CA6001 Chapter 4
Natural Language Processing

Dr Zhang Jiehuang

College of Computing and Data Science

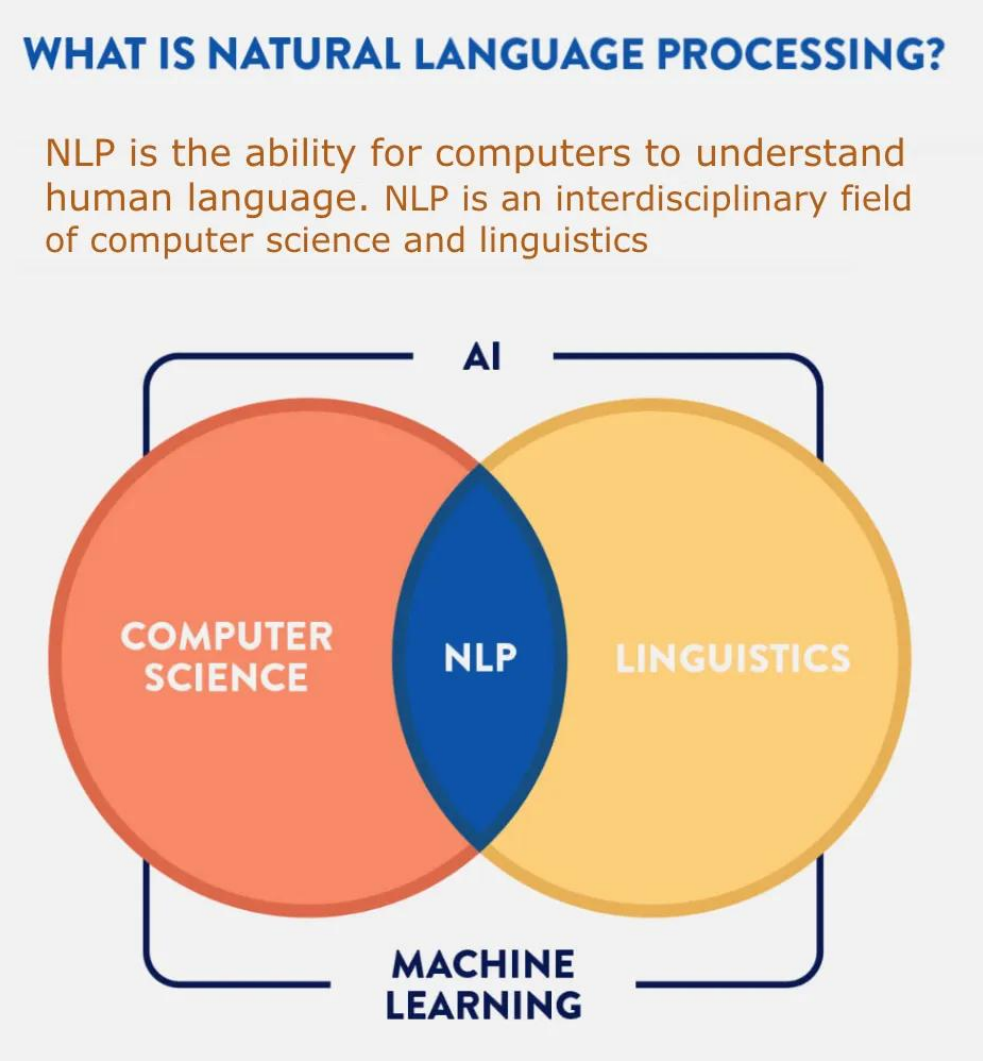
Nanyang Technological University

email: *jiehuang.zhang@ntu.edu.sg*

Chapter 10 – Natural Language Processing

1. What Is NLP – Key Concepts
2. Applications of NLP
3. Key Components and Core Tasks of NLP
4. Word Embeddings
5. Word2Vec and Skipgram
6. Sequence to Sequence
7. Recurrent Neural Networks
8. Transformers and Attention
9. Challenges and Future Directions

What is Natural Language Processing?



NLP studies formalisms, models and algorithms to allow computers to perform useful tasks involving knowledge about human languages

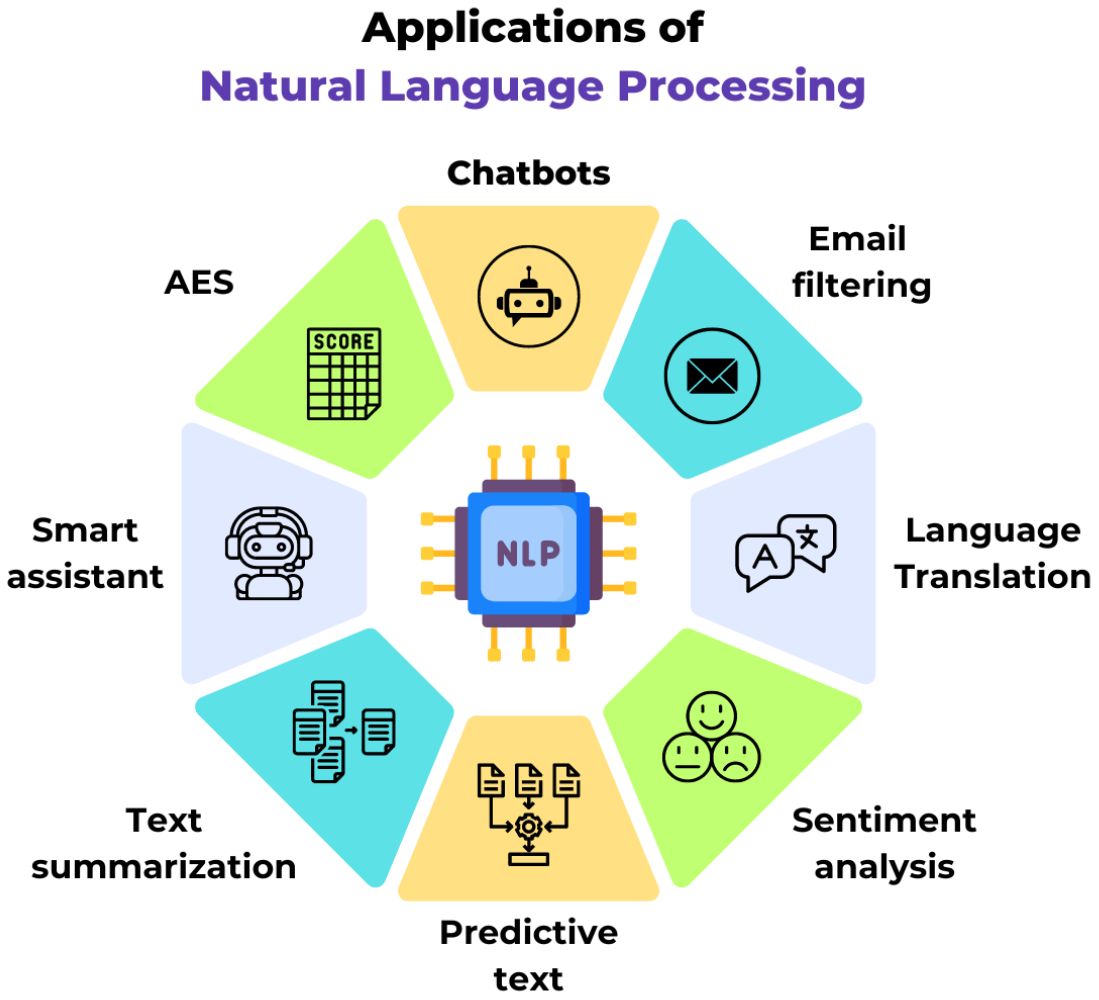
Goal of NLP is to generate human language (text or speech) in a way that is meaningful and useful

