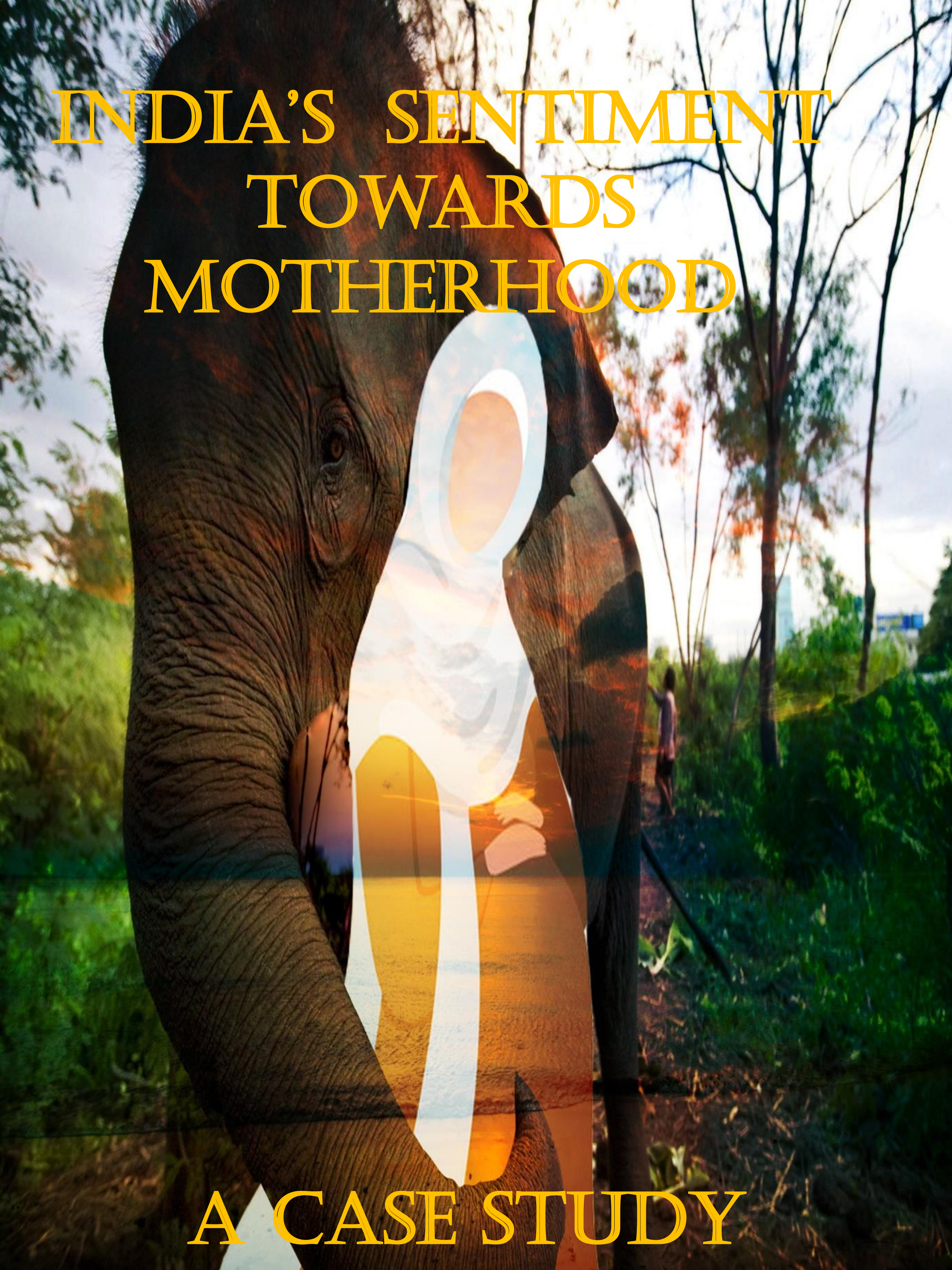
India's Sentiment Towards Motherhood



A Case Study

CONTENTS

INTRODUCTION

Kerala Elephant:

The news of a pregnant elephant that died in Kerala, on 27 May, after she ate a fruit that was stuffed with explosives, allegedly left by some locals, drew criticism and shock from people across the country.

