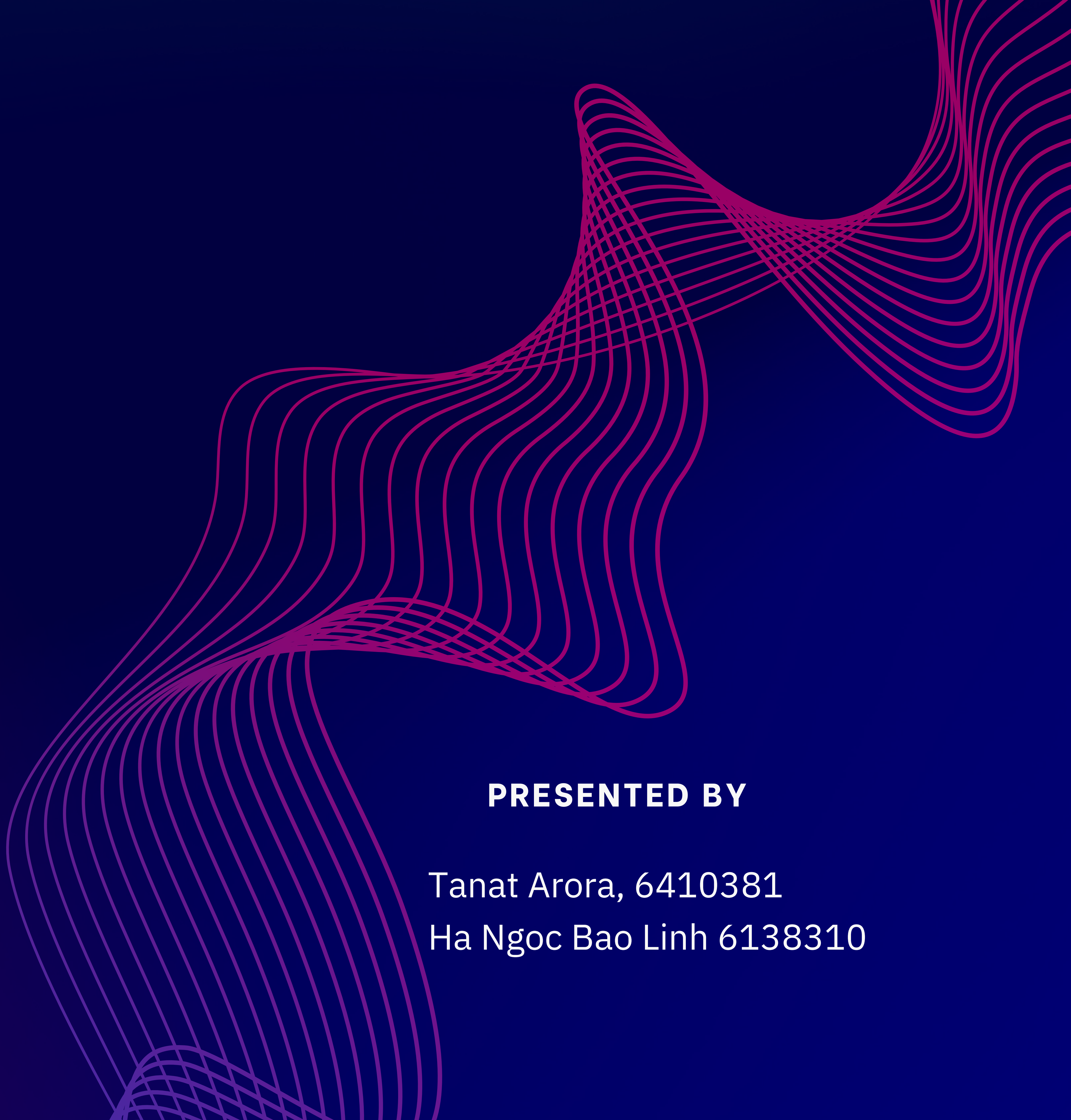
TOPIC MODELING



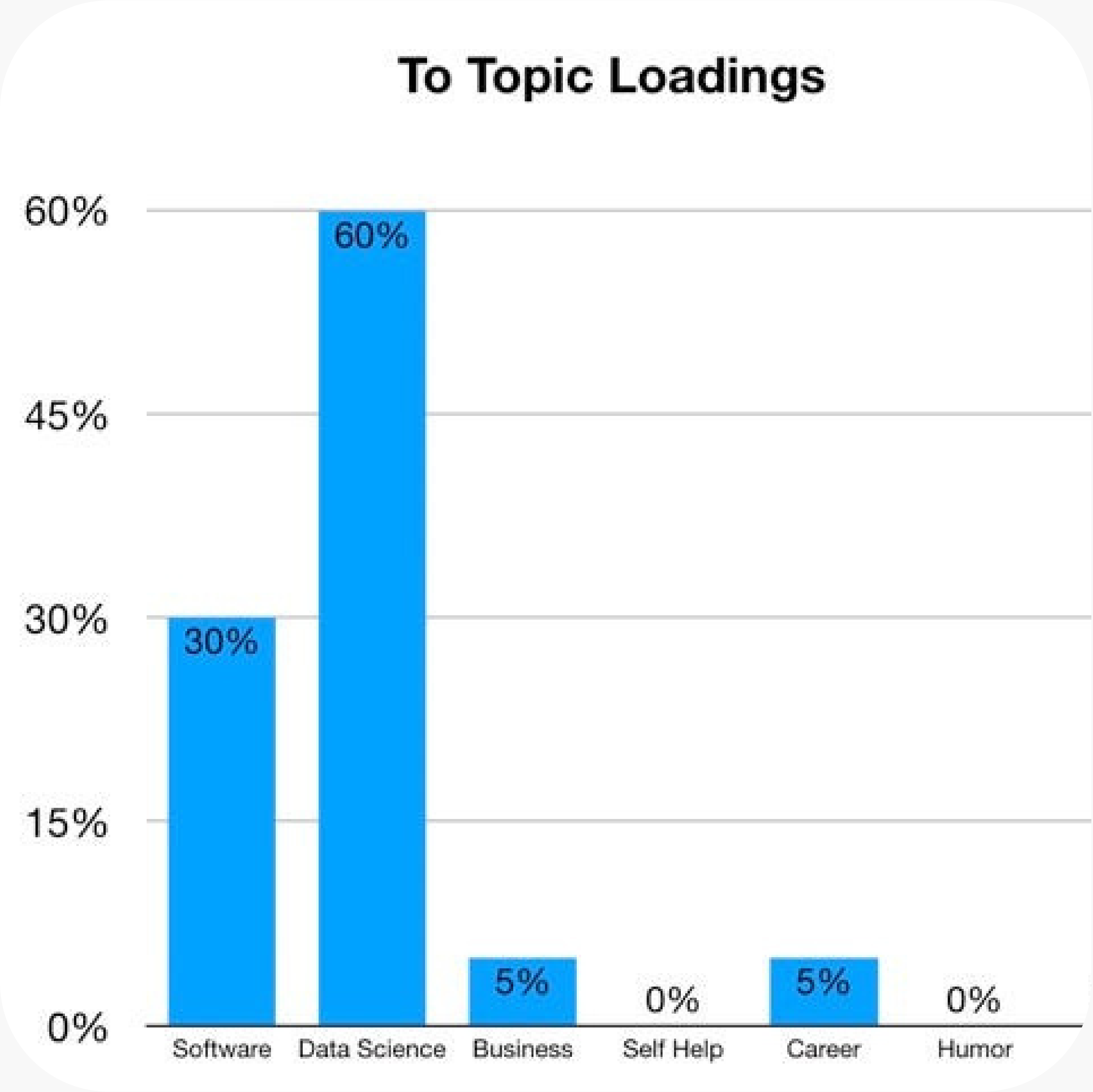
PRESENTED BY

Tanat Arora, 6410381
Ha Ngoc Bao Linh 6138310

Content

- What is Topic Modeling
- Usage of Topic Modeling
- Algorithms for Topic Modeling
- How do we evaluate the model

Topic Modeling



- Topic modeling is a text mining method that uses unsupervised machine learning to discover abstract ‘topics’ that exist within a collection of documents
- It uses statistical and probabilistic models to identify clusters or groups of similar words that reveal semantic structures in text
- It provides a way to organize and summarize textual data, making it easier to understand, explore, and navigate large document collections.

Why do we use Topic Modeling?

Discover hidden patterns

Topic modeling is used to discover hidden semantic structures in a text body.