NLP emphasizes collective contextual understanding rather than a standalone comprehension of individual words

Transformers and LLMs are the state of the art in the field of NLP, enabling large scale training and deployment

https://aiml.com/what-is-natural-language-processing-nlp-list-some-of-the-most-common-nlp-tasks/

Applications of NLP



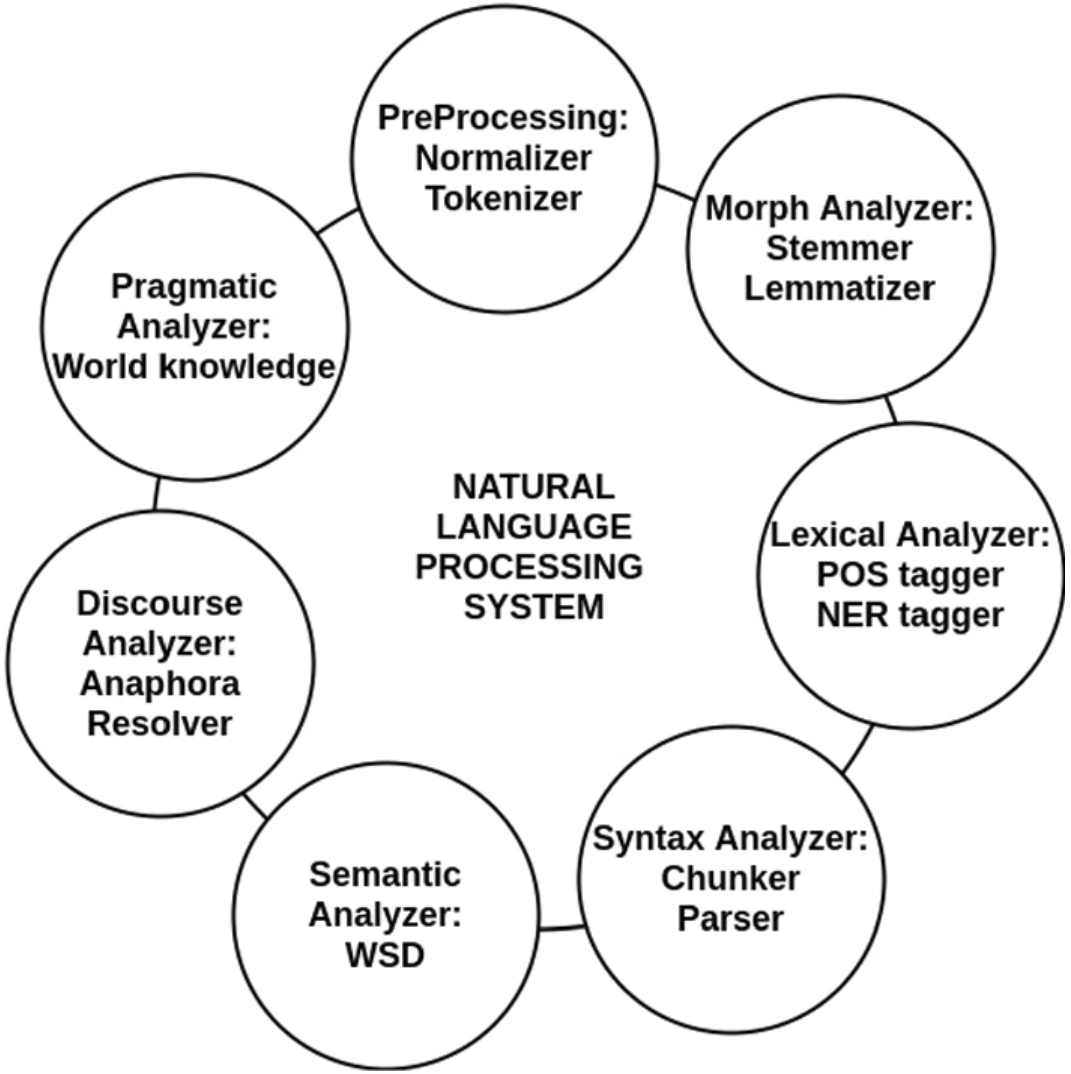
NLP has revolutionized how we interact with our devices, since the launch of ChatGPT in 2022

Large Language Models has the ability to understand and predict human language that borders on passing the Turing test

Most of these applications are enhanced by the power of LLMs and we are discovering new use cases on a regular basis

https://eastgate-software.com/top-8-applications-of-natural-language-processing-nlp/

Key Components of NLP



1. **Text Preprocessing**: Tokenization, stopword removal, lemmatization/stemming

2. **Syntactic Analysis**: Parsing, part-of-speech tagging

3. **Semantic Analysis**: Named Entity Recognition (NER), word sense disambiguation

4. **Pragmatics**: Contextual understanding, intent recognition

https://www.researchgate.net/figure/Phases-wise-Components-in-NLP-systems_fig1_357875502

Core NLP Preprocessing Tasks

Tokenization: Splitting text into smaller units (words, subwords, or sentences)

- Input: "Natural Language Processing is amazing!“
- Word tokenization: ["Natural", "Language", "Processing", "is", "amazing", "!"]
- Sentence tokenization: ["Natural Language Processing is amazing!"]

Stopword Removal: Eliminating common words that do not add much meaning (e.g., "the", "and", "is") to reduce dimensionality.

