Agenda:

✓ ① Parts of Speech Tagging

✓ ② Topic Modelling.

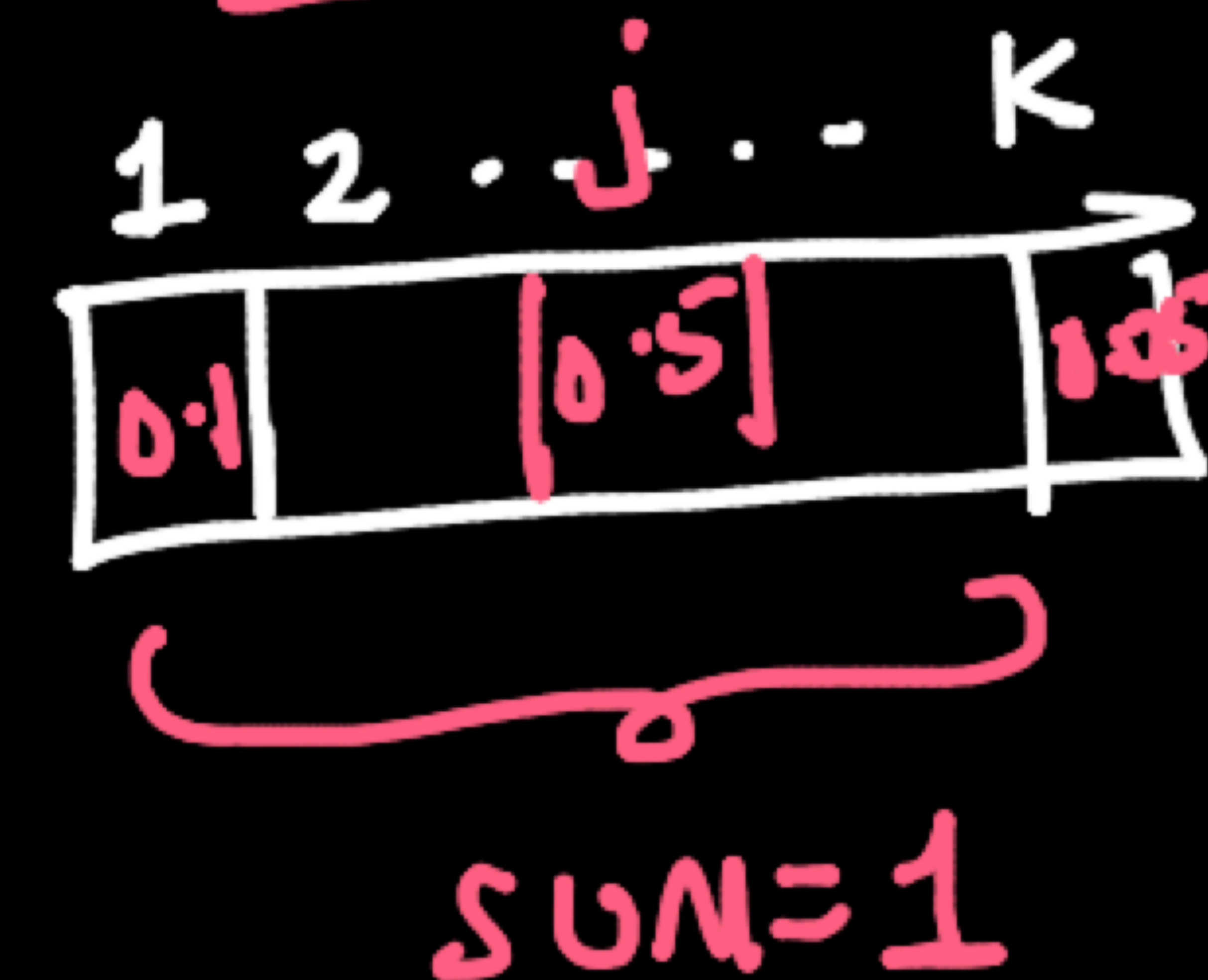
[Topic Modeling:]

e.g.: news Data

Task: {
M docs

each d_i →

3 K topics



Ideas:

doc-doc sim: TFIDF-w2vec

① → ② M-docs → words

↳ Aggl. clustering

[cluster ① ≈ topic] ✓

di

← d-dimen

② K-topics

② soft K-means ≈ GMM

Generative

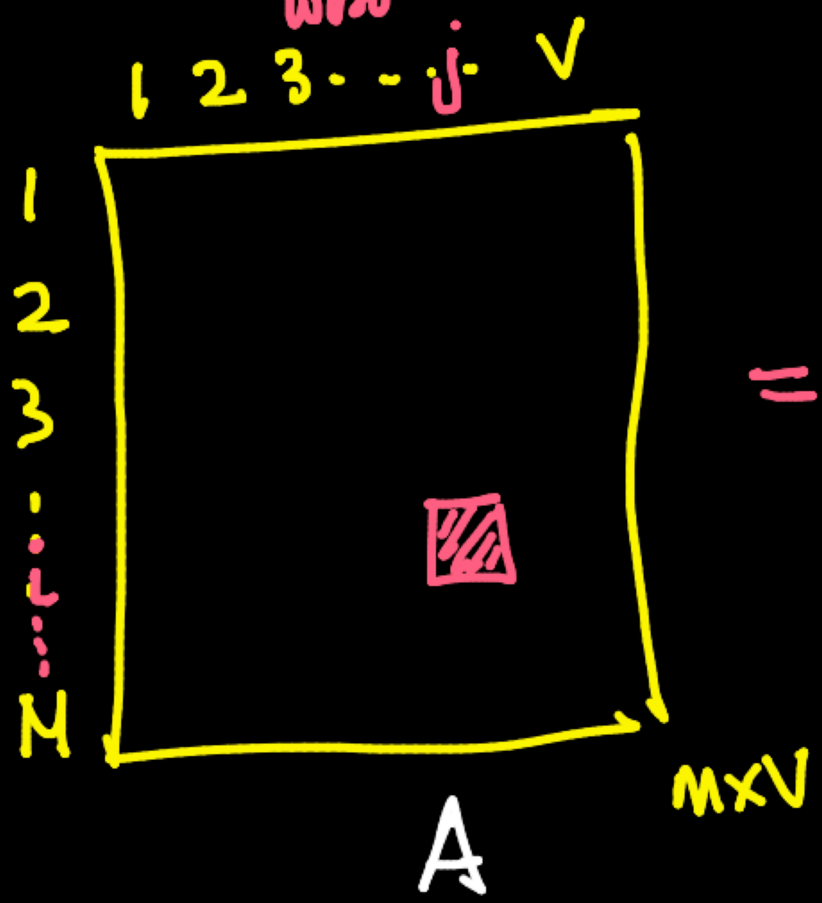
MF (?)