Document organization and navigation

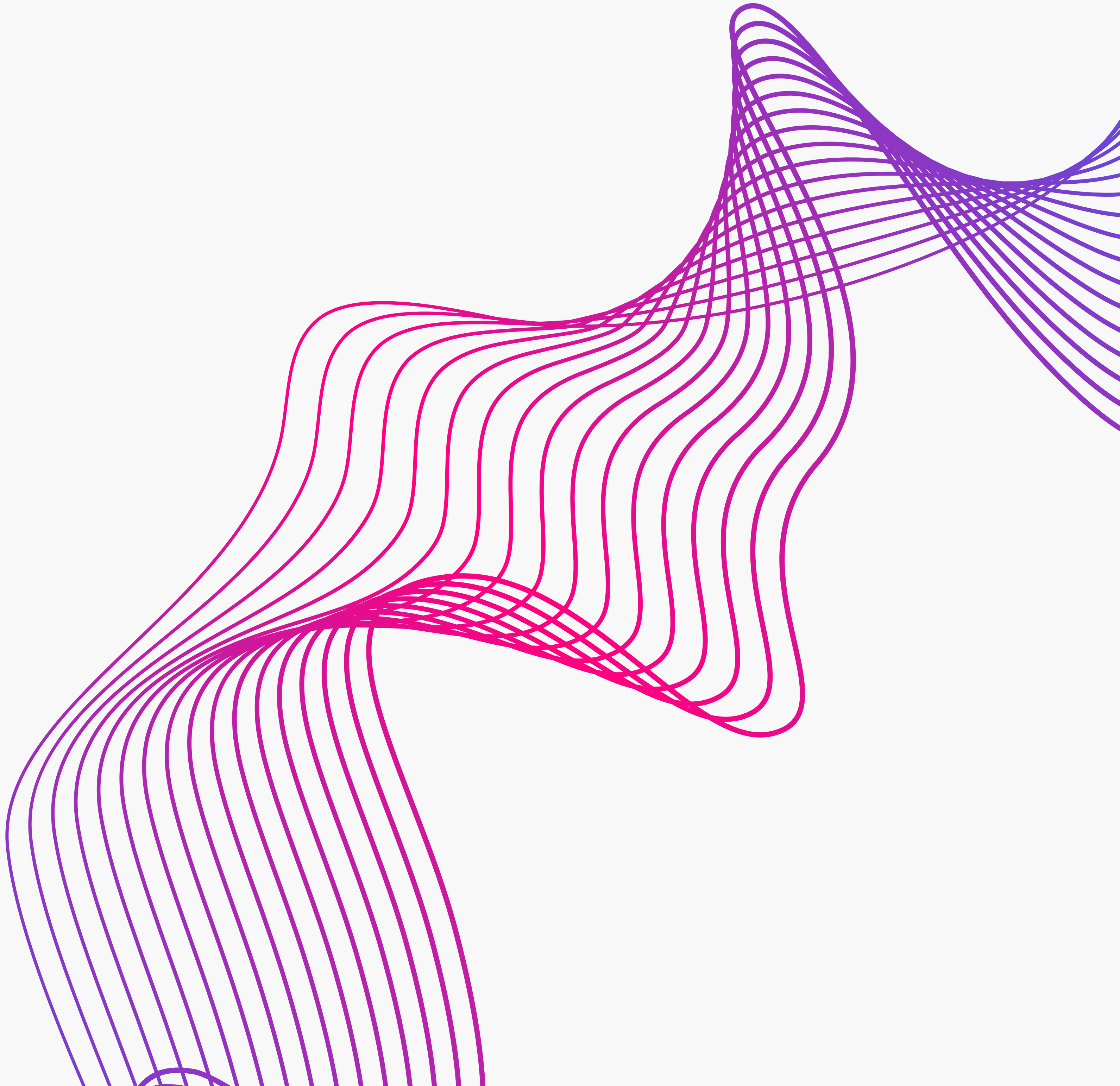
Topic modeling allows us to group similar documents together based on their content.

Summarization and information retrieval

It generate summaries or key phrases that represent the content of each document.

Content recommendation

Topic modeling can be used to recommend similar or related documents to users based on their interests.