- Example: "The cat is on the mat" → "cat mat“

Stemming and Lemmatization: Reducing words to their root forms:

- Stemming: Cuts off prefixes/suffixes (e.g., "running" → "run")
- Lemmatization: Uses linguistic rules to find the base form (e.g., "better" → "good")

Syntax and Semantics

NLP models analyze sentence structure and meaning to improve understanding.

Part-of-Speech (POS) Tagging: Assigning grammatical categories to words (e.g., noun, verb, adjective) Example: "The quick brown fox jumps over the lazy dog.“

The” is tagged as determiner (DT)

“quick” is tagged as adjective (JJ)

“brown” is tagged as adjective (JJ)

“fox” is tagged as noun (NN)

“jumps” is tagged as verb (VBZ)

“over” is tagged as preposition (IN)

“the” is tagged as determiner (DT)

“lazy” is tagged as adjective (JJ)

“dog” is tagged as noun (NN)

Syntax and Semantics

Named Entity Recognition (NER): Identifying proper names in text (e.g., people, locations, organizations).

Example: "*Apple Inc. was founded by Steve Jobs in California*."

- Apple Inc. → Organization
- Steve Jobs → Person
- California → Location

Dependency Parsing: Identifying relationships between words in a sentence.

- Example: *"She enjoys playing soccer."*
- "enjoys" (verb) → subject: "She", object: "playing soccer"

Sentiment Analysis

Once text is structured, models attempt to extract meaning and relationships.

Sentiment Analysis: Detecting the sentiment of a text (positive, neutral, or negative).

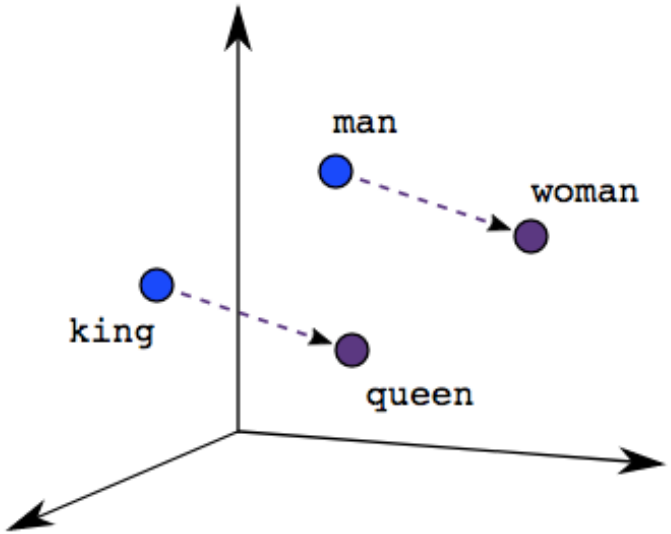
- Example: *"The movie was fantastic!"* → **Positive**
- Use case: Social media monitoring, customer feedback analysis

Word Embeddings: Representing words as numerical vectors to capture relationships.

- **Word2Vec**: Words with similar meanings have similar vectors (e.g., *king - man + woman ≈ queen*)

- **BERT and GPT**: Uses transformers to understand word context dynamically.

Word Embeddings



Male-Female

1.Word embeddings are a way of representing words as vectors in a multi-dimensional space

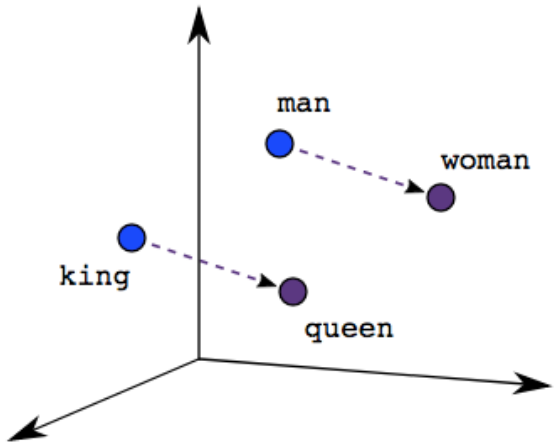
2.Word embeddings are integral to tasks such as text classification, sentiment analysis, machine translation and more

3. The distance and direction between vectors reflect the similarity and relationships among the corresponding words

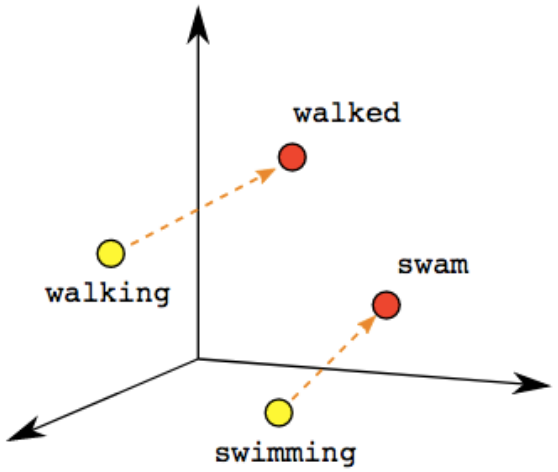
4. For example: man, king will have the same distance and direction as woman, queen in the multi-dimensional space,

https://www.ibm.com/think/topics/word-embeddings#:~:text=Word%20embeddings%20capture%20semantic%20relationships,more%20nuanced%20understanding%20of%20language.

