.192*"task" + 0.192*"machine" + 0.192*"learning" + 0.192*"classification" + 0.192*"nlp"

DEMONSTRATION

Ps1 Project (github.com)

<https://gist.github.com/akshat4ever/4fdbaaeac38916860054114192b9232e>

FUTURE SCOPE

- We have tried our best to optimize the algorithm and have had some success, but there is still a lot of room for improvement. For example, a more streamlined algorithm may assist us in directly reach possible paragraph titles rather than providing relatable words.
- While working on the project, we realized that the algorithms we're using may be applied to a variety of situations. For example, in the output, we're receiving words that are closely related to the paragraph, and we're using it to find headings instead of taking into account a larger number of words. Further, we can correctly order the words and put them into a brief summary.

CONCLUSION

- We learned a lot about machine learning algorithms and Python fundamentals in this assignment. We attempted to tackle the problem in the most efficient manner possible. Thanks to topic analysis with the help of which we can execute complex tasks more effectively and obtain valuable insights from the available data which will lead to better business decisions..We hope that our solution will help to solve real-world issues.