The elephant died in the Mannarkkad division of Palakkad district.

Safoora Zargar:

Zargar, a 27-year-old research scholar in sociology at Jamia Millia Islamia, Delhi, was arrested on April 10 2020, on charges of blocking a road and obstructing traffic. After securing bail in that case, she was re-arrested on April 13 under the draconian provisions of the Unlawful Activities (Prevention) Act, 1967. She was implicated in a conspiracy that allegedly sparked the violence that engulfed Delhi at the end of February and placed in judicial custody. At the time of her arrest, Safoora Zargar was pregnant. She has been denied bail third time since May and continues to stay in Tihar jail despite the corona pandemic.

ABOUT THE STUDY

OBJECTIVE:

The focus of this study is to analyse how a random sample of the Indian society felt and reacted to two similar occasions where justice was required.
The analysis was done by scraping tweets regarding the two issues and Latent Dirichlet Allocation was performed on these two sets of tweets to extract the main topics in other words the essence of these tweets.

ABOUT THE DATA:

The tweets were scraped in the R platform using API key and tokens. The scraped data was then saved.

METHODOLOGY

TOPIC MODELLING:

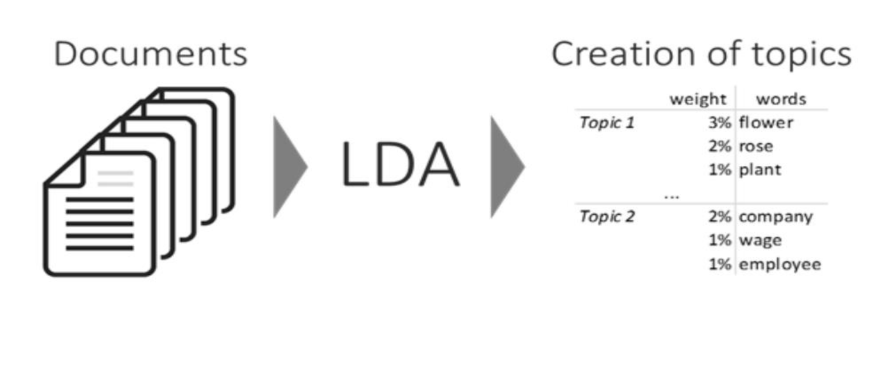
Topic Modelling is an unsupervised machine learning technique that’s capable of scanning a set of documents, detecting word and phrase patterns within them, and automatically clustering word groups and similar expressions that best characterize a set of documents.

There are several topic modelling methods, some of which are:

- Latent Dirichlet allocation
- Latent semantic analysis

LDA:

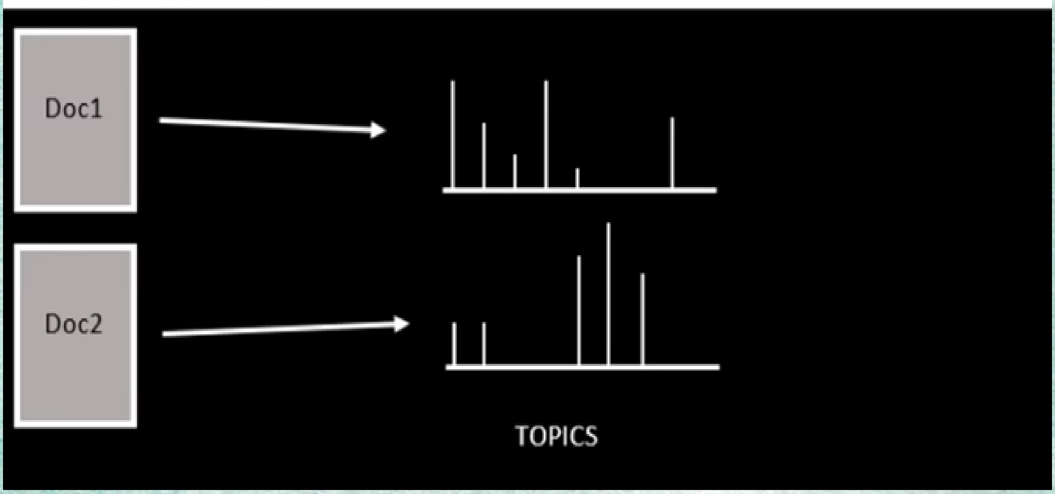
LDA (short for Latent Dirichlet Allocation) is an unsupervised machine-learning model that takes documents as input and finds topics as output. The model also says in what percentage each document talks about each topic.