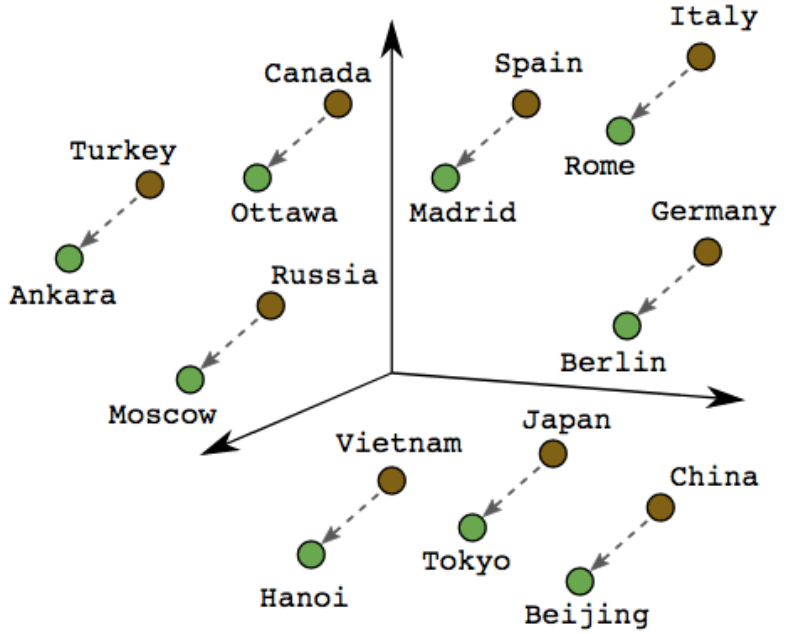
Word Embeddings



Male-Female



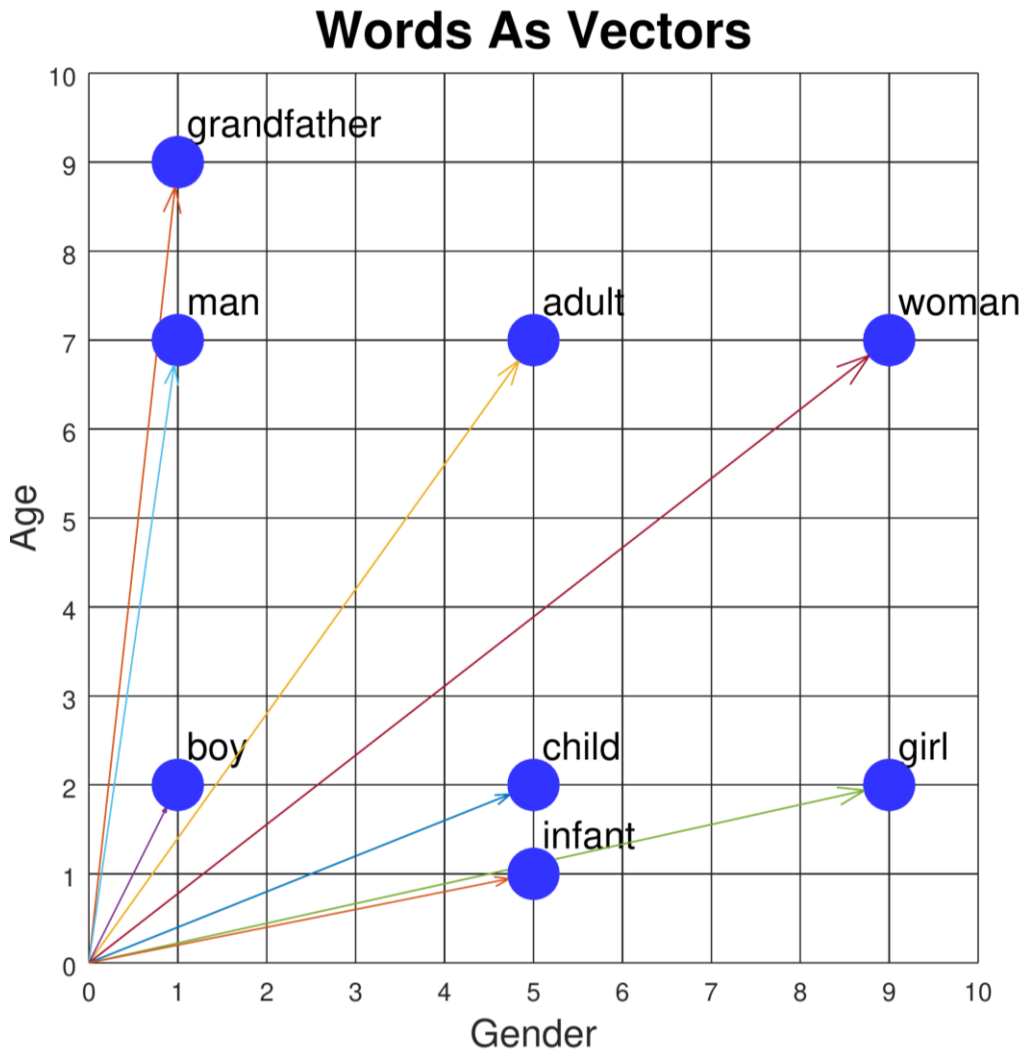
Verb Tense



Country-Capital

https://towardsdatascience.com/a-guide-to-word-embeddings-8a23817ab60f

Word2Vec and Skipgram



Word2Vec (Word to Vector) is a foundational technique for learning word embeddings in NLP

Word2Vec consists of 2 main models:

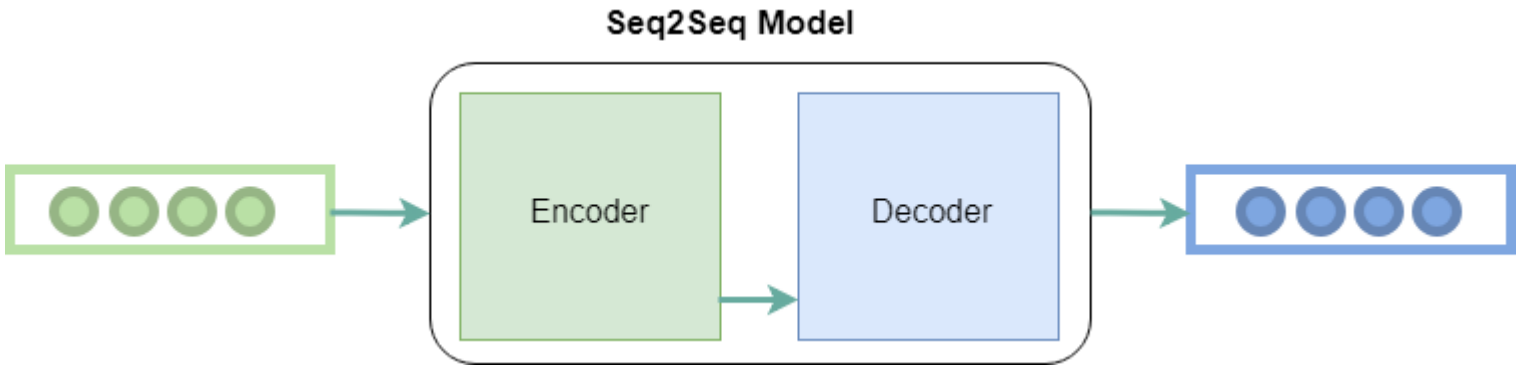
- Continuous Bag of Words (CBOW)
- Continuous Skip-gram

CBOW model takes a fixed number of context words (words surrounding the target word) as input. Each context word is represented as an embedding (vector) through a shared embedding layer. These embeddings are learned during the training process.