M-docs

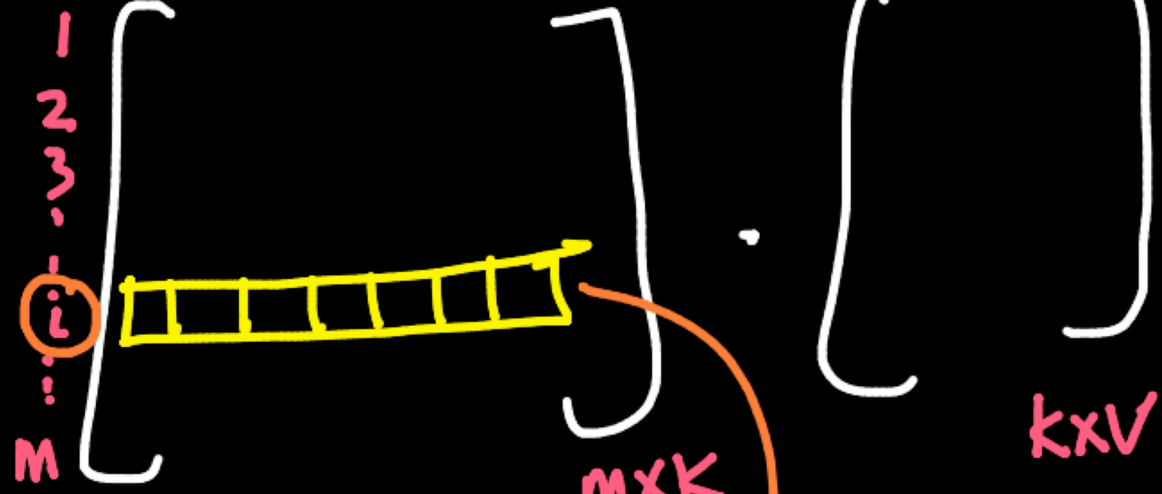
V: voc size

K-topics (soft-assign)

3



=



$\sum_m = 1$
(soft assignment)

Clustering & MF

$A \approx B \cdot C$

s.t each row B should form to 1

Loss: typical MF : $\sum_{ij} (A_{ij} - B_i \cdot C_j)^2$

(4)

B i^{th} row → prob. dist of i^{th} doc's assignment to K-topics

✓ { KL-divergence : measure how different 2 dist are?

Summary:

M-docs

↳ topics

K: numerical

soft K-clustering

↓

MF + KL-divergence

✓ { Latent .... Dirichlet Allocation (LDA) (~2003)

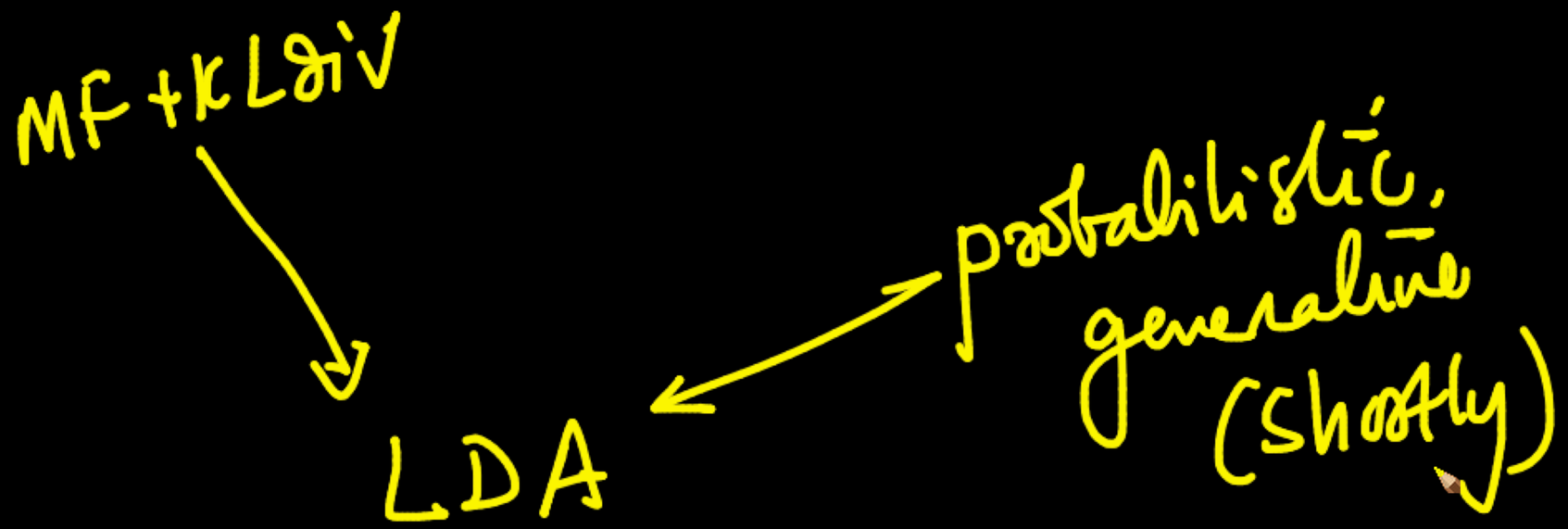
✓ probabilistic generative (intuition)

algo

GMM

MF

MF + KL-div ✓



✓ [parts of speech (pos) tagging

sentence: w_1 w_3 w_6 w_{12}

N PR V Adv ...

pos-tagging (NLP)

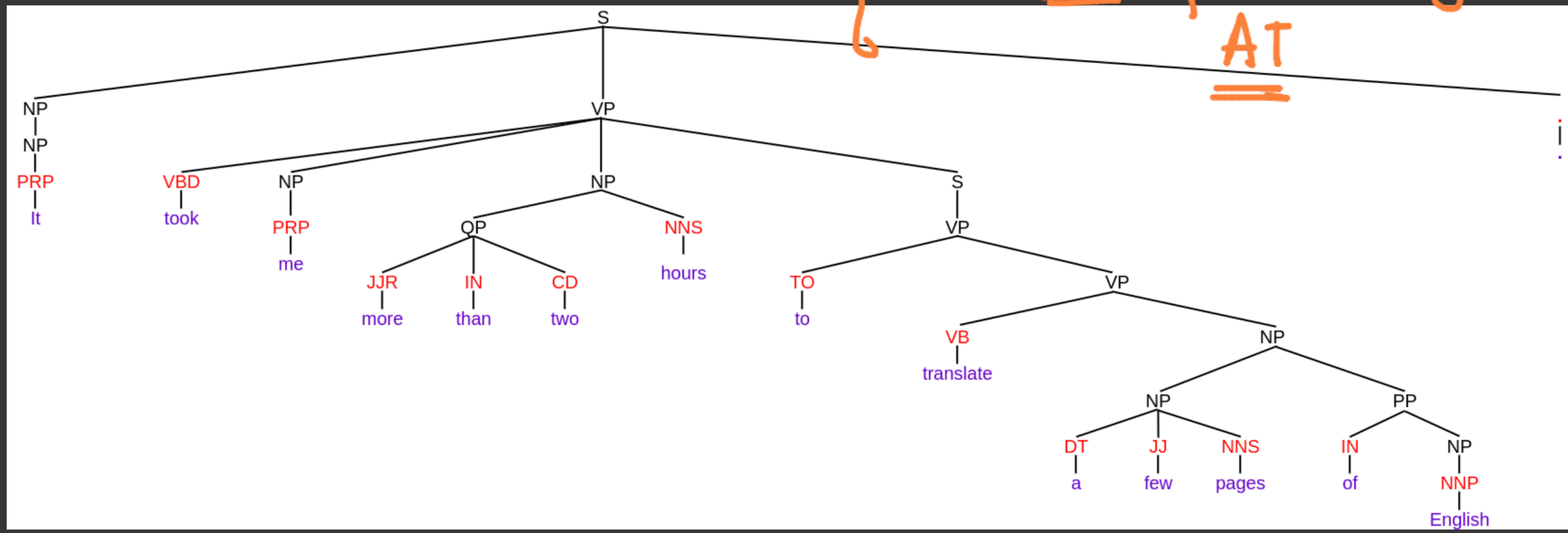
↳ rule-based / Grammar-based

→ probabilistic techs

→ DL-techniques

CS: Compiler design

AT



If you use spacy, though, you don't need to worry about writing your own rules for getting Noun Phrases.

```
[ ] counter_all = Counter(flatten(review_data["noun_phrases"].tolist()))
    counter_all.most_common(10)
```

