Natural Language Processing (NLP)

Rina BUOY

Source:

American University of Phnom Penh

Study Locally. Live Globally.

Introduction

- Natural Language Processing (NLP) is one of the hottest areas of artificial intelligence (AI) thanks to applications:
    - Text generators
    - Chatbots
    - Text-to-image
- NLP - building machines that can manipulate human language
    - computational linguistics
- NLP is an engineering discipline that seeks to build technology to accomplish useful tasks.

NLP vs NLU vs NLG

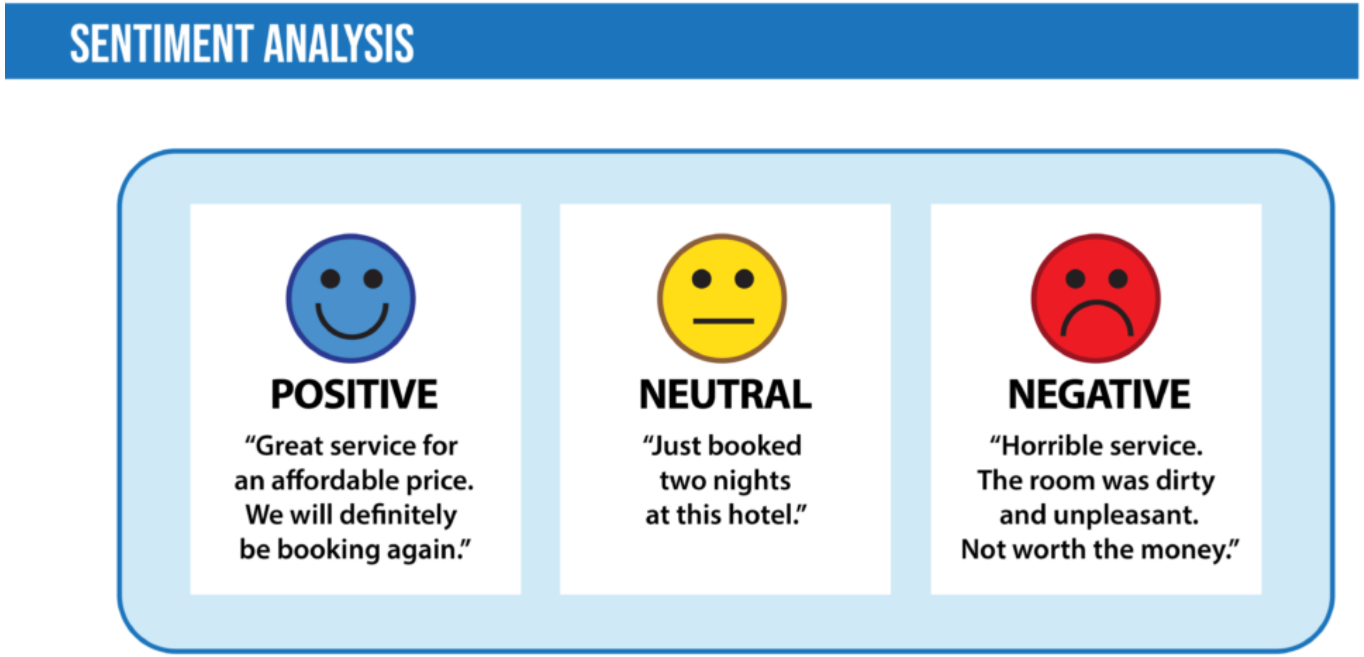
- NLP can be divided into :
  - natural language understanding (NLU), which focuses on semantic analysis or determining the intended meaning of text
  - natural language generation (NLG), which focuses on text generation by a machine.
- NLP is separate from — but often used in conjunction with — speech recognition, which seeks to parse spoken language into words, turning sound into text and vice versa.

Why Does Natural Language Processing (NLP) Matter?

- NLP is an integral part of everyday life. Language technology is applied to diverse fields like:
  - retailing (for instance, in customer service chatbots)
  - medicine (interpreting or summarizing electronic health records).
- Google uses NLP to improve its search engine results, and social networks like Facebook use it to detect and filter hate speech.
- Current systems are prone to bias and incoherence, and occasionally behave erratically.
- Despite the challenges, machine learning engineers have many opportunities to apply NLP in ways that are ever more central to a functioning society.

Applications

- NLP is used for a wide variety of language-related tasks, including answering questions, classifying text in a variety of ways, and conversing with users.
  - **Sentiment analysis** is the process of classifying the emotional intent of text.



SENTIMENT ANALYSIS

POSITIVE

"Great service for an affordable price. We will definitely be booking again."

NEUTRAL

"Just booked two nights at this hotel."

NEGATIVE

"Horrible service. The room was dirty and unpleasant. Not worth the money."

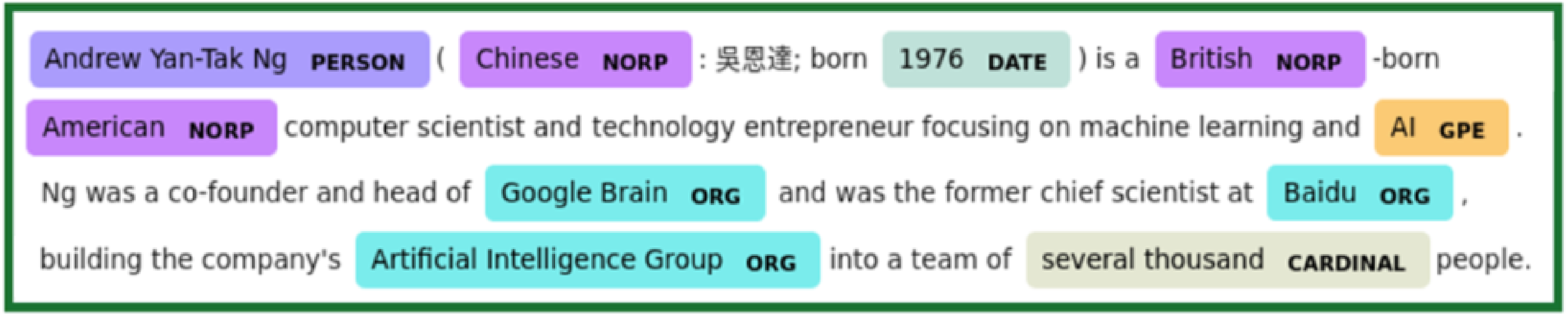
Given text, sentiment analysis classifies its emotional quality.

Applications

- **Toxicity classification** is a branch of sentiment analysis where the aim is not just to classify hostile intent but also to classify particular categories such as threats, insults, obscenities, and hatred towards certain identities.
- **Machine translation** automates translation between different languages. The input to such a model is text in a specified source language, and the output is the text in a specified target language.
- **Named entity recognition** aims to extract entities in a piece of text into predefined categories such as personal names, organizations, locations, and quantities.

Applications

NAMED ENTITY RECOGNITION (NER) TAGGING



Andrew Yan-Tak Ng PERSON (Chinese NORP : 吳恩達; born 1976 DATE) is a British NORP -born American NORP computer scientist and technology entrepreneur focusing on machine learning and AI GPE . Ng was a co-founder and head of Google Brain ORG and was the former chief scientist at Baidu ORG , building the company's Artificial Intelligence Group ORG into a team of several thousand CARDINAL people.