The Continuous Skip-gram model uses training data to predict the context words based on the target word's embedding. Specifically, it outputs a probability distribution over the vocabulary, indicating the likelihood of each word being in the context given the target word.

https://www.cs.cmu.edu/~dst/WordEmbeddingDemo/tutorial.html

Sequence to Sequence (Seq2Seq)

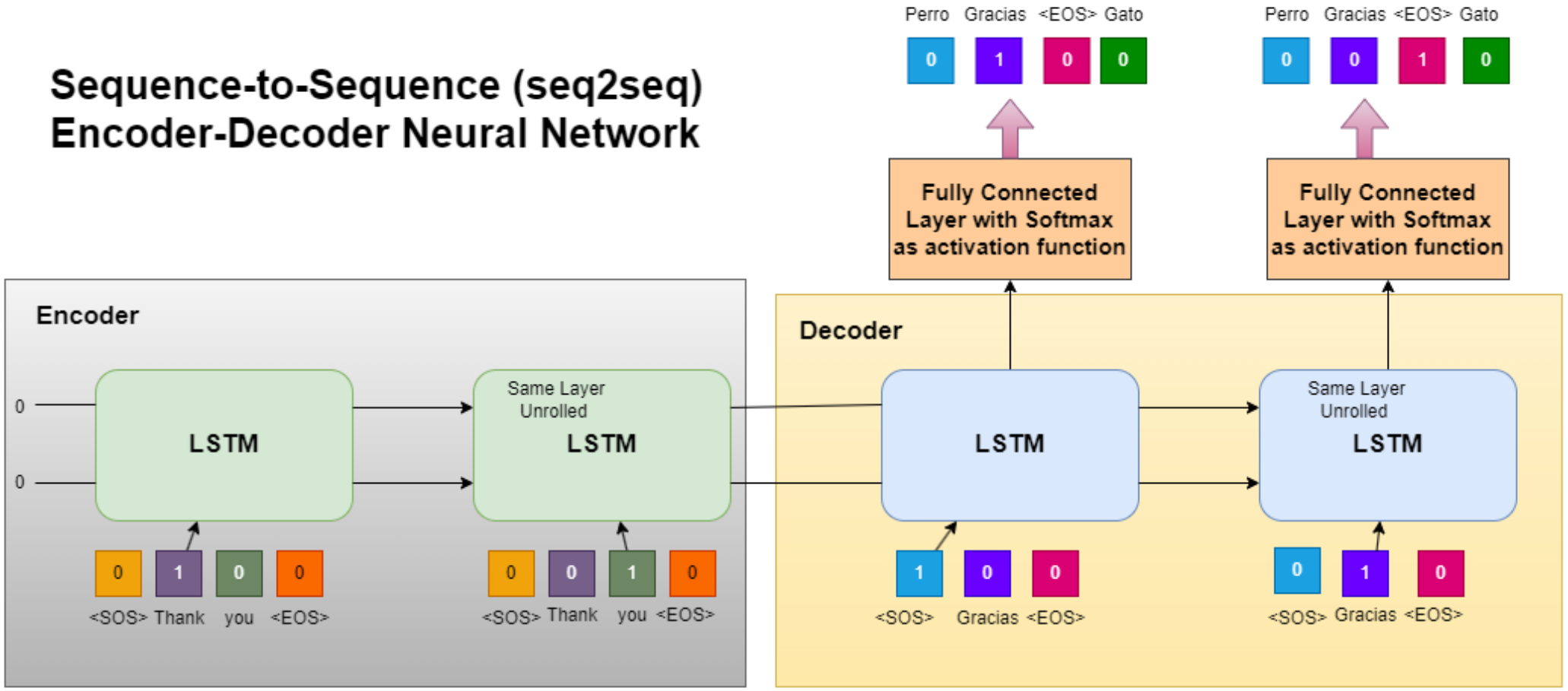


Seq2seq is a machine learning model designed for tasks involving sequential data, consisting of an encoder decoder both fundamental components. The neural network takes a input sequence, processes it, and generates a output sequence.

The encoder processes the input sequence and transforms it into a fixed-size hidden representation. The decoder uses the hidden representation to generate output sequence. The encoder-decoder structure allows them to handle input and output sequences of different lengths, making them capable to handle sequential data.

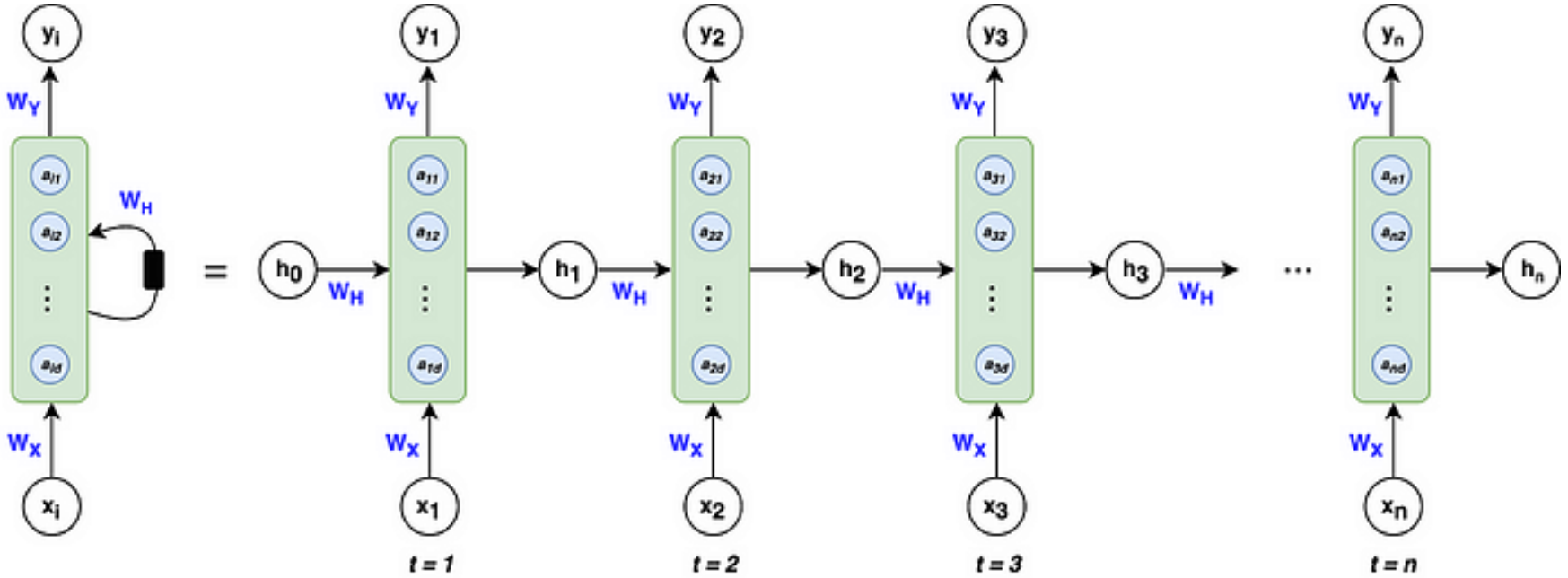
https://www.ibm.com/think/topics/word-embeddings#:~:text=Word%20embeddings%20capture%20semantic%20relationships,more%20nuanced%20understanding%20of%20language.