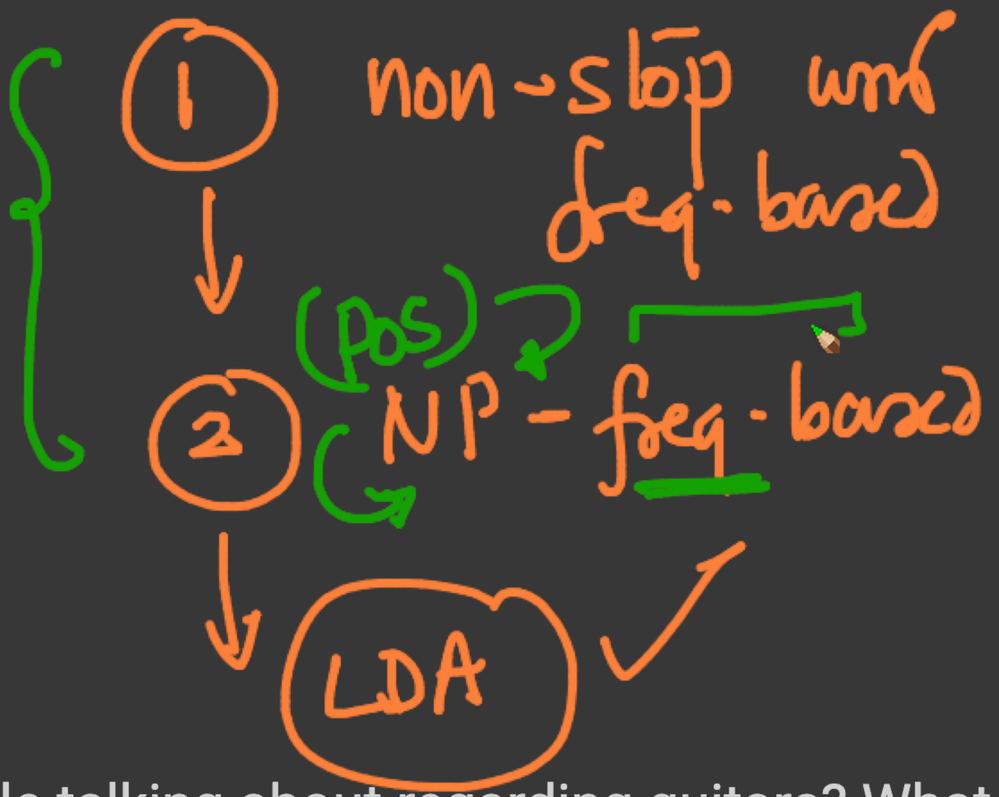
```
[('guitar', 3212),
 ('strings', 2526),
 ('price', 1795),
 ('sound', 1698),
 ('pedal', 1539),
 ('one', 1256),
 ('amp', 1173),
 ('time', 1146),
 ('tone', 1042),
 ('guitars', 984)]
```

But... these still aren't very actionable.

For example, we still don't know what are people talking about regarding guitars? What word/words is it related to?

```
[ ]  counter_all = Counter(flatten(review_data["noun_phrases"].tolist()))
     counter_all.most_common(10)
```

```
[('guitar', 3212),
 ('strings', 2526),
 ('price', 1795),
 ('sound', 1698),
 ('pedal', 1539),
 ('one', 1256),
 ('amp', 1173),
 ('time', 1146),
 ('tone', 1042),
 ('guitars', 984)]
```



But... these still aren't very actionable.

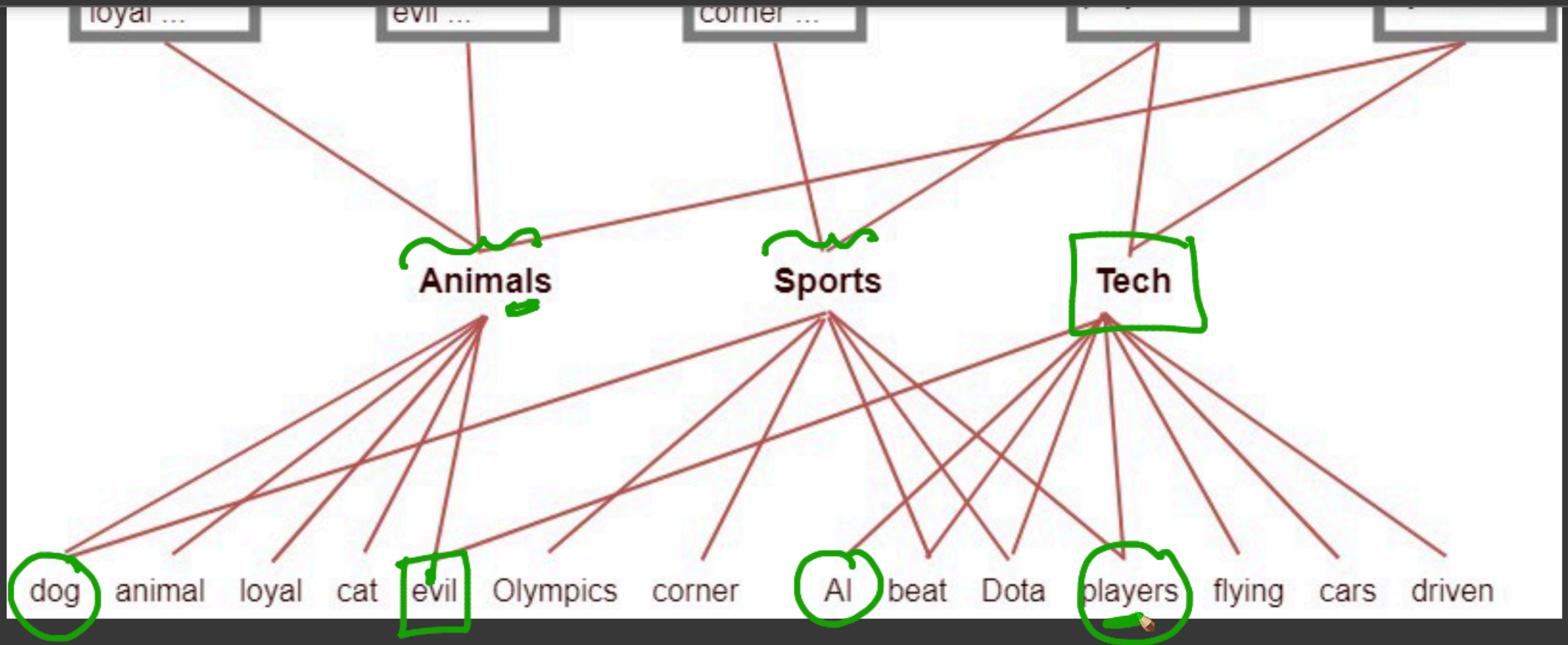
For example, we still don't know what are people talking about regarding guitars? What word/words is it related to?