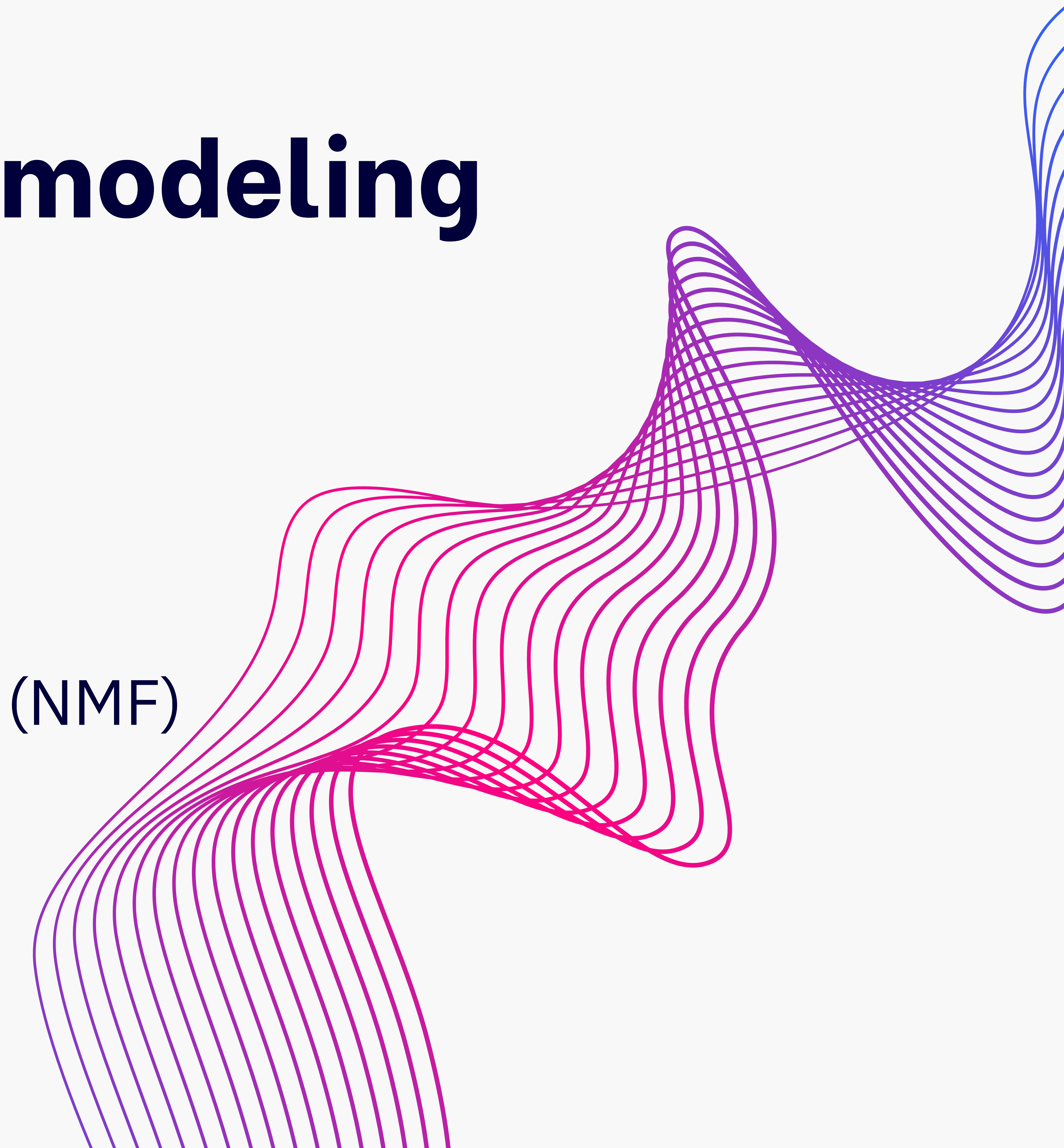
Algorithms in topic modeling

1

Latent Dirichlet Allocation (LDA)

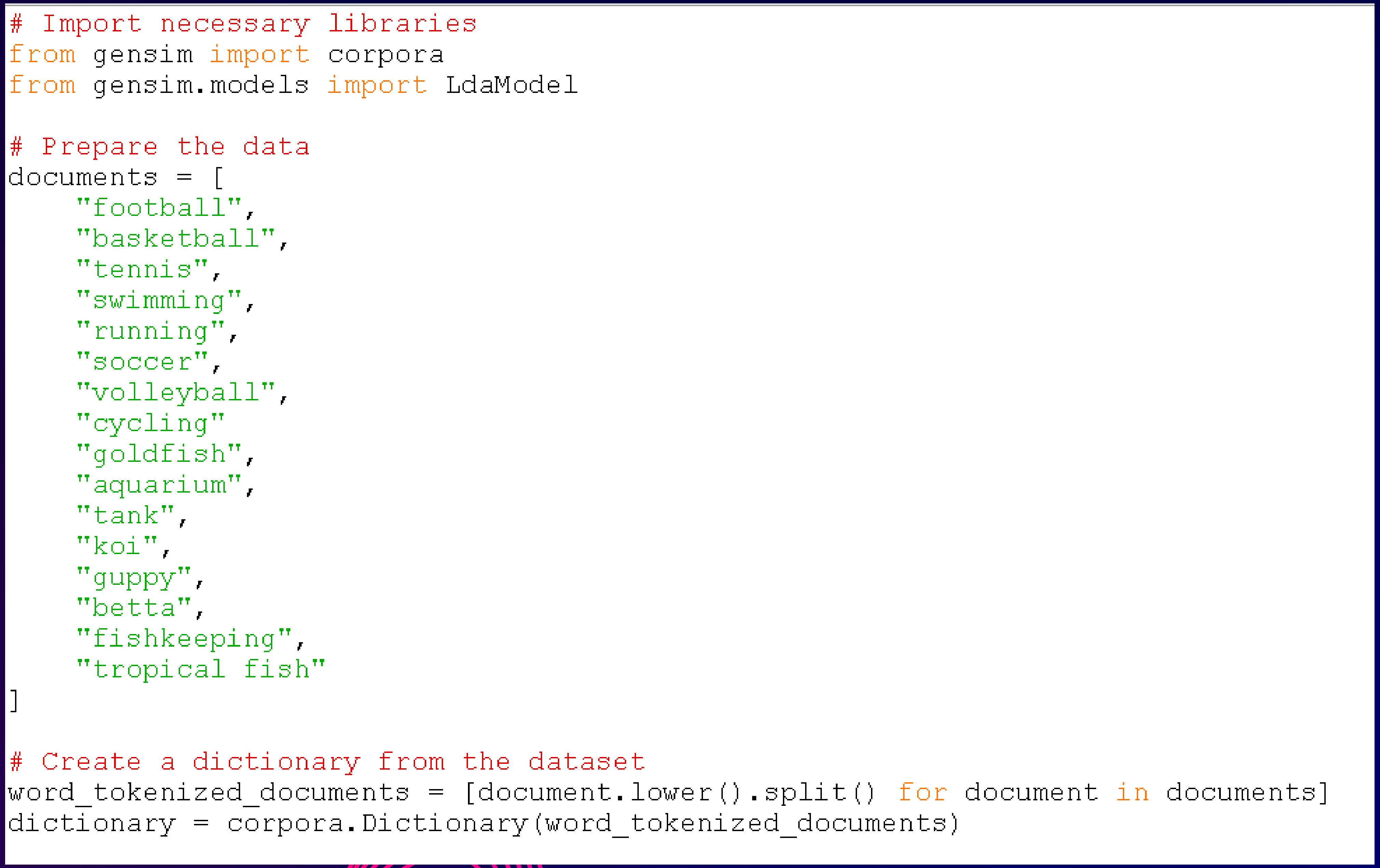
2

Non-negative Matrix Factorization (NMF)