Sequence-to-Sequence (seq2seq) Encoder-Decoder Neural Network



Seq2Seq models are trained using a dataset of input-output pairs, where the input is a sequence of tokens, and the output is also a sequence of tokens. The model is trained to maximize the likelihood of the correct output sequence given the input sequence.

Recurrent Neural Networks (RNN)



In NLP, the context and order is extremely important, how do we address this?
RNNs introduce a mechanism where the output from one step is fed back as input to the next, as context from previous steps is essential
The hidden state (memory state) preserves essential information from previous inputs in the sequence
By using the same parameters across all steps, RNNs perform consistently across inputs

https://www.ibm.com/think/topics/word-embeddings#:~:text=Word%20embeddings%20capture%20semantic%20relationships,more%20nuanced%20understanding%20of%20language.

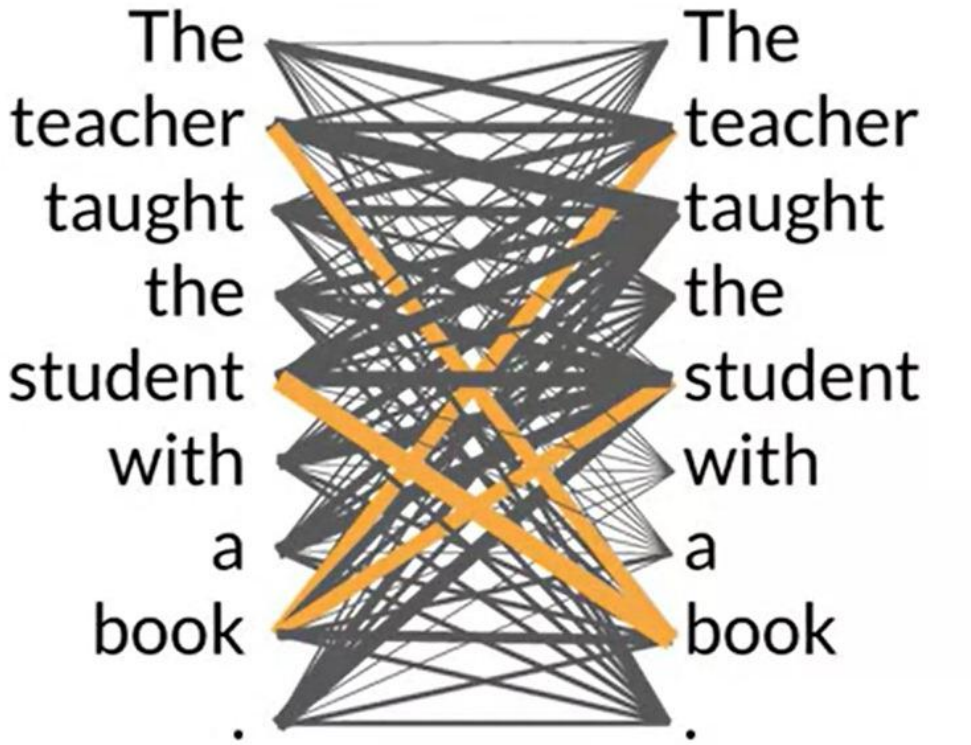
Transformer > RNN

While recurrent neural networks (RNNs) were once dominant for sequential data, the transformer architecture has largely replaced them as the go-to architecture, particularly in natural language processing (NLP) and other fields, due to its superior performance and scalability.

- **Advantages of Transformers:**
- **Parallel Processing:** Self-attention enables parallel processing, leading to faster training and inference times.
- **Long-Range Dependencies:** Transformers can effectively capture long-range dependencies in sequences, something RNNs struggle with.
- **Scalability:** Transformers are more scalable than RNNs, making them suitable for large datasets and complex tasks.

Attention Map

The teacher taught the student with a book



How does the model know the relevance of each word with respect to other words?

Self Attention allows the model to learn attention weights which shows how important each word is with respect to other words, not just the word next to it.

In the attention map on left, we see that “teacher” is strongly connected with “book”

Attention allows LLMs to encode human language

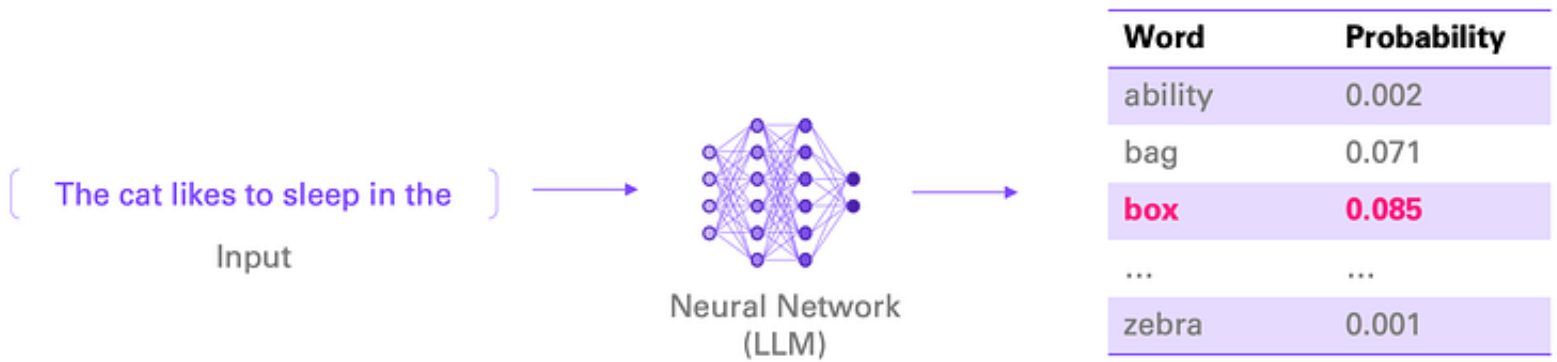
Language modeling

Imagine the following task: Predict the next word in a sequence

The cat likes to sleep in the ____ → What word comes next?