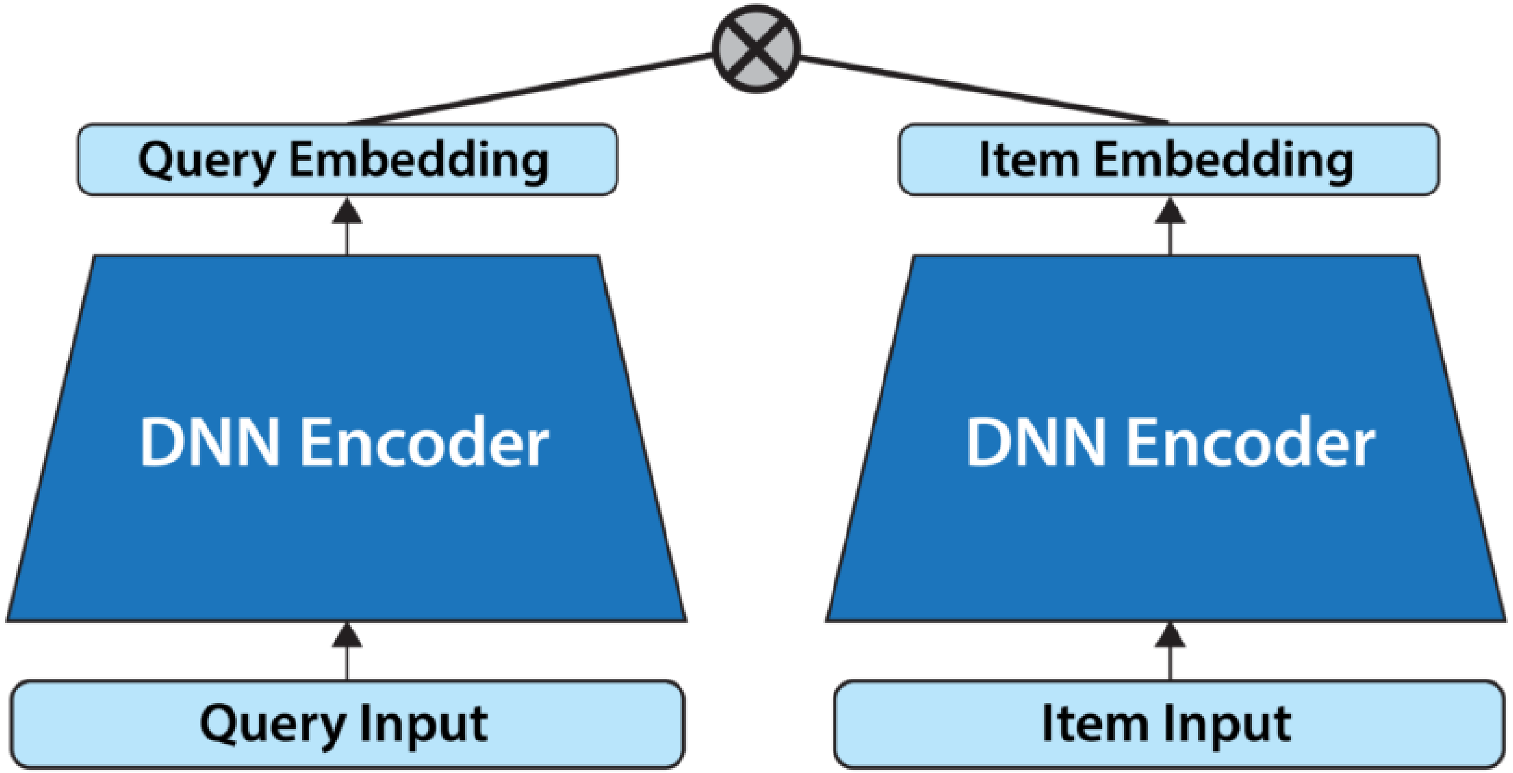
spaCy named entity recognition tagging of the first paragraph of Andrew Ng's Wikipedia page. "NORP" stands for nationalities or religious or political groups. Note that spaCy incorrectly labels "AI" as "GPE," for geopolitical entity.

Applications

- **Spam detection** is a prevalent binary classification problem in NLP, where the purpose is to classify emails as either spam or not.
- **Grammatical error correction** models encode grammatical rules to correct the grammar within text.
- **Topic modeling** is an unsupervised text mining task that takes a corpus of documents and discovers abstract topics within that corpus.
- **Text generation**, more formally known as natural language generation (NLG), produces text that’s similar to human-written text.
- **Information retrieval** finds the documents that are most relevant to a query.

Applications

INFORMATION RETRIEVAL



A two-tower network creates a representation of an input query and a group of documents (or items) through two separate networks. Then it compares the representation of the query with that of the documents to find documents that are most relevant to the query.

Applications

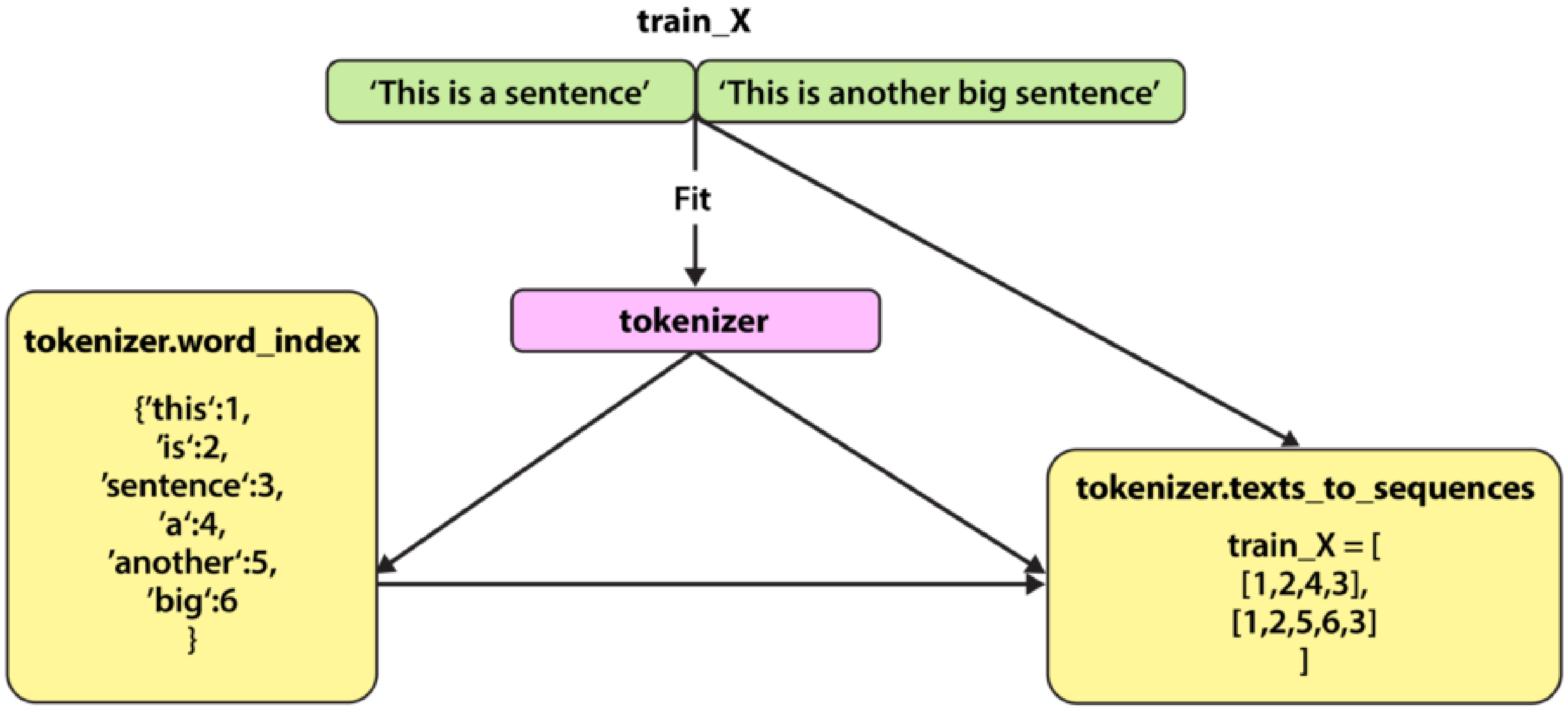
- **Summarization** is the task of shortening text to highlight the most relevant information.
- **Question answering** deals with answering questions posed by humans in a natural language.