	weight	words
Topic 1	3%	flower
	2%	rose
	1%	plant
...		
Topic 2	2%	company
	1%	wage
	1%	employee

ASSUMPTIONS:

- Documents with similar topics will make use of similar group of words.
- Documents are probability distributions over latent topics.
- Topics are probability distributions over words.



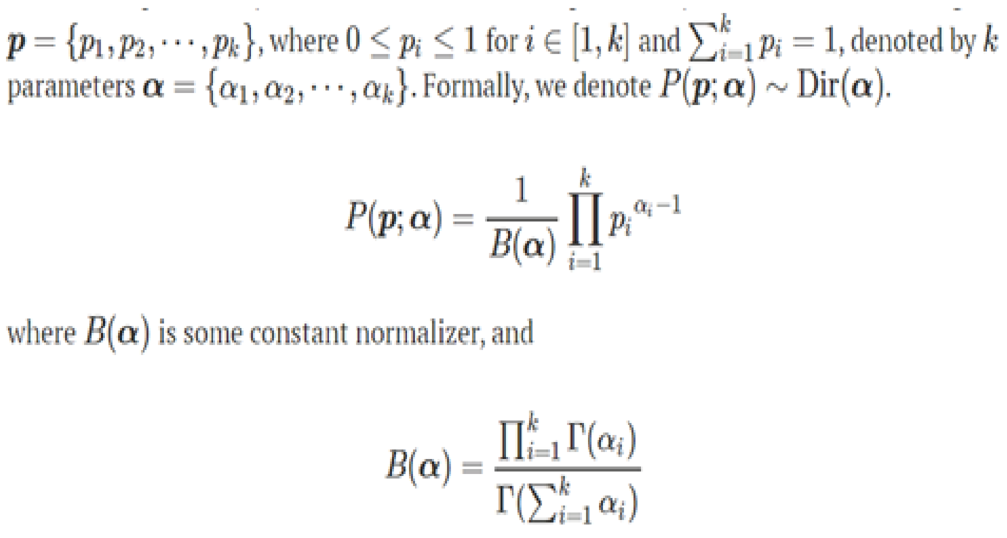
- Assumes that documents are written with arrangements of words and that those arrangements determine topics.
- Assumes that all words in the document can be assigned a probability of belonging to a topic.
- LDA assumes that the distribution of topics in a document and the distribution of words in topics are Dirichlet distributions.

Beta Distribution

Beta distribution is a family of continuous probability distributions well defined on the interval [0,1] parameterized by two positive shape parameters, denoted by α and β, that controls the shape of the distribution. Formally, we denote P(p:α,β)~Beta(α,β).

Dirichlet Distribution

Dirichlet is the multinomial version for the beta distribution. Dirichlet distribution is a family of continuous probability distribution for a discrete probability distribution for k categories



$\boldsymbol{p} = \{p_1, p_2, \cdots, p_k\}$, where $0 \le p_i \le 1$ for $i \in [1, k]$ and $\sum_{i=1}^{k} p_i = 1$, denoted by k parameters $\boldsymbol{\alpha} = \{\alpha_1, \alpha_2, \cdots, \alpha_k\}$. Formally, we denote $P(\boldsymbol{p}; \boldsymbol{\alpha}) \sim \mathrm{Dir}(\boldsymbol{\alpha})$.