100% 10261/10261 [04:08<00:00, 25.99it/s]

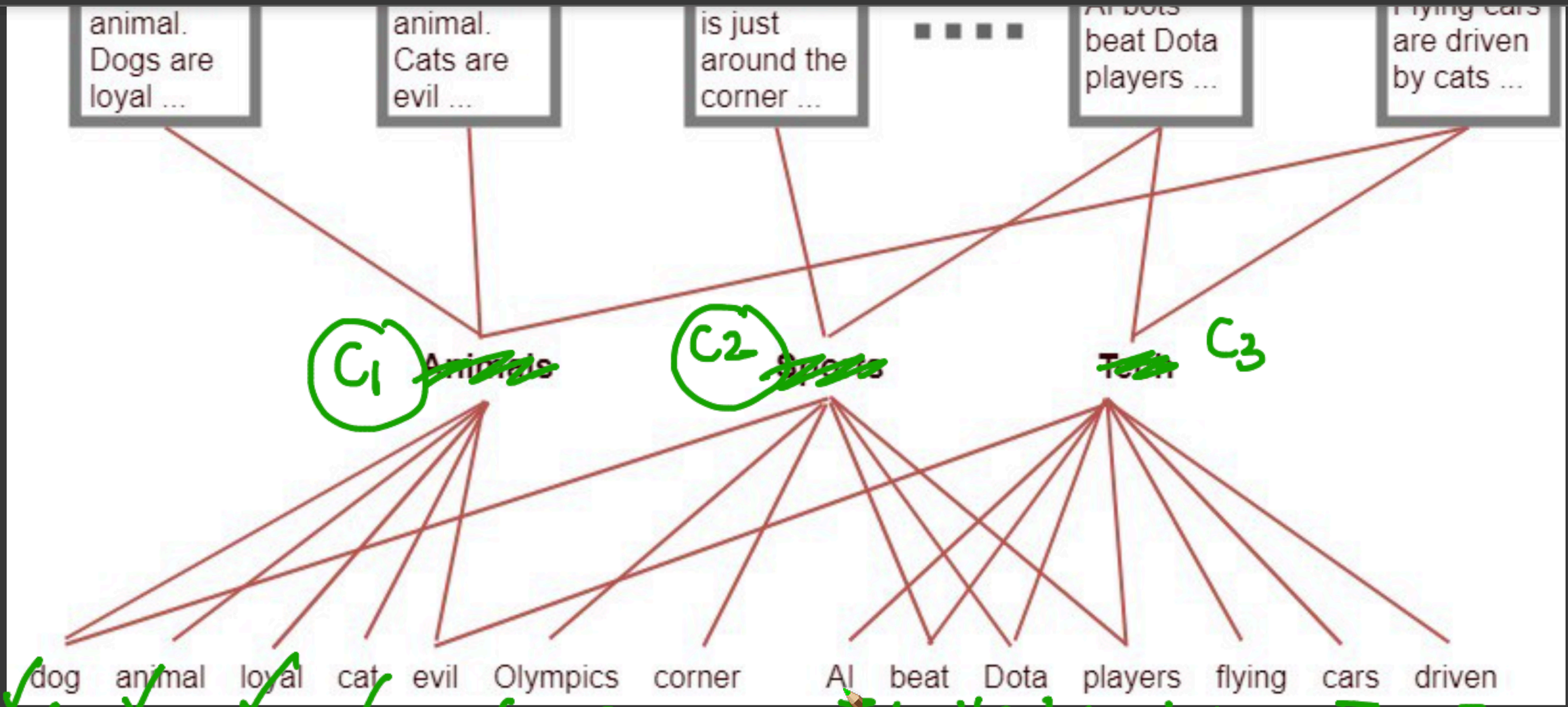
```
counter_all = Counter(flatten(review_data["noun_phrases_parent"].tolist()))
counter_all.most_common(10)
```

```
[('price->for', 583),
 ('years->for', 503),
 ('guitar->on', 400),
 ('thing->is', 347),
 ('strings->are', 278),
 ('pedal->is', 261),
 ('guitar->of', 255),
 ('tune->in', 255),
 ('job->does', 245),
 ('strings->of', 242)]
```