How Does Natural Language Processing (NLP) Work?

- NLP architectures use various methods for data preprocessing, feature extraction, and modeling.
  - **Data preprocessing:** Before a model processes text for a specific task, the text often needs to be preprocessed to improve model performance or to turn words and characters into a format the model can understand.
    - **Stemming and lemmatization**: Stemming is an informal process of converting words to their base forms using heuristic rules. Lemmatization is a more formal way to find roots by analyzing a word's morphology using vocabulary from a dictionary. Stemming and lemmatization are provided by libraries like spaCy and NLTK.
    - **Sentence segmentation** breaks a large piece of text into linguistically meaningful sentence units.
    - **Stop word removal** aims to remove the most commonly occurring words that don't add much information to the text.
    - **Tokenization** splits text into individual words and word fragments.

How Does Natural Language Processing (NLP) Work?

TOKENIZERS



```
{'this':1,
'is':2,
'sentence':3,
'a':4,
'another':5,
'big':6
}
```

```
train_X = [
[1,2,4,3],
[1,2,5,6,3]
]
```

Given a corpus of documents, a tokenizer maps every word to an index. Then it can translate any document into a sequence of numbers.

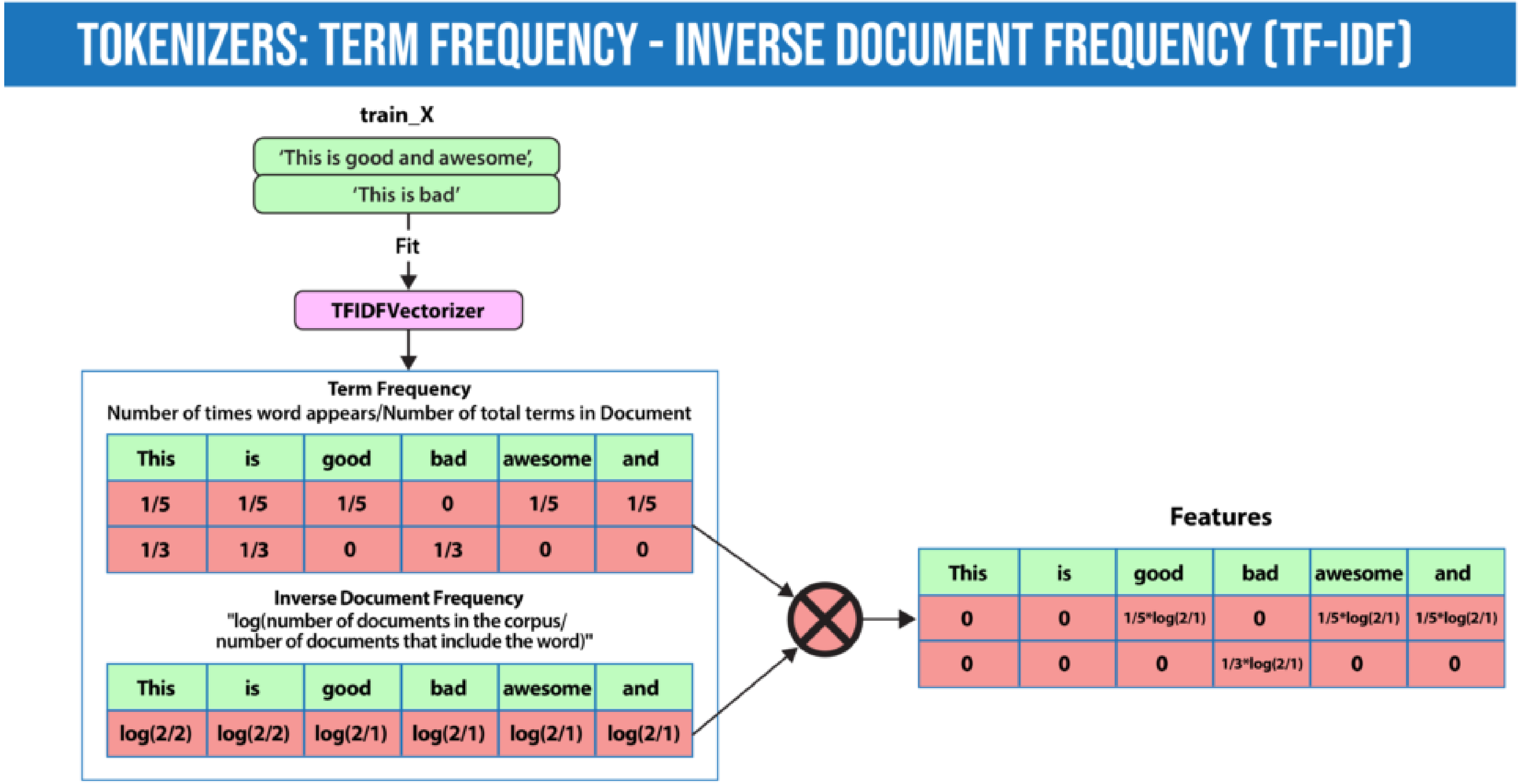
How Does Natural Language Processing (NLP) Work?

- Feature extraction: generally numbers that describe a document in relation to the corpus that contains it – created by either Bag-of-Words, TF-IDF, or generic feature engineering such as document length, word polarity, and metadata (for instance, if the text has associated tags or scores).
- More recent techniques include Word2Vec, GLoVE, and learning the features during the training process of a neural network.
- **Bag-of-Words:** Bag-of-Words counts the number of times each word or n-gram (combination of n words) appears in a document.

How Does Natural Language Processing (NLP) Work?

- **TF-IDF:** In Bag-of-Words, we count the occurrence of each word or n-gram in a document. In contrast, with TF-IDF, we weight each word by its importance. To evaluate a word’s significance, we consider two things:
  - **Term Frequency:** How important is the word in the document?
    - *TF(word in a document)= Number of occurrences of that word in document / Number of words in document*
  - **Inverse Document Frequency:** How important is the term in the whole corpus?
    - *IDF(word in a corpus)=log(number of documents in the corpus / number of documents that include the word)*

How Does Natural Language Processing (NLP) Work?