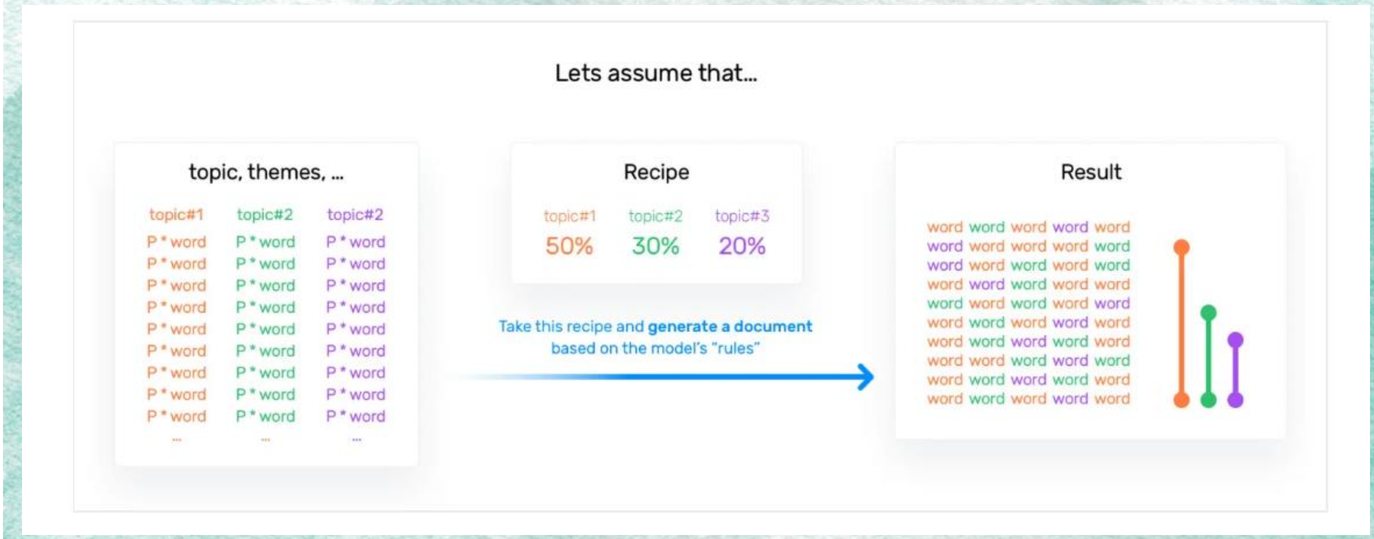
$$P(\boldsymbol{p}; \boldsymbol{\alpha}) = \frac{1}{B(\boldsymbol{\alpha})} \prod_{i=1}^{k} p_i^{\alpha_i - 1}$$