Hmm that turned out to not be adding much information for our purpose

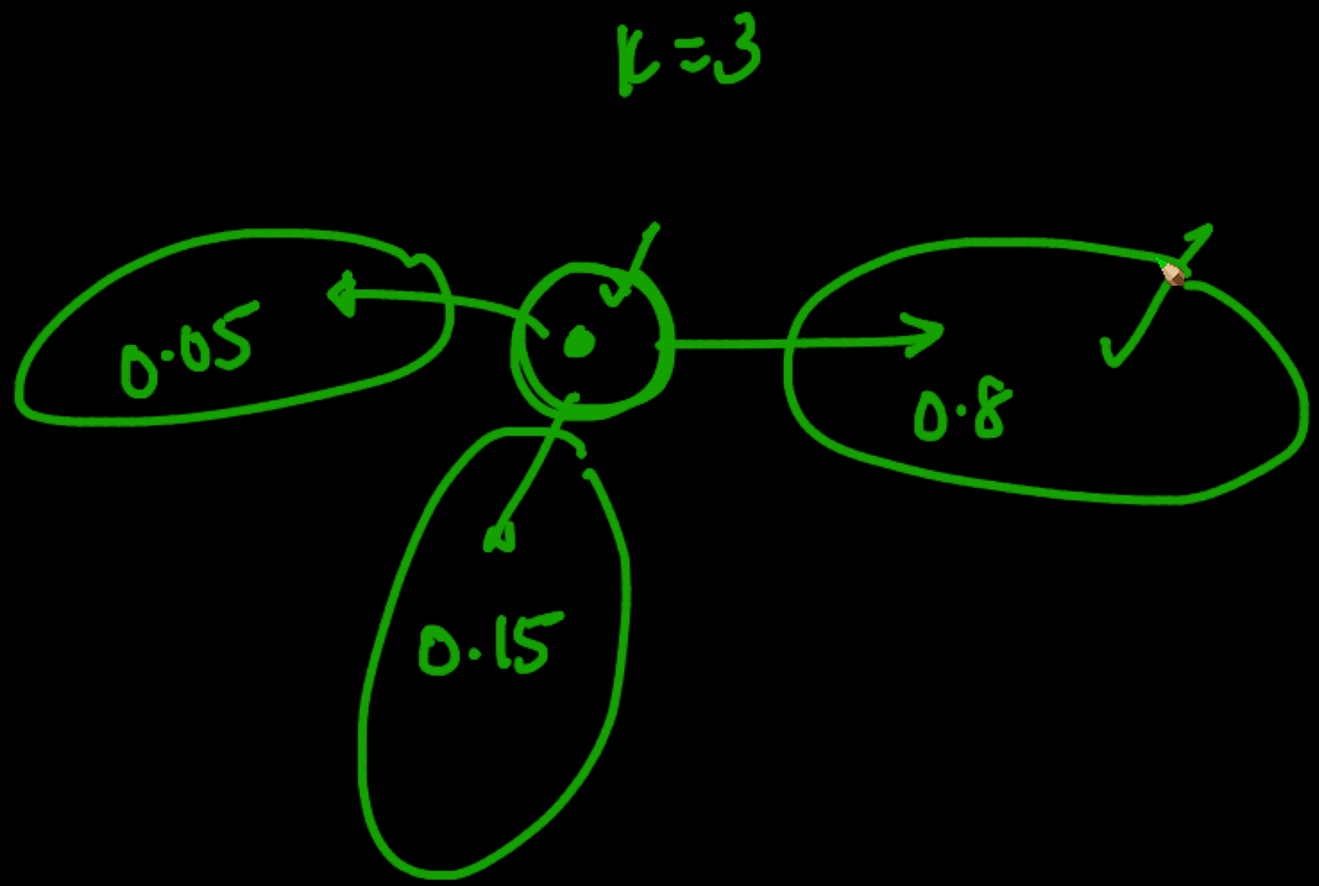


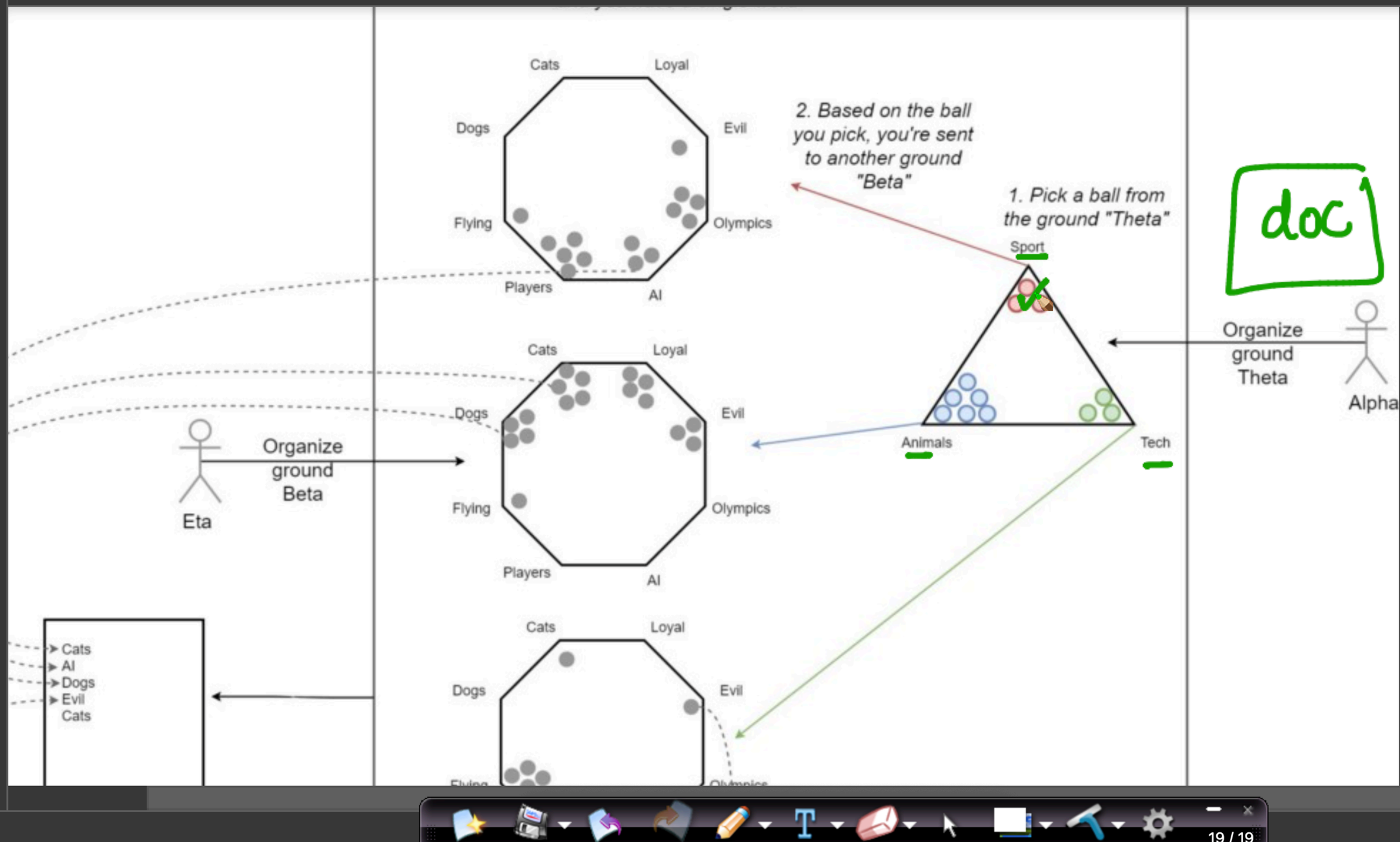
This re-definition of a review/document forms the very basis of a concept called as "Topic Modelling" where we try to find the topics in a set of document, by redifining the documents as a set of topics

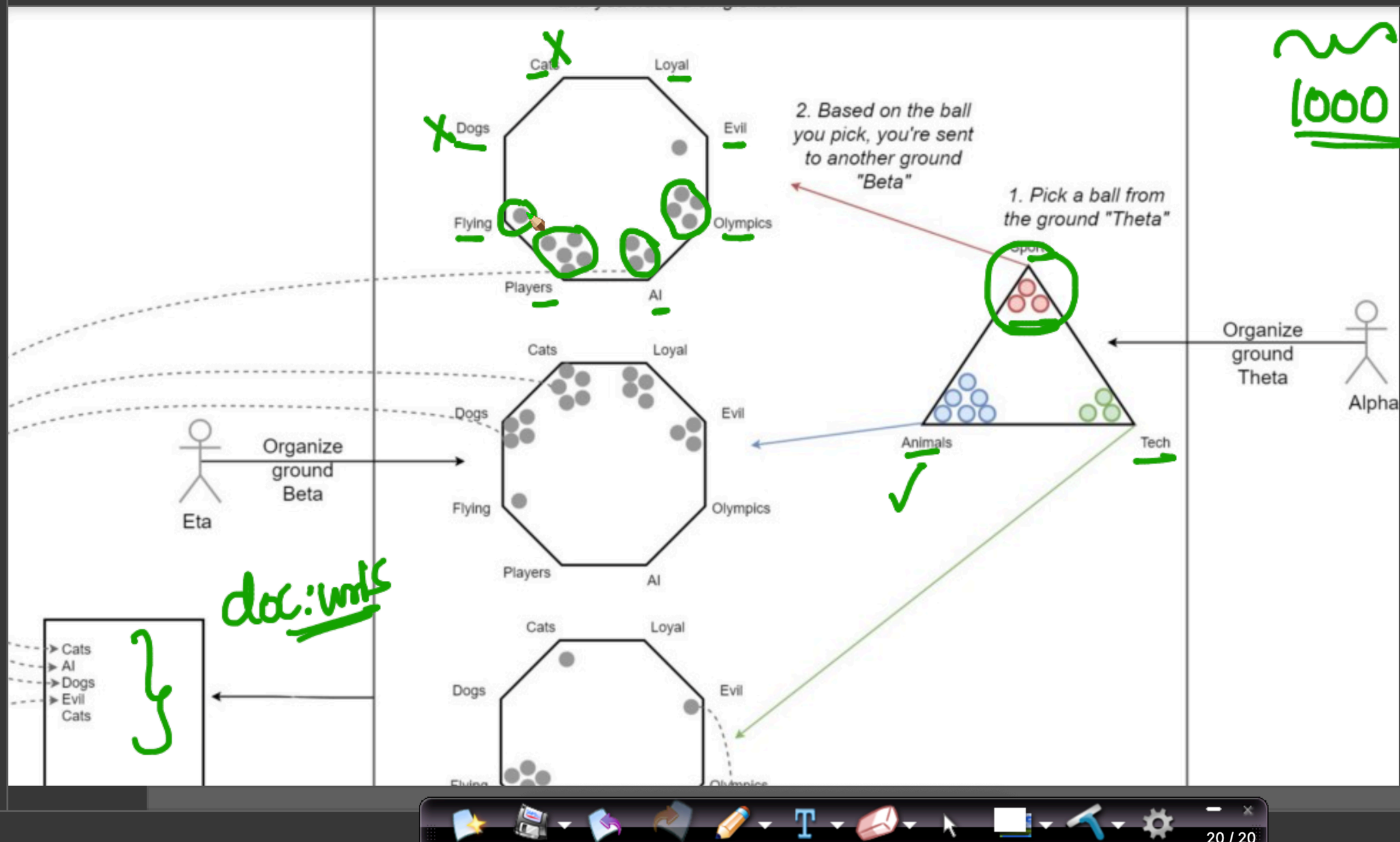


This re-definition of a review/document forms the very basis of a concept called as "Topic Modelling"