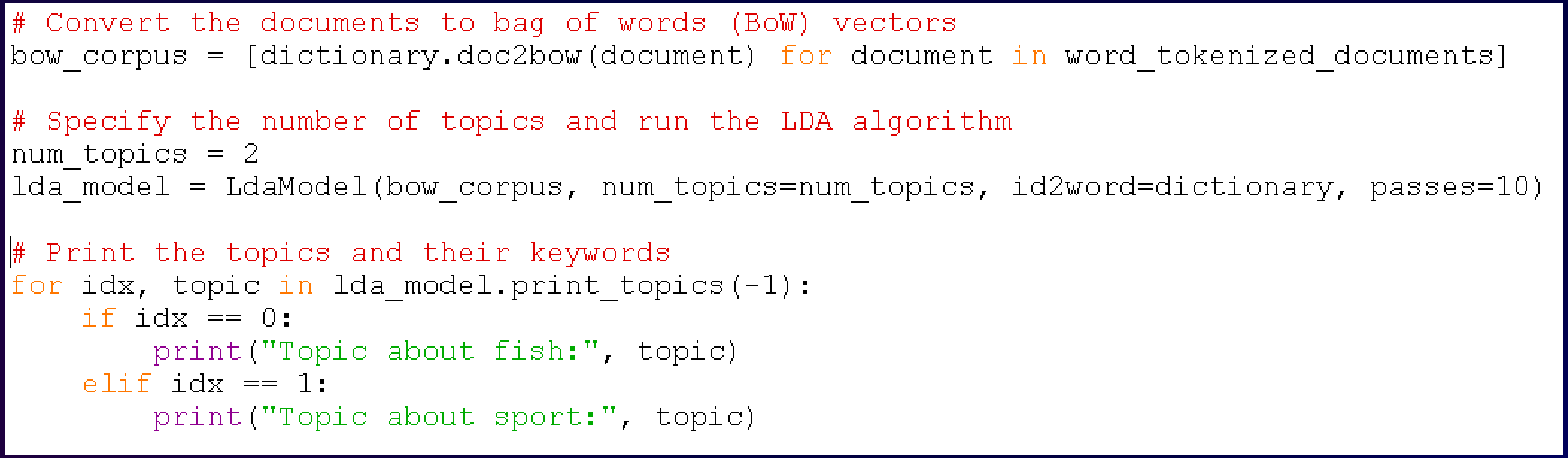


Latent Dirichlet Allocation (LDA)

LDA IS AN UNSUPERVISED LEARNING ALGORITHM USED TO ANALYZE AND GROUP TEXTS BASED ON HIDDEN TOPICS.

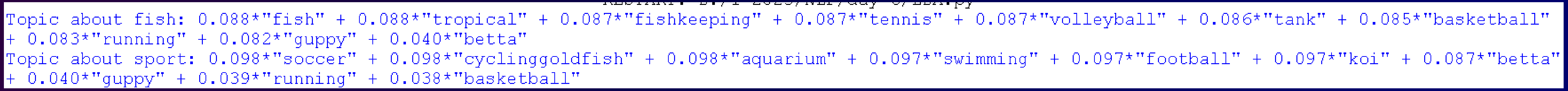


```
# Import necessary libraries
from gensim import corpora
from gensim.models import LdaModel

# Prepare the data
documents = [
    "football",
    "basketball",
    "tennis",
    "swimming",
    "running",
    "soccer",
    "volleyball",
    "cycling"
    "goldfish",
    "aquarium",
    "tank",
    "koi",
    "guppy",
    "betta",
    "fishkeeping",
    "tropical fish"
]

# Create a dictionary from the dataset
word_tokenized_documents = [document.lower().split() for document in documents]
dictionary = corpora.Dictionary(word_tokenized_documents)
```



```python
# Convert the documents to bag of words (BoW) vectors
bow_corpus = [dictionary.doc2bow(document) for document in word_tokenized_documents]

# Specify the number of topics and run the LDA algorithm
num_topics = 2
lda_model = LdaModel(bow_corpus, num_topics=num_topics, id2word=dictionary, passes=10)

# Print the topics and their keywords
for idx, topic in lda_model.print_topics(-1):
    if idx == 0:
        print("Topic about fish:", topic)
    elif idx == 1:
        print("Topic about sport:", topic)
```