where $B(\boldsymbol{\alpha})$ is some constant normalizer, and

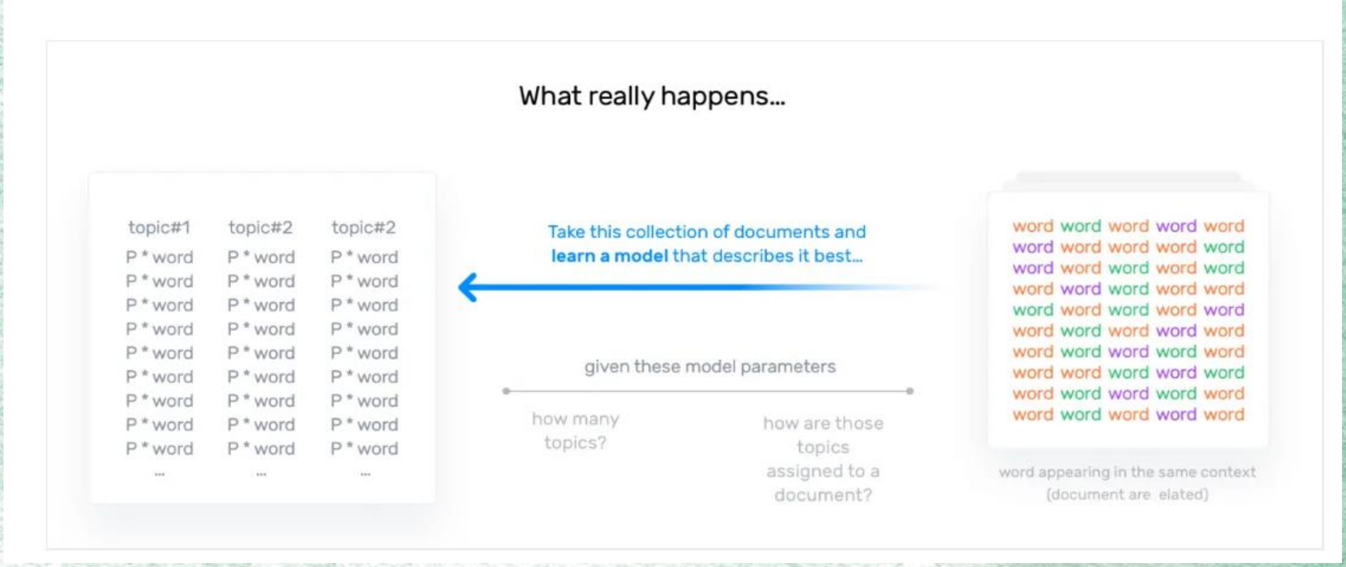
$$B(\boldsymbol{\alpha}) = \frac{\prod_{i=1}^{k} \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^{k} \alpha_i)}$$

Not surprisingly, when k=2, Dirichlet distribution becomes beta distribution.

- LDA assumes that documents are generated like this:



Using such generative model for a collection of documents, LDA tries to backtrack from the documents to find a set of topics that are likely to have generated the collection.



OVERVIEW OF HOW THE ALGORITHM WORKS:

Assumption:
We are assuming that all topic assignments except for the current word in question are correct, and update the assignment of the current word using our model on every iteration.

Suppose we have a collection of documents and we want to learn `K` topics out of them.

1. We will go through each document (d).
2. Then for each word (w) in the document we will calculate the following:-

- X = p(topic | document) = the proportion of words in document d that are currently assigned to topic t.
- Y = p(word w | topic t) = the proportion of assignments to topic t over all documents that come from this word w. (The same word can be in multiple documents, hence its coverage over the documents)

3. X * Y is essentially the probability that topic `t` generated word `w`

After repeating these steps for large enough number of times, we will get pretty good topic assignments, such that they generate words describing the documents.

PARAMETERS OF LDA

Alpha and Beta Hyperparameters :

Alpha represents document-topic density and Beta represents topic-word density. Higher the value of alpha, documents are composed of more topics and lower the value of alpha, documents contain fewer topics. On the other hand, higher the beta, topics are composed of a large number of words in the corpus, and with the lower value of beta, they are composed of few words.

Number of Topics:

Number of topics to be extracted from the corpus.

ANALYSIS

SAFOORA ZARGAR:

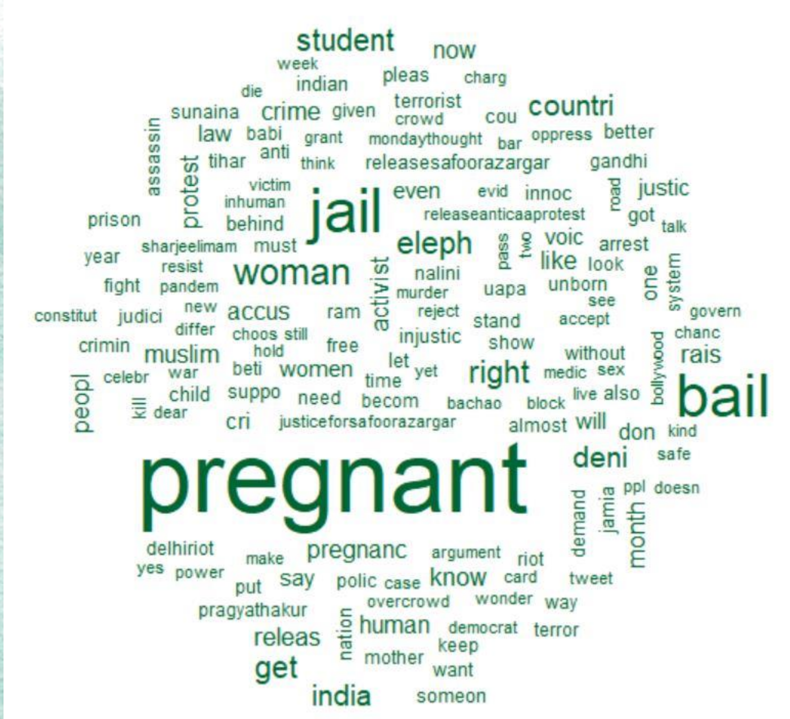


Fig 1

TOPIC 1	TOPIC 2	TOPIC 3
bail	pregnant	student
woman	jail	get
countri	eleph	india
rais	right	deni
peopl	releas	muslim
month	know	now
even	accus	women
pregnanc	like	justic
one	protest	don
better	will	law

Table 1

TOPIC	TERM	BETA
1	bail	0.0461
2	bail	0.00118
3	bail	0.00130
1	grant	0.000112
2	grant	0.000107
3	grant	0.00484
1	monster	0.00123
2	monster	0.000107
3	monster	0.000118
1	accus	0.000112

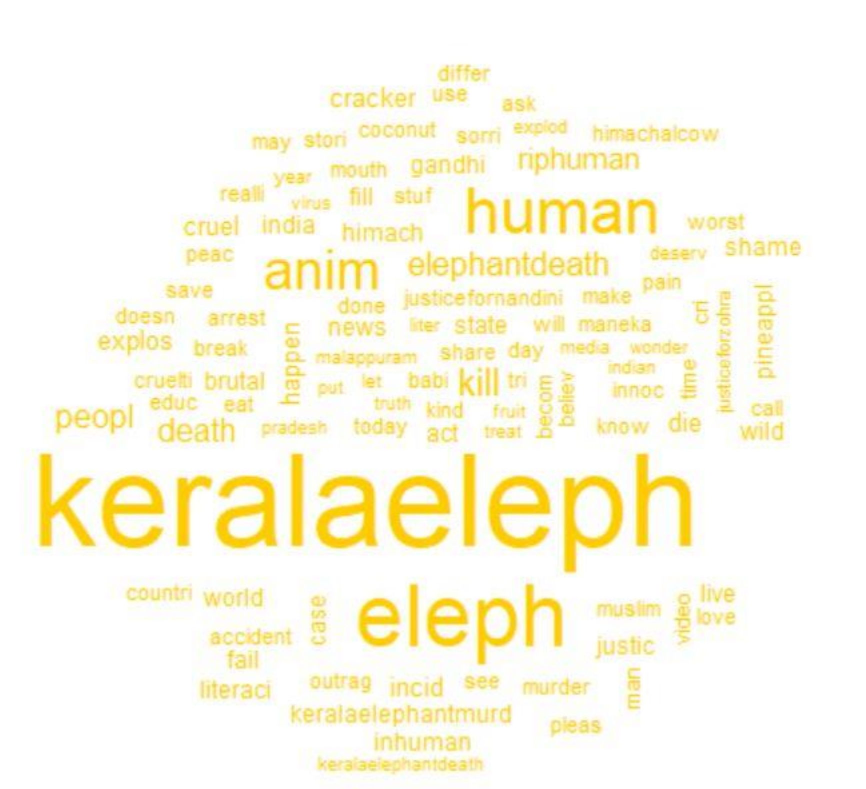
Table 2

DOCUMENT	TOPIC	GAMMA
1	1	0.352
2	1	0.316
3	1	0.350
4	1	0.299
5	1	0.333
6	1	0.292
7	1	0.311
8	1	0.339
9	1	0.361
10	1	0.378