GMM:







Recap:-

M docs:

1

T1

0.3

0.2

T2

T3

2

doc

words

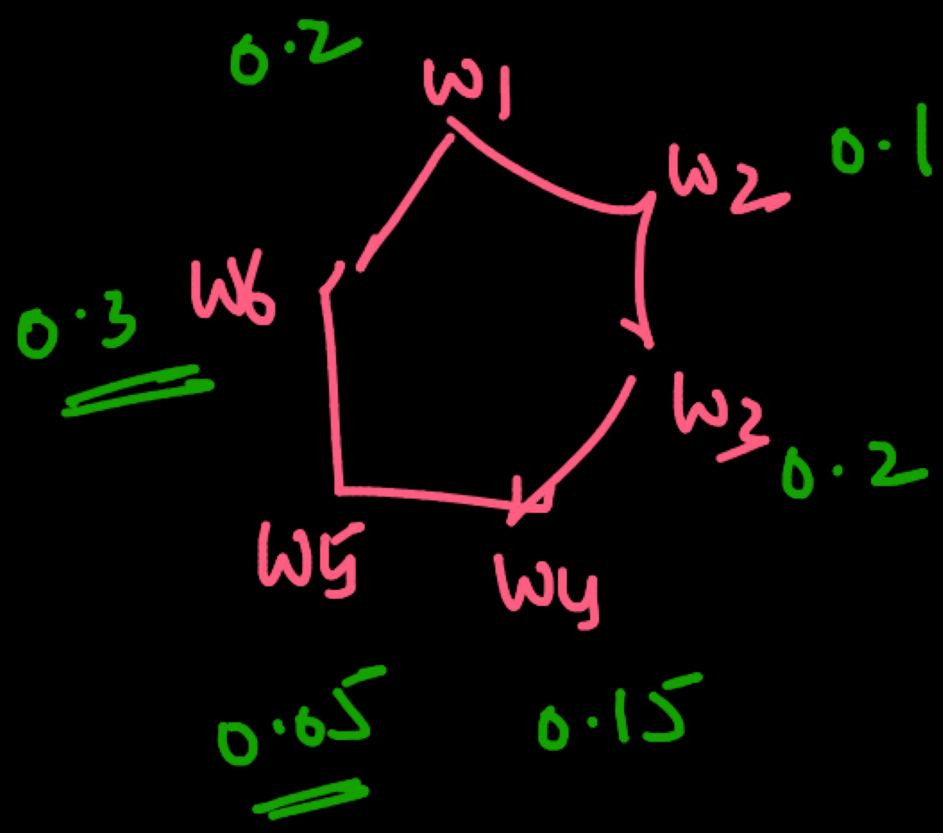
w3

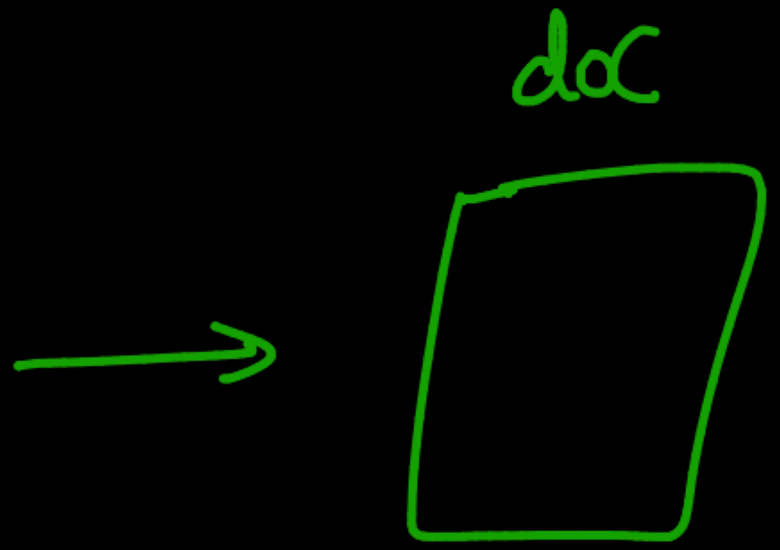
w5

w6

LDA

disb of words

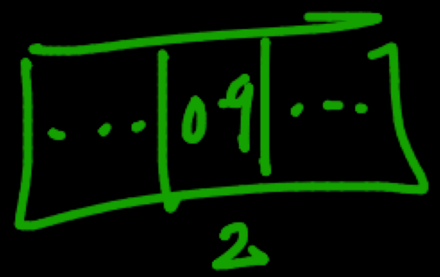




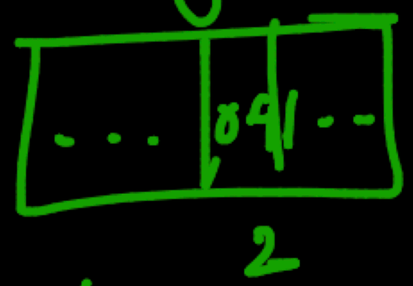