TOKENIZERS: TERM FREQUENCY - INVERSE DOCUMENT FREQUENCY (TF-IDF)

train_X

'This is good and awesome',

'This is bad'

Fit

TFIDFVectorizer

Term Frequency

Number of times word appears/Number of total terms in Document

This	is	good	bad	awesome	and
1/5	1/5	1/5	0	1/5	1/5
1/3	1/3	0	1/3	0	0

Inverse Document Frequency

"log(number of documents in the corpus/ number of documents that include the word)"

This	is	good	bad	awesome	and
log(2/2)	log(2/2)	log(2/1)	log(2/1)	log(2/1)	log(2/1)

Features

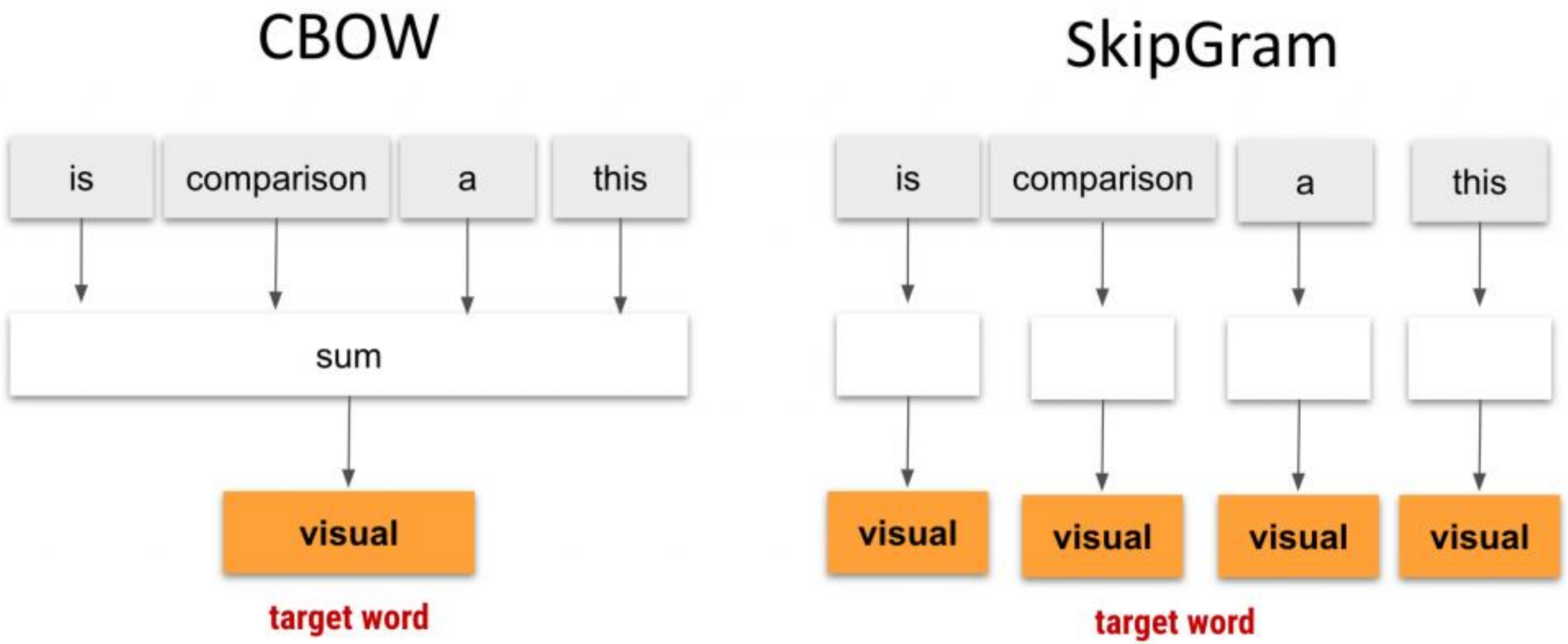
This	is	good	bad	awesome	and
0	0	1/5*log(2/1)	0	1/5*log(2/1)	1/5*log(2/1)
0	0	0	1/3*log(2/1)	0	0

TF-IDF creates features for each document based on how often each word shows up in a document versus the entire corpus.

How Does Natural Language Processing (NLP) Work?

- **Word2Vec**, introduced in 2013, uses a vanilla neural network to learn high-dimensional word embeddings from raw text.
- It comes in two variations:
  - Skip-Gram, in which we try to predict surrounding words given a target word, and
  - Continuous Bag-of-Words (CBOW), which tries to predict the target word from surrounding words.
- After discarding the final layer after training, these models take a word as input and output a word embedding that can be used as an input to many NLP tasks.
- Embeddings from Word2Vec capture context. If particular words appear in similar contexts, their embeddings will be similar.

Word2Vec



By: Kavita Ganesan

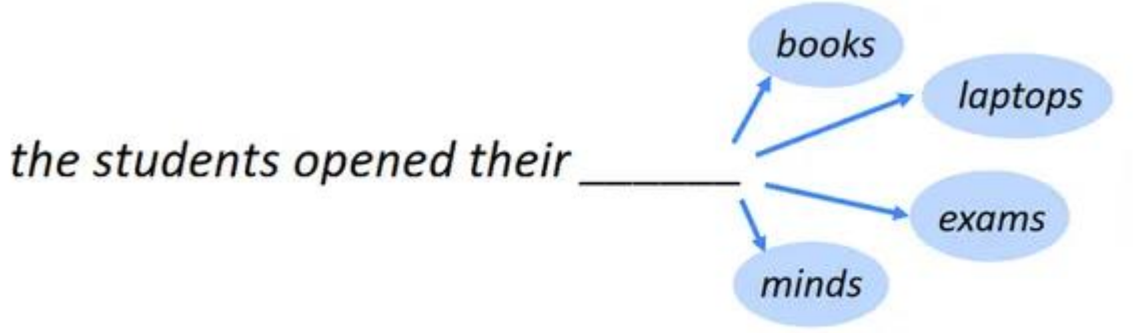
This is a visual comparison

How Does Natural Language Processing (NLP) Work?

- **Modeling:** After data is preprocessed, it is fed into an NLP architecture that models the data to accomplish a variety of tasks.
- **Numerical features** extracted by the techniques described above can be fed into various models depending on the task at hand.
    - Logistic regression, naive Bayes, decision trees, or gradient boosted trees.
    - Deep neural networks typically work without using extracted features, although we can still use TF-IDF or Bag-of-Words features as an input.
- **Language Models**: In very basic terms, the objective of a language model is to predict the next word when given a stream of input words.

How Does Natural Language Processing (NLP) Work?

- Deep learning is also used to create such language models.
- Deep-learning models take as input a word embedding and, at each time state, return the probability distribution of the next word as the probability for every word in the dictionary.



- Pre-trained language models learn the structure of a particular language by processing a large corpus, such as Wikipedia. They can then be fine-tuned for a particular task.