Table 3

KERALA ELEPHANT:

Fig 2



TOPIC 1	TOPIC 2	TOPIC 3
human	eleph	keralaeleph
anim	kill	death
peopl	elephantdeath	incid
riphuman	explos	pineappl
die	cruel	case
shame	justic	gandhi
keralaelephantmurd	himach	fail
act	news	wild
fill	world	inhuman
state	india	mouth

Table 4

TOPIC	TERM	BETA
1	anywer	0.0000705
2	anywer	0.0000728
3	anywer	0.00153
1	busi	0.00148
2	busi	0.0000728
3	busi	0.0000730
1	case	0.0000705
2	case	0.0000728
3	case	0.0110
1	china	0.0000705

Table 5

DOCUMENT	TOPIC	GAMMA
1	1	0.292
2	1	0.351
3	1	0.327
4	1	0.292
5	1	0.355
6	1	0.367
7	1	0.280
8	1	0.322
9	1	0.333
10	1	0.345

Table 6

CONCLUSION

SAFOORA ZARGAR:

Figure 1 shows the words that appeared mostly in the tweets, we can see that majority of the words were highlighting the injustice towards Ms Zargar and demanding a bail to release her.

From Table 1 we can say that :
"**Topic 1**" highlights granting **Justice** for Ms Zargar.
"**Topic 2**" highlights **Right to Freedom** of Ms Zargar.
"**Topic 3**" highlights **Discrimination** against Ms Zargar.

Table 2 (3,163 x 3) gives the a one-topic-per-term-per-row format probabilities, called "Beta" from the model.
For each combination, the model computes the probability of that term being generated from that topic.
For example, the term "bail" has a 0.0461 probability of being generated from Topic 1, 0.00118 probability of being generated from Topic 2 and 0.00130 probability of being generated from Topic 3.

Table 3 (861 x 3) examines the per-document-per-topic probabilities called "gamma" as LDA also models each document as a mixture of topics. Each of these values is an estimated proportion of words from that document that are generated from that topic. For example, the model estimates that only about 35% of the words in document 1 were generated from topic 1.

KERALA ELEPHANT:

Figure 2 shows the words that appeared mostly in the tweets, we can see that majority of the words were highlighting the inhumanity towards the elephant and demanding justice for the cruelty.

From Table 4 we can say that :
"**Topic 1**" highlights existence of **Inhumanity** .
"**Topic 2**" highlights granting **Justice**.
"**Topic 3**" highlights **Shamelessness** .

Table 5 (3,816 x 3) gives the a one-topic-per-term-per-row format probabilities, called "Beta" from the model.
For each combination, the model computes the probability of that term being generated from that topic.
For example, the term "anywer" has a 0.0000705 probability of being generated from Topic 1, 0.0000728 probability of being generated from Topic 2 and 0.00153 probability of being generated from Topic 3.

Table 6 (1,383 x 3) examines the per-document-per-topic probabilities called "gamma" as LDA also models each document as a mixture of topics. Each of these values is an estimated proportion of words from that document that are generated from that topic. For example, the model estimates that only about 29% of the words in document 1 were generated from topic 1.

REFERENCES

https://monkeylearn.com/blog/introduction-to-topic-modeling/
https://towardsdatascience.com/the-complete-guide-for-topics-extraction-in-python-a6aaa6cedbbc
https://www.youtube.com/watch?v=DWJYZq_fQ2A&t=132s
https://medium.com/@pratikbarhate/latent-dirichlet-allocation-for-beginners-a-high-level-intuition-23f8a5cbad71
https://leimao.github.io/blog/Introduction-to-Dirichlet-Distribution/
https://www.analyticsvidhya.com/blog/2016/08/beginners-guide-to-topic-modeling-in-python/
https://www.tidytextmining.com/tidytext.html