0.065 0.015

8

d_i

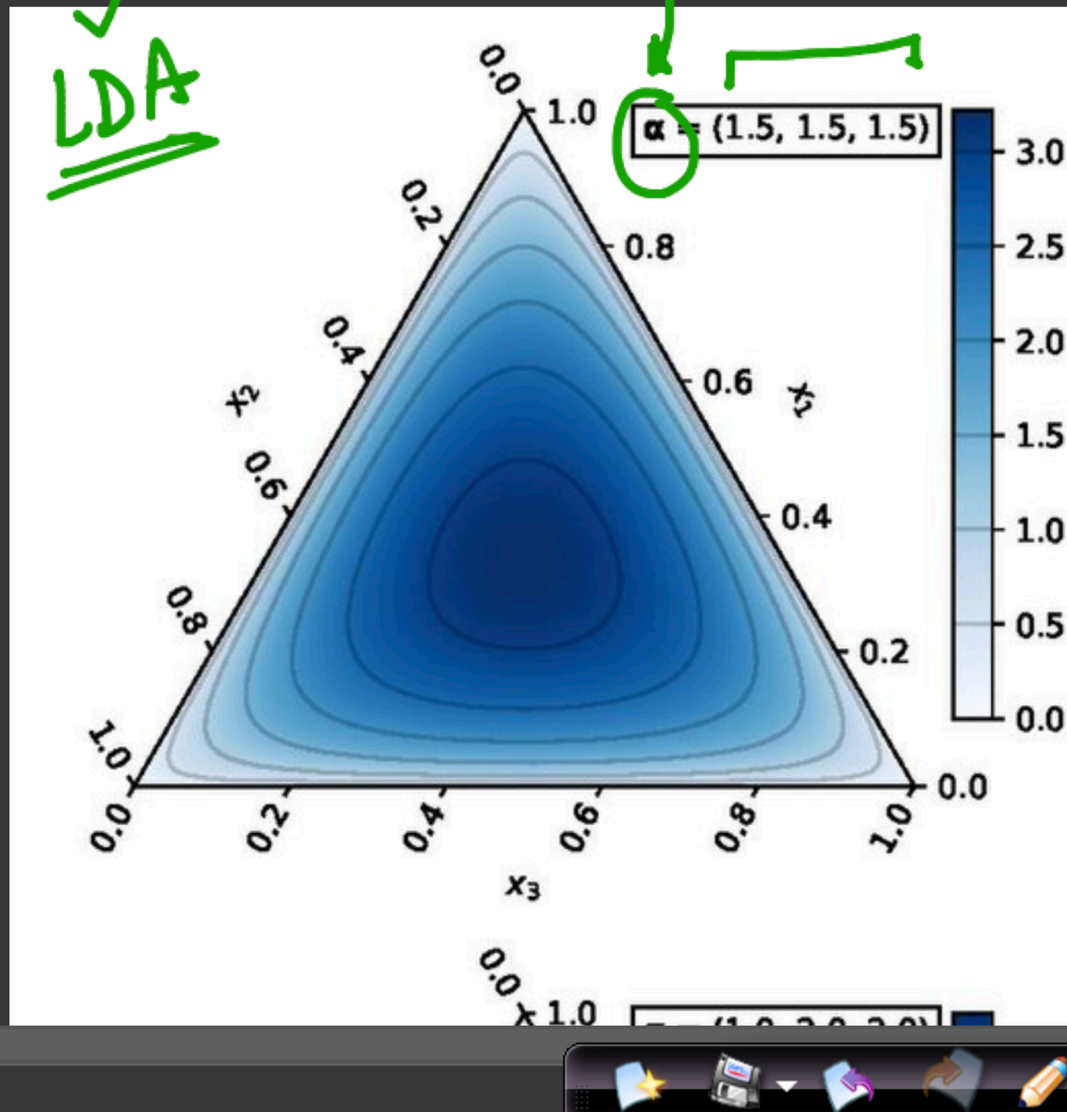


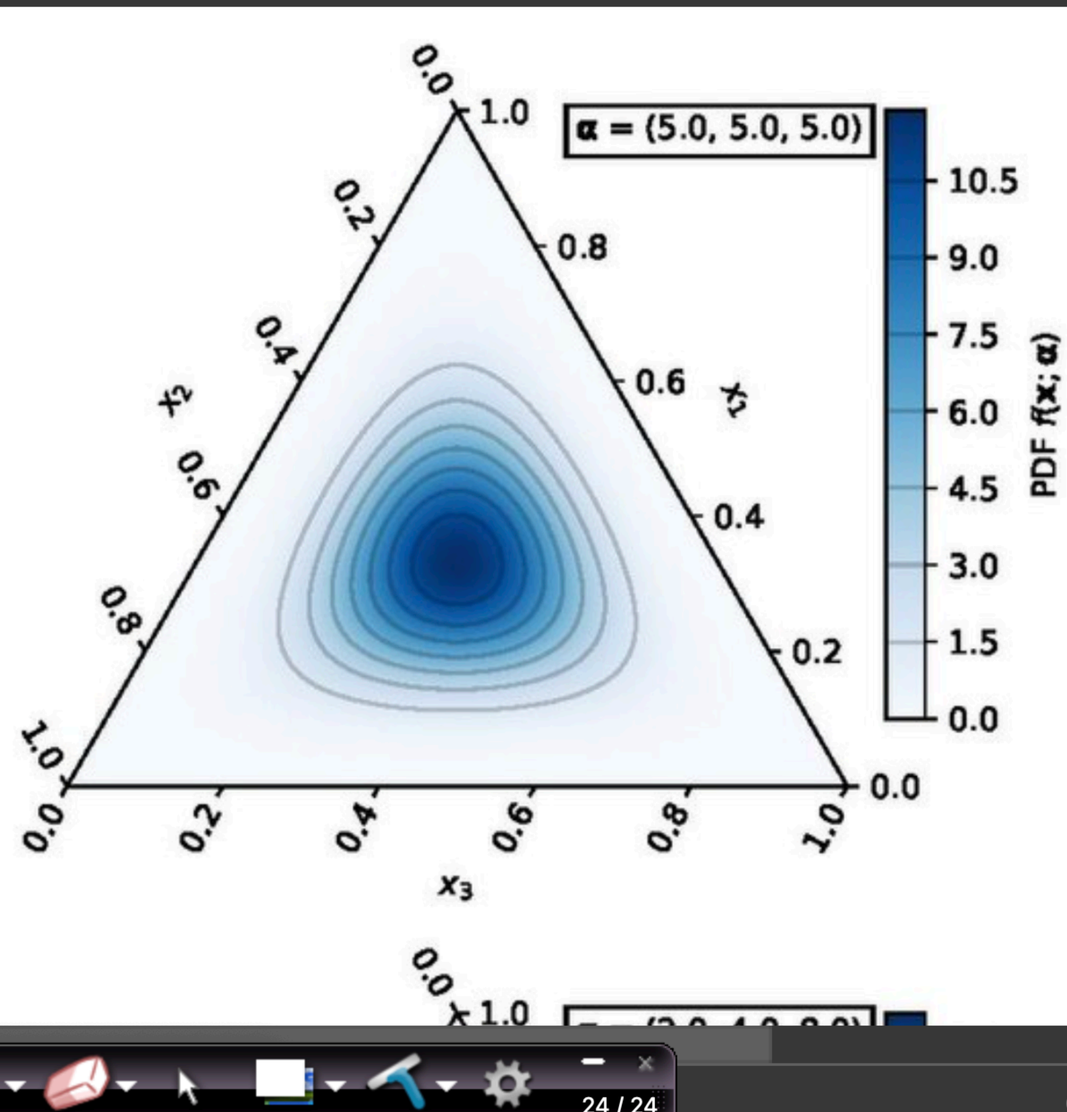
d_j



similar as b4 words if doc's are long..

Below are several visulaization of drichlet distribution at different α for 3 topics





LDA

Matrix-div
(optmzn)