Now we have (say) ~50,000 classes (i.e. words)

Can we frame this as a ML problem? Yes, it's a classification task.



Word	Probability
ability	0.002
bag	0.071
box	**0.085**
...	...
zebra	0.001

Output

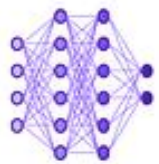
https://medium.com/data-science-at-microsoft/how-large-language-models-work-91c362f5b78f

Natural language generation

After training: We can generate text by predicting one word at a time

A trained language model can

Input



LLM

Word	Probability
speak	0.065
generate	**0.072**
politics	0.001
...	...
walk	0.003

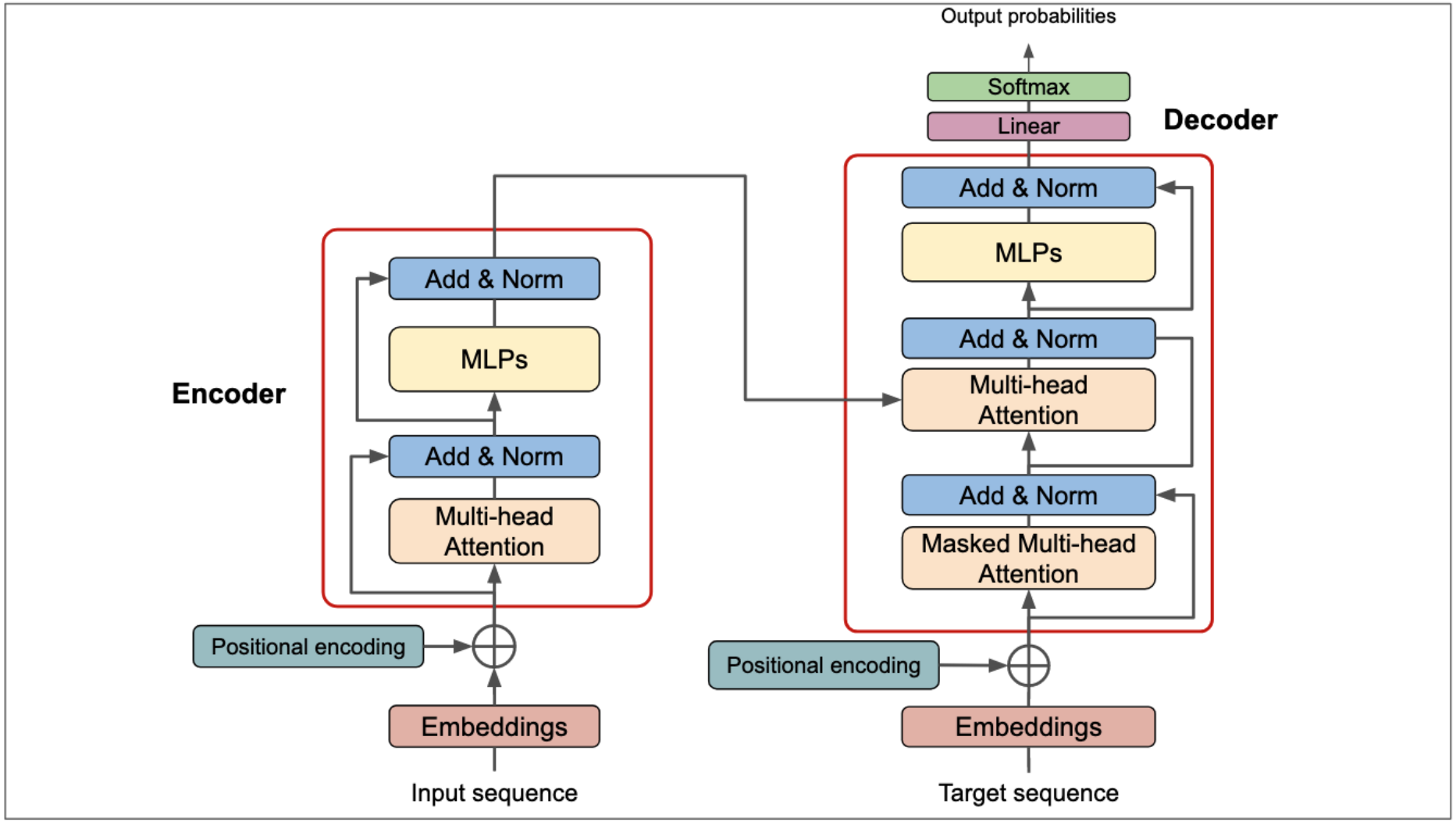
Output at step 1

Word	Probability
ability	0.002
text	0.084
coherent	**0.085**
...	...
ideas	0.041

Output at step 2

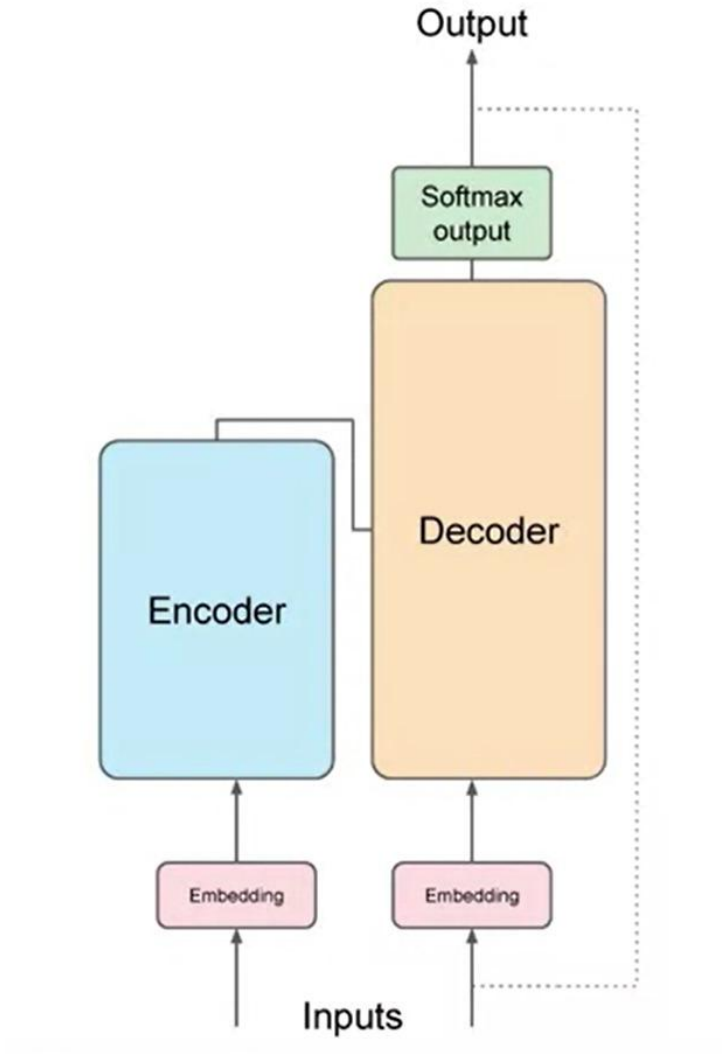
LLMs are an example of what's called "Generative AI"