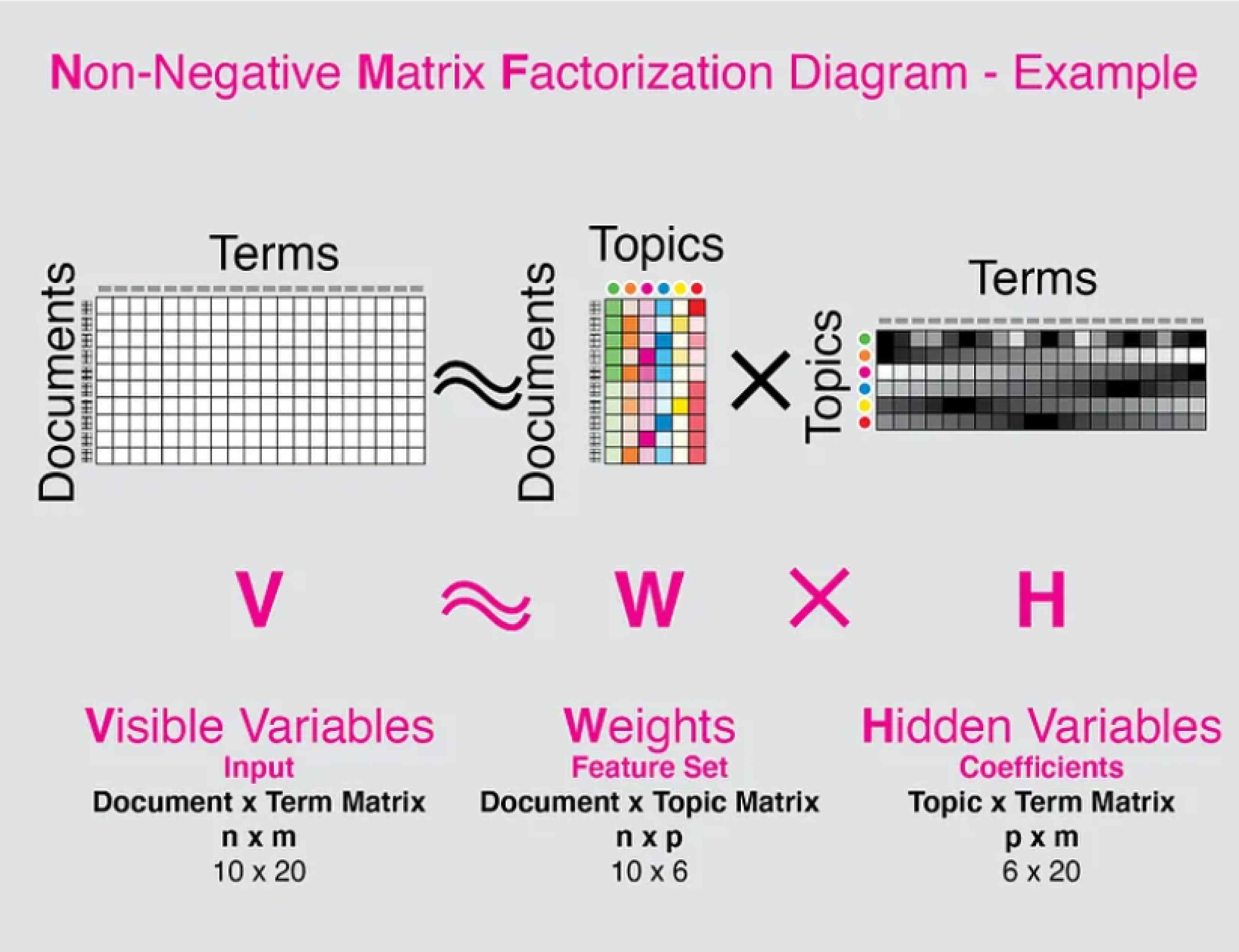
Output

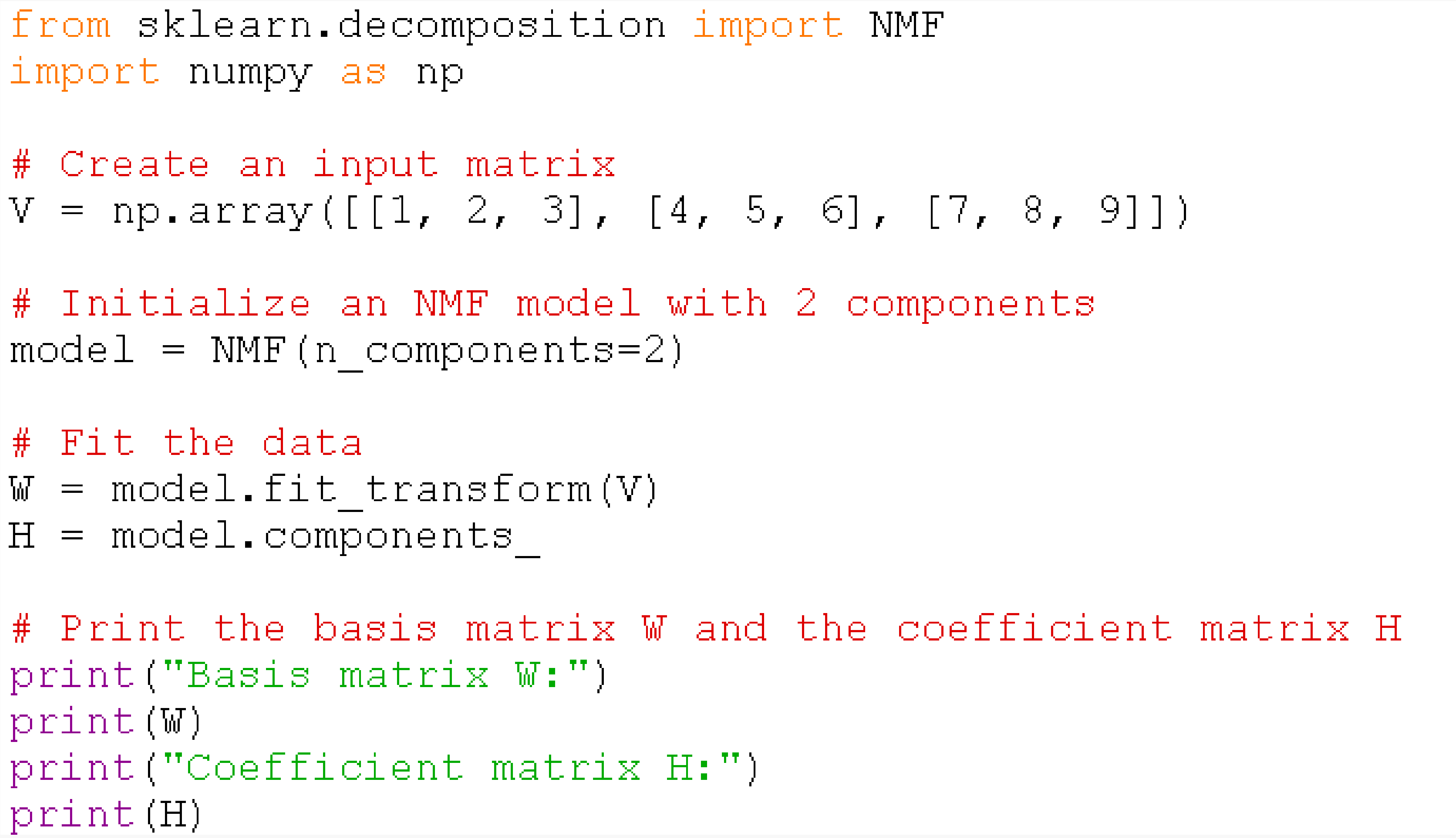


```
Topic about fish: 0.088*"fish" + 0.088*"tropical" + 0.087*"fishkeeping" + 0.087*"tennis" + 0.087*"volleyball" + 0.086*"tank" + 0.085*"basketball"
+ 0.083*"running" + 0.082*"guppy" + 0.040*"betta"
Topic about sport: 0.098*"soccer" + 0.098*"cyclinggoldfish" + 0.098*"aquarium" + 0.097*"swimming" + 0.097*"football" + 0.097*"koi" + 0.087*"betta"
+ 0.040*"guppy" + 0.039*"running" + 0.038*"basketball"
```