find distbs
distb on topics
distb on words per topic

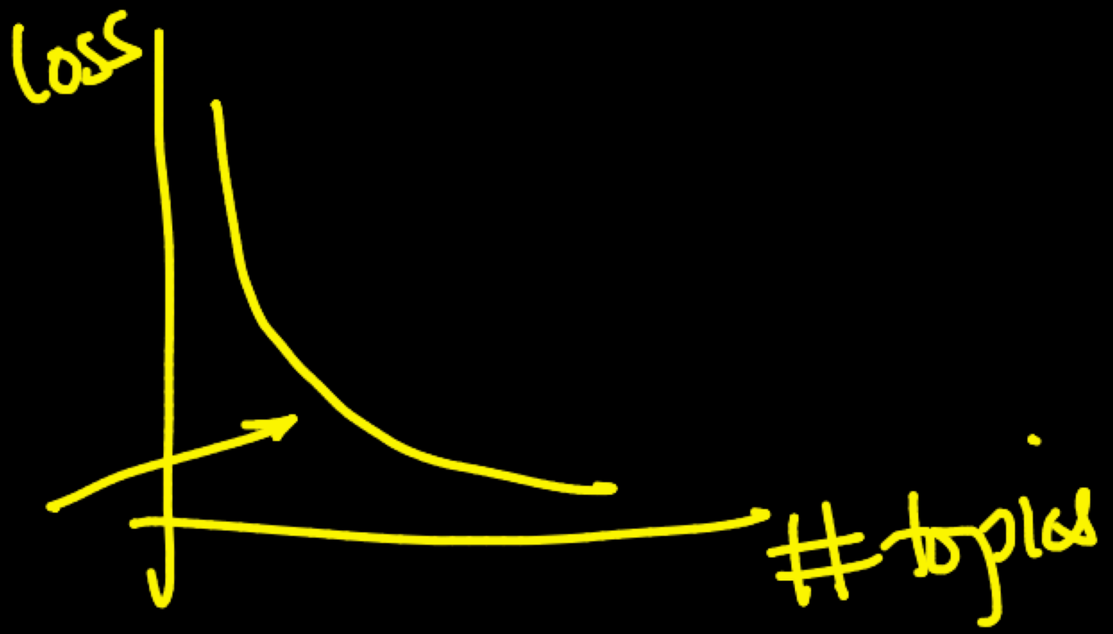
MLE

M-doc-Corpus

Q

#topics = 10 ?

LDA : NMF +
KL-div loss



$A_{n \times m} = B_{n \times d}\, C_{d \times m}$

Customer reviews

4.2 out of 5

8,894 global ratings

5 star	61%
4 star	21%
3 star	8%
2 star	3%
1 star	8%

How are ratings calculated?

By feature

Fingerprint reader	4.1
Battery life	4.0
Sheerness	4.0
Value for money	3.9
Picture quality	3.9
For gaming	3.5

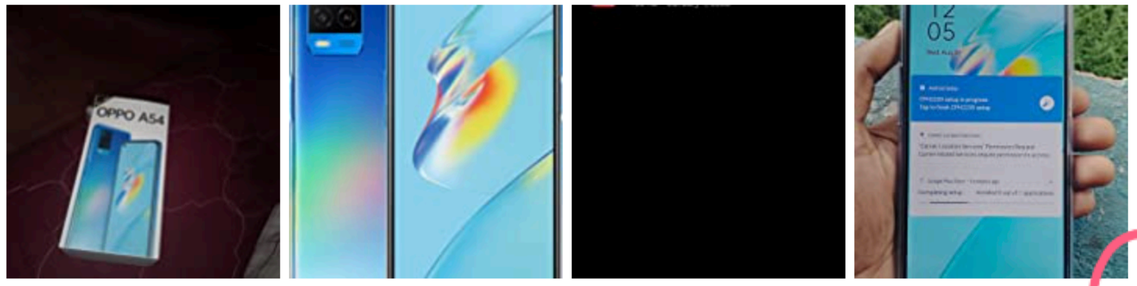
See less

Review this product

Share your thoughts with other customers

Write a product review

Reviews with images



See all customer images

Read reviews that mention

value for money | camera quality | battery life | waste of money | battery backup | hang hota | worst phone | phone ever | good battery | price range | quality is also | quality is very poor | good phone

Top reviews

Top reviews from India

sathish

Best for money

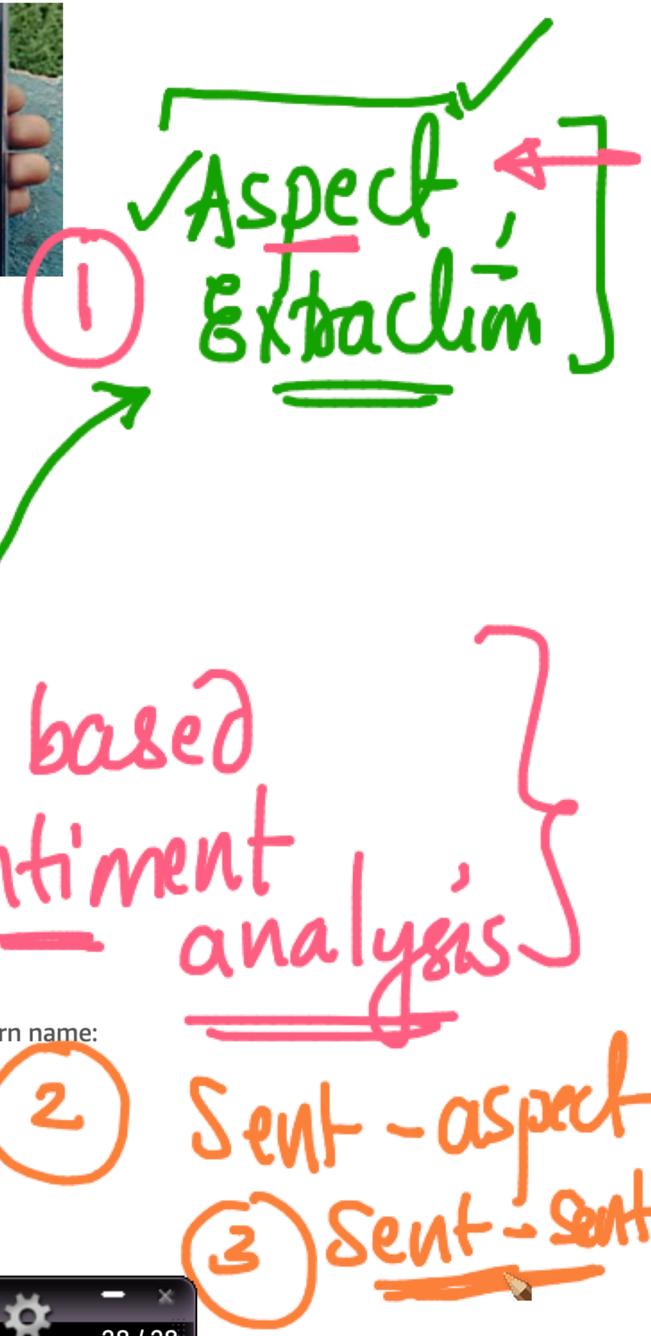
Reviewed in India on 3 November 2022

Colour: Starry Blue | Size name: 6GB RAM|128GB Storage | Style name: With Offer | Pattern name: Smartphone | Verified Purchase

Best quality mobile

Helpful | Report abuse

SIMA DEVI



Chrome File Edit View History Bookmarks Profiles Tab Window Help 10:42

PosTagging_Topic_Mode | nouns in english gramma | Gensim: Topic modelling | models.ldamodel – Later | Object Detection | Paper | Oppo A54 (Starry Blue, 6

colab.research.google.com/drive/1xaSfjSo1IJZkzfqMhGQL1tXI7jZliCEg#scrollTo=aiqoAUmTL1DJ

Update

+ Code + Text Connect

Link to dataset:
http://snap.stanford.edu/data/amazon/productGraph/categoryFiles/reviews_Musical_Instruments_5.json.gz

Dataset: Amazon Musical Instruments Reviews

Situation: Data Scientist @ Amazon

Today we will tackling a very common type of EDA/Insight discovery problem, where you've been given a lot of text, and you just have to figure out what is being talked about.

In amazon, where let's say you're a data scientist at, reviews are a very important source of information. Lot's of sellars use it to figure what are the issues customers are facing, how to fix them, how to improve products etc. Amazon wants sellers to improve too, as if thier sales improves, amazon's

29 / 29