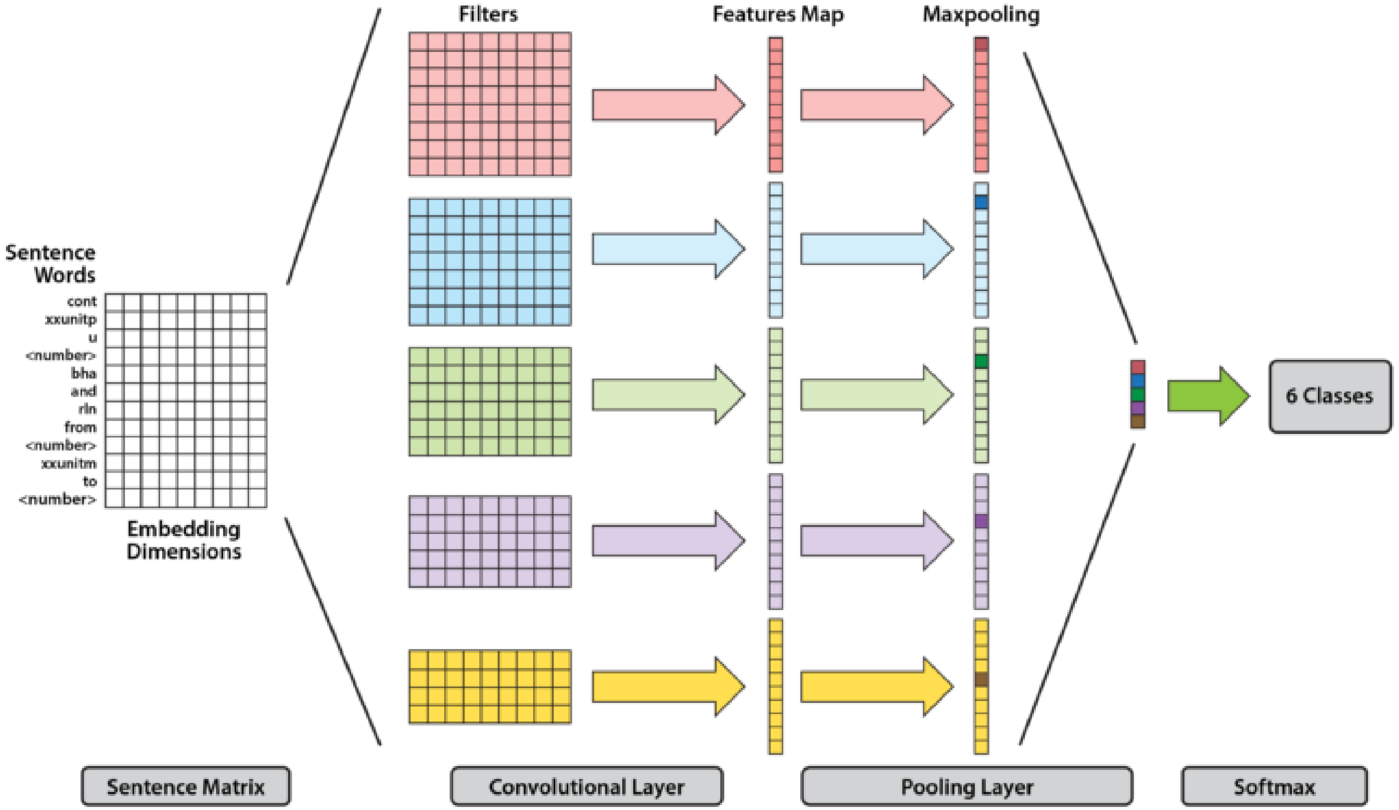
NLP Techniques

- Traditional:
  - Logistic regression
  - Naive Bayes
  - Decision trees
  - …

- Deep Learning:
  - Convolutional Neural Network (CNN)
  - Recurrent Neural Network (RNN)
  - Autoencoders
  - …

NLP Techniques

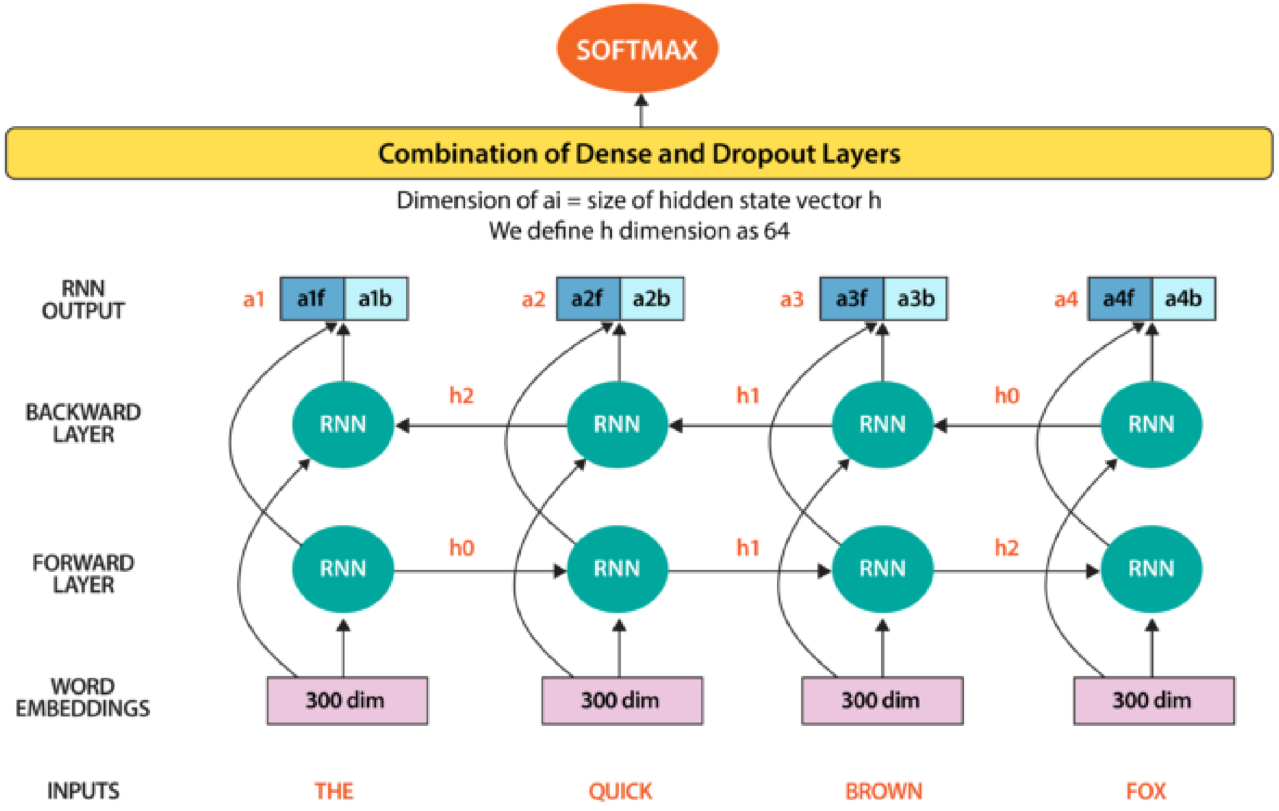
CONVOLUTIONAL NEURAL NETWORK-BASED TEXT CLASSIFICATION NETWORK



Given a sentence, a convolutional neural network uses convolutional layers to refine representations of input words, before combining them to render a classification.

NLP Techniques

RECURRENT NEURAL NETWORK



A bidirectional recurrent neural network processes the input both forward and backward to improve the representations it produces.

Libraries, And Frameworks

- **Natural Language Toolkit (NLTK)** is one of the first NLP libraries written in Python.
- **spaCy** is one of the most versatile open source NLP libraries. It supports more than 66 languages. spaCy also provides pre-trained word vectors and implements many popular models like BERT.
- **Deep Learning libraries:** Popular deep learning libraries include TensorFlow and PyTorch, which make it easier to create models with features like automatic differentiation.
- **Hugging Face** offers open-source implementations and weights of over 135 state-of-the-art models.
- **Gensim** provides vector space modeling and topic modeling algorithms.
- Python is the most-used programming language to tackle NLP tasks.