Popular Architectures: Transformer



https://deeprevision.github.io/posts/001-transformer/

Popular Architectures: Transformer

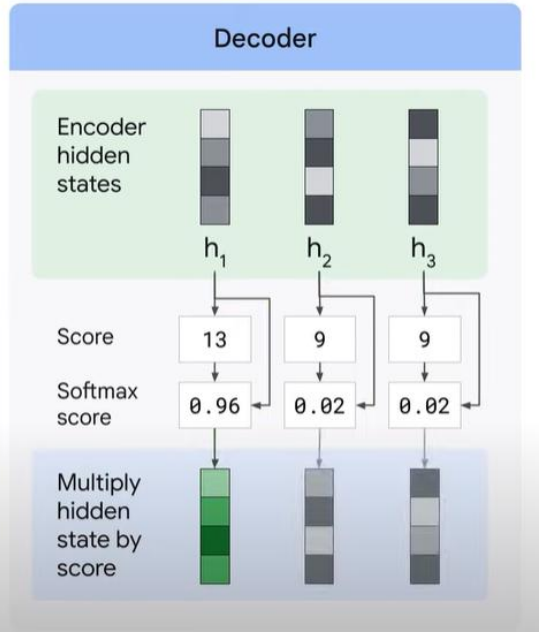


https://deeprevision.github.io/posts/001-transformer/

Attention and Transformers

To focus on the most relevant parts of the input:

1. Look at the set of encoder hidden states that it received.
2. Give each hidden state a score.
3. Multiply each hidden state by its soft-maxed score.



Attention mechanisms and transformer models consider contextual information and bidirectional relationships between words, leading to more advanced language representations.

Attention mechanisms were introduced to improve the ability of neural networks to focus on specific parts of the input sequence when making predictions. Instead of treating all parts of the input equally, attention mechanisms allow the model to selectively attend to relevant portions of the input.

Transformers use a self-attention mechanism to capture relationships between different words in a sequence. This mechanism allows each word to attend to all other words in the sequence, capturing long-range dependencies.

https://www.ibm.com/think/topics/word-embeddings#:~:text=Word%20embeddings%20capture%20semantic%20relationships,more%20nuanced%20understanding%20of%20language.

Self-Attention Mechanism

The quick brown fox jumped over the wall…. It was so agile

1. Input Representation:
Each word in the sequence is converted into a numerical vector (embedding), which contains its meaning in a high-dimensional space.

2. Three Key Matrices: Query, Key, and Value:
•**Query (Q):** Represents the word we're focusing on.
•**Key (K):** Represents all the other words in the sequence.
•**Value (V):** Contains the actual information of the words.
These matrices are learned during training and help the model calculate relationships between words.

3. Calculating Attention Scores:
The attention score measures how relevant each word is to the word being focused on. This is done by:
•Taking the **dot product** of the Query vector with each Key vector (indicates similarity).
•Scaling the score by dividing by the square root of the Key's dimension (helps with stable gradients).
•Applying a **softmax function** to convert the scores into probabilities.
For example, if "it" is the Query, the scores determine how much attention should be given to each word like "cat," "mat," etc.

4. Weighted Summation:
Each word's Value vector is multiplied by its attention score (from Step 3), and the results are summed up. This gives a new representation of the word "it," now enriched with context from other relevant words.

Social Impact: Chatbots for Mental Health

To address escalating stress anxiety and depression, many organisations are exploring the use of AI-driven wellness chatbots for employee support

These chatbots imitate human therapist interactions and offer tailer mental well being guidance and reduce the waiting time for patients to engage with support services

For example: Amazon adopted the Twill app, a self guided mental health program for their employees. Twill provides mood tracking, science backed games and activities designed to help employees navigate stress and depressive thoughts

https://www.wellable.co/blog/ai-chatbots-bridge-mental-healthcare-accessibility-gap/

Challenges

- **Ambiguity**: Polysemy, homonyms, sarcasm
- **Context Understanding**: Pragmatics, cultural nuances
- **Data Scarcity**: For low-resource languages or domains
- **Ethical Concerns**: Bias in data and models
- **Scalability**: Processing massive volumes of text efficiently

Future Directions

- **Multimodal NLP**: Integrating text with images and videos
- **Zero-Shot and Few-Shot Learning**: Handling unseen tasks with minimal data
- **Responsible AI**: Bias mitigation, explainability
- **Domain-Specific NLP**: Tailored models for specialized fields like medicine and law

References for Chapter 10 – NLP

1. https://www.datacamp.com/blog/top-machine-learning-use-cases-and-algorithms
2. https://www.databricks.com/resources/ebook/big-book-of-machine-learning-use-cases/thank-you?scid=7018Y000001Fi19QAC&utm_source=google&utm_adgroup=141597893652&utm_offer=big-book-of-machine-learning-use-cases&utm_term=machine+learning+use+cases&gad_source=1&gclid=CjwKCAiAxqC6BhBcEiwAlXp45zG
3. https://www.researchgate.net/publication/351021675_Artificial_intelligence_in_cancer_diagnostics_and_therapy_Current_perspectives-G9y0tvxwNF2eskPqGlVAsxxtPXDibjGQBobW-_5A4ZhFFsDKTRoCWT8QAvD_BwE
4. https://www.heavy.ai/technical-glossary/fraud-detection-and-prevention
5. Andrew Ng's Machine Learning course https://www.coursera.org/learn/machine-learning/lecture/Q8Vvp/supervised-learning-part-2
6. https://cloud.google.com/discover/what-is-unsupervised-learning
7. https://blog.aspiresys.com/data-and-analytics/customer-segmentation-empowered-by-machine-learning-reap-the-benefits-of-ai-to-serve-your-customers-better/
8. https://www.v7labs.com/blog/supervised-vs-unsupervised-learning

Note: All online articles were accessed between Oct to Nov 2024

The End
Questions?