Non-negative Matrix Factorization (NMF)

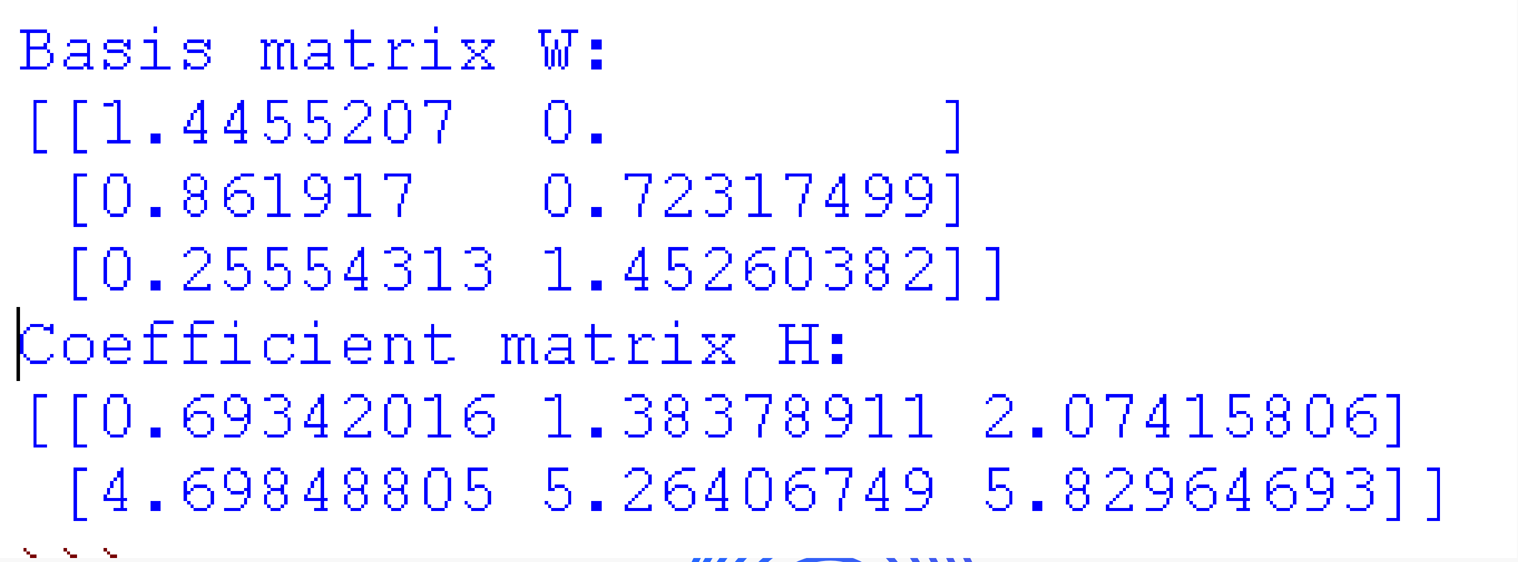
NMF is an algorithm that decomposes a non-negative matrix into two non-negative matrices, representing the underlying patterns in the data and their contributions to each data point.





```python
from sklearn.decomposition import NMF
import numpy as np

# Create an input matrix
V = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])

# Initialize an NMF model with 2 components
model = NMF(n_components=2)

# Fit the data
W = model.fit_transform(V)
H = model.components_

# Print the basis matrix W and the coefficient matrix H
print("Basis matrix W:")
print(W)
print("Coefficient matrix H:")
print(H)
```

Output



```
Basis matrix W:
[[1.4455207  0.        ]
 [0.861917   0.72317499]
 [0.25554313 1.45260382]]
Coefficient matrix H:
[[0.69342016 1.38378911 2.07415806]
 [4.69848805 5.26406749 5.82964693]]
```

How do we evaluate the model?