Controversies Surrounding NLP

- **Stochastic parrots:** The authors point out that huge, uncurated datasets scraped from the web are bound to include social biases and other undesirable information, and models that are trained on them will absorb these flaws.
- **Coherence versus sentience:** Recently, a Google engineer tasked with evaluating the LaMDA language model was so impressed by the quality of its chat output that he believed it to be sentient. The fallacy of attributing human-like intelligence to AI dates back to some of the earliest NLP experiments.
- **Environmental impact:** Large language models require a lot of energy during both training and inference. One study estimated that training a single large language model can emit five times as much carbon dioxide as a single automobile over its operational lifespan.

Controversies Surrounding NLP

- **High cost leaves out non-corporate researchers:** The computational requirements needed to train or deploy large language models are too expensive for many small companies.
- **Black box:** When a deep learning model renders an output, it’s difficult or impossible to know why it generated that particular result. While traditional models like logistic regression enable engineers to examine the impact on the output of individual features, neural network methods in natural language processing are essentially black boxes.

Conclusions

- NLP is one of the fast-growing research domains in AI, with applications that involve tasks including translation, summarization, text generation, and sentiment analysis.
- Businesses use NLP to power a growing number of applications, both internal — like detecting insurance fraud, determining customer sentiment, and optimizing aircraft maintenance — and customer-facing, like Google Translate.
- NLP is an exciting and rewarding discipline, and has potential to profoundly impact the world in many positive ways.
- Unfortunately, NLP is also the focus of several controversies, and understanding them is also part of being a responsible practitioner.

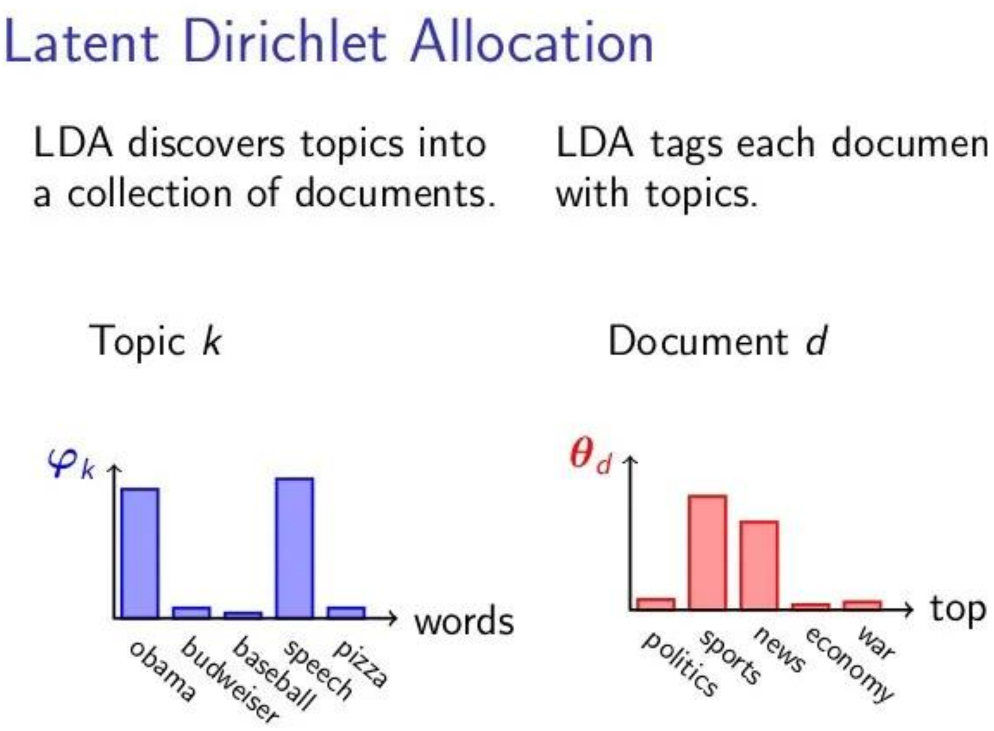
Advanced Topics

Topic Modelling (source)

Latent Dirichlet Allocation

- Latent Dirichlet Allocation is a statistical technique:
    - for dimensionality reduction
    - topic modeling to automatically summarise text or find hidden associations automatically from data.
- LDA is also one of the most powerful techniques in text mining for data mining, data discovery, and find relationships among data and text documents.
- An unsupervised algorithm with the aim of describing a set of observations in the dataset (preferably text) as a mixture of distinct topics or categories.
    - The number of topics to be discovered is to be specified by the user to the algorithm.
    - The topics are assumed to be shared by all the documents within the text corpus.

Latent Dirichlet Allocation



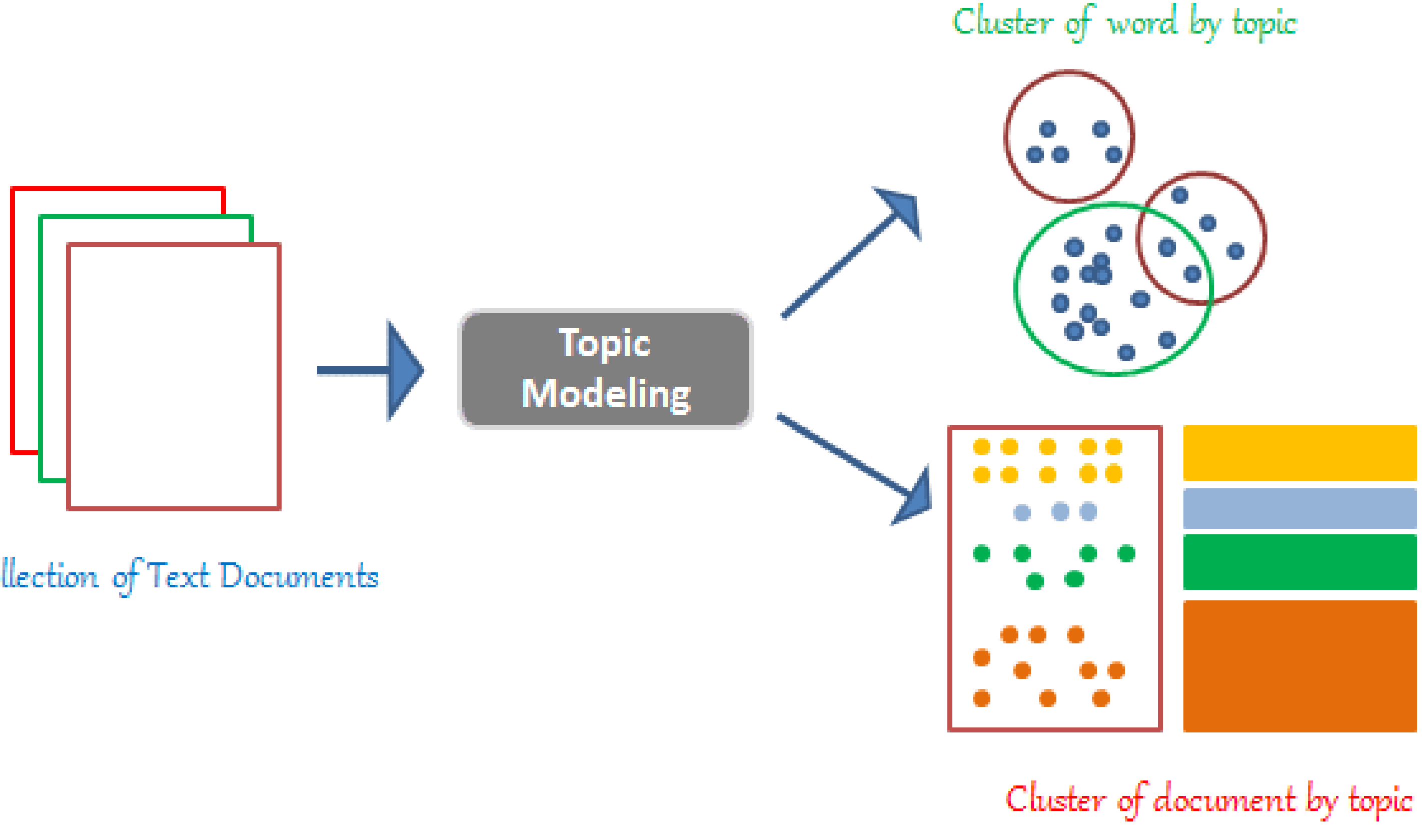
Latent Dirichlet Allocation

LDA discovers topics into a collection of documents.

