



# **ICTSS00120 - Artificial Intelligence Skill Set**