Perplexity

It measures how well the model predicts unseen data.

Coherence

Coherence measures the semantic interpretability of the topics generated by the model.

Topic diversity

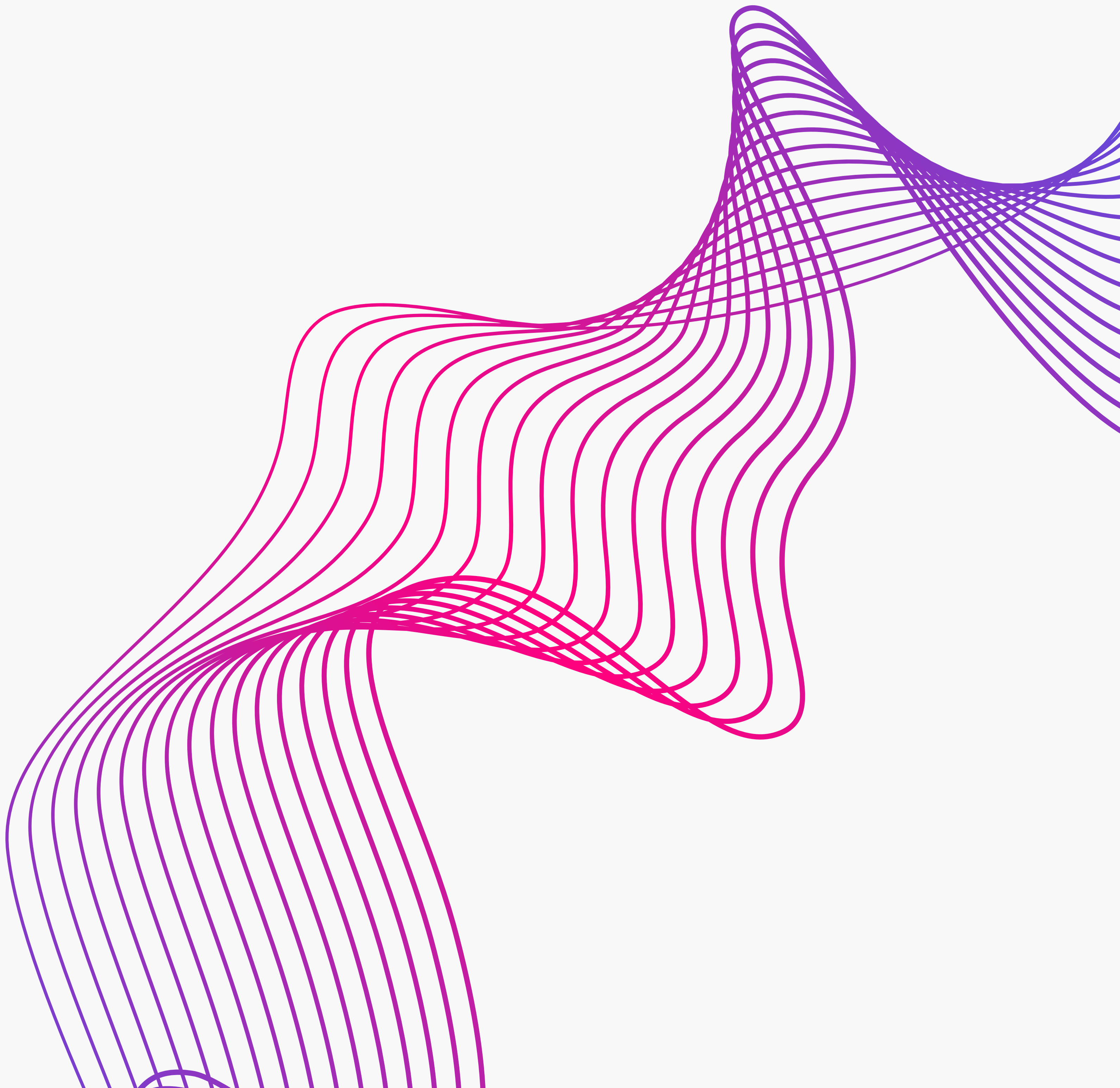
Topic diversity assesses the variety of topics generated by the model.

Domain-specific Metric

Additional metrics specific to the task can be used for evaluation.

Human Evaluation

It involves having domain experts or users assess the quality of the generated topics based on their expertise or subjective judgment.



Reference

- https://www.qualtrics.com/experience-management/research/topic-modeling/
- https://highdemandskills.com/topic-model-evaluation/#h2-2
- https://levity.ai/blog/what-is-topic-modeling
- https://mattilyra.github.io/2017/07/30/evaluating-topic-models.html
- https://towardsdatascience.com/nmf-a-visual-explainer-and-python-implementation-7ecdd73491f8