LDA tags each documen with topics.

Topic k

Document d



Cluster of word by topic

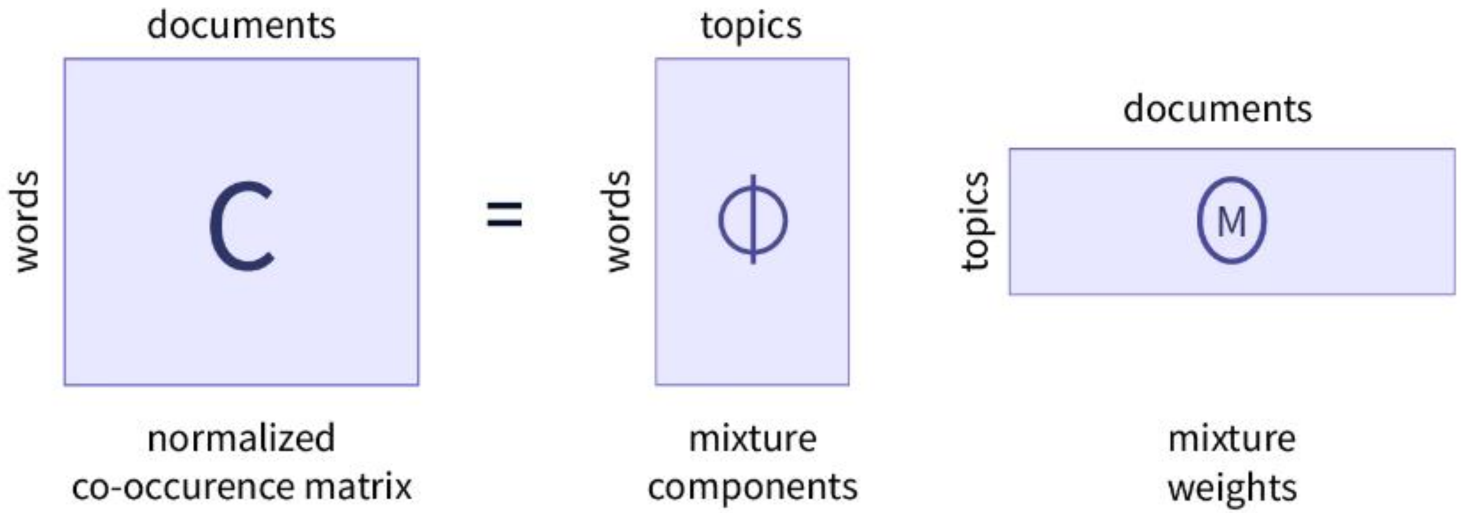
Topic Modeling

Collection of Text Documents

Cluster of document by topic

How does Latent Dirichlet Allocation Work?

- LDA works in two parts:
  - Finding words that belong to a document, which is already seen from the corpus of documents.
  - Finding the words that belong to a topic or the probability of words belonging to a topic, that are calculated from the Latent Dirichlet Allocation model.



Hyperparameters in Latent Dirichlet Allocation

- α: Document topic prior, is a prior estimate on topic probability, controls the number of topics expected in the document.
  - Low value of α is used to imply that fewer number of topics in the mix is expected and a higher value implies that one would expect the documents to have higher number topics in the mix.
  - Specifies the word distribution (the average frequency that each topic within a given document occurs).
- β: Topic-word prior, controls the distribution of words per topic.
  - Is a collection of k topics where each topic is given a probability distribution over the vocabulary used in a document corpus.
  - The topics will likely have fewer words at lower values of β and at higher values, topics will likely have more words.
- K: The number of topics to be discovered/considered.

Data Preprocessing for Latent Dirichlet Allocation

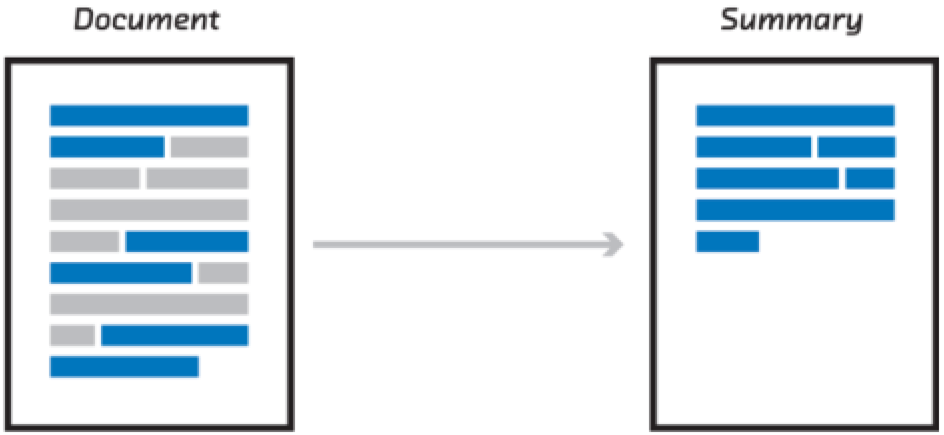
- Data cleaning is the main step for obtaining good results in the form of useful topic model for the latent dirichlet allocation model.
  - Tokenizing:
    - Since words are the lowest level elements in LDA, we segment each document into all the words, which is called tokenizing to words
  - Stop word removal:
    - Some unimportant words like if, for, etc. which are ununcessary / meaningless are removed from the above token list.
  - Stemming:
    - This is to reduce topically similar words to their root. Helps in viewing similar terms as equivalent and improves their importance in the model.
- Vectorization: This is done with bag of words implementation into a document term matrix which is the feature input into LDA.

Advanced Topics

Text summarization (source)

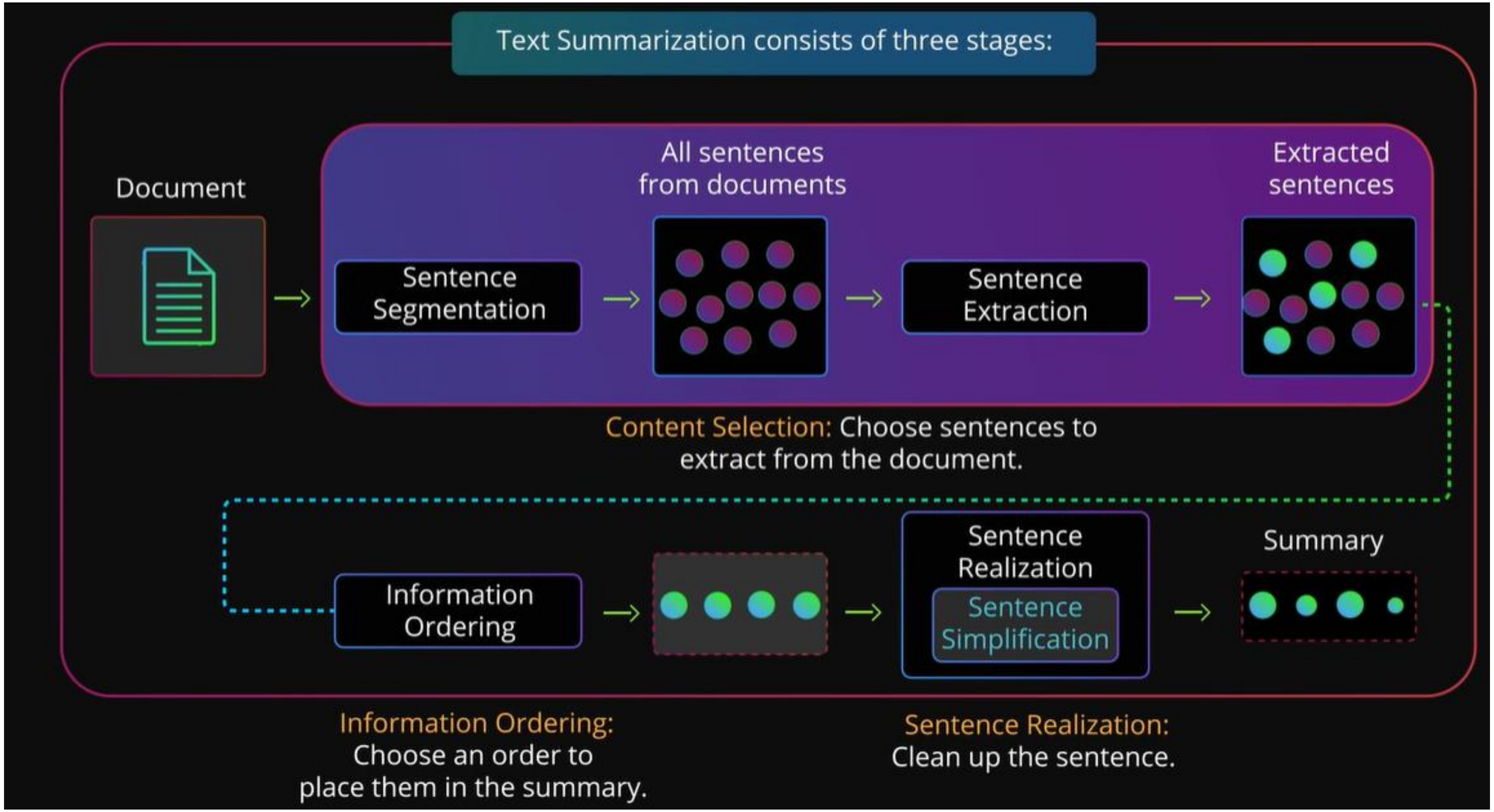
Why Text Summarization ?

- Text summarization is a very useful and important part NLP.
  - many lines of text data in any form
  - smaller semantic text form.
- Text summarization is the process of creating shorter text without removing the semantic structure of text.



Source:

Stages



Content Selection: Choose sentences to extract from the document.

Information Ordering: Choose an order to place them in the summary.

Sentence Realization: Clean up the sentence.

Approaches

- There are two approaches to text summarization.
  - Extractive approaches
  - Abstractive approaches

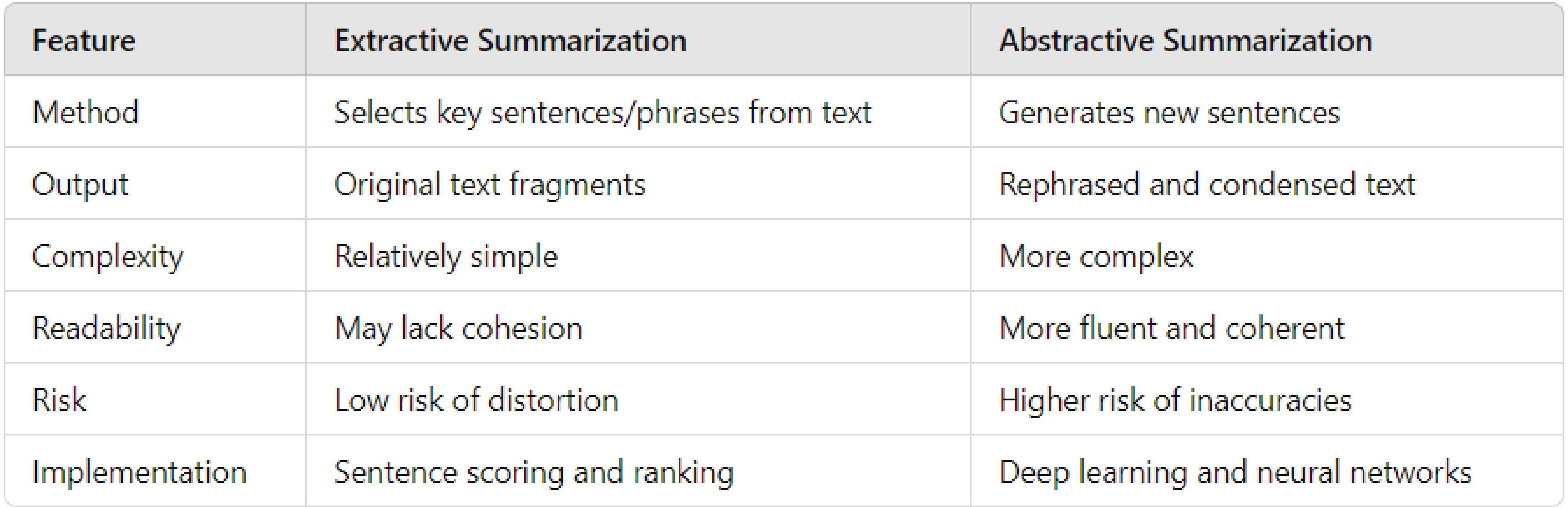
Extractive Summarization

- Definition: Extractive summarization involves selecting key sentences, phrases, or segments directly from the source text and combining them to create a summary. It does not generate new sentences but rather extracts and collates existing ones.
- Characteristics:
  - Selection-Based: The summary is created by identifying and selecting the most important parts of the original text.
  - Preservation of Original Phrasing: The extracted text maintains the original phrasing, vocabulary, and structure.
  - Relatively Simple: Typically easier to implement than abstractive summarization as it relies on techniques such as sentence scoring, ranking, and heuristic rules.

Abstractive Summarization

- Definition: Abstractive summarization involves generating new sentences that convey the most critical information from the original text. It attempts to paraphrase and shorten the source material, producing a more coherent and readable summary.
- Characteristics:
    - Generation-Based: The summary is created by understanding the content and generating new sentences that convey the same meaning.
    - Rephrasing: Uses natural language generation techniques to rewrite and condense the text.
    - Complex Algorithms: Often employs advanced techniques like neural networks, deep learning, and sequence-to-sequence models.

Comparisons



Feature	Extractive Summarization	Abstractive Summarization
Method	Selects key sentences/phrases from text	Generates new sentences
Output	Original text fragments	Rephrased and condensed text
Complexity	Relatively simple	More complex
Readability	May lack cohesion	More fluent and coherent
Risk	Low risk of distortion	Higher risk of inaccuracies
Implementation	Sentence scoring and ranking	Deep learning and neural networks

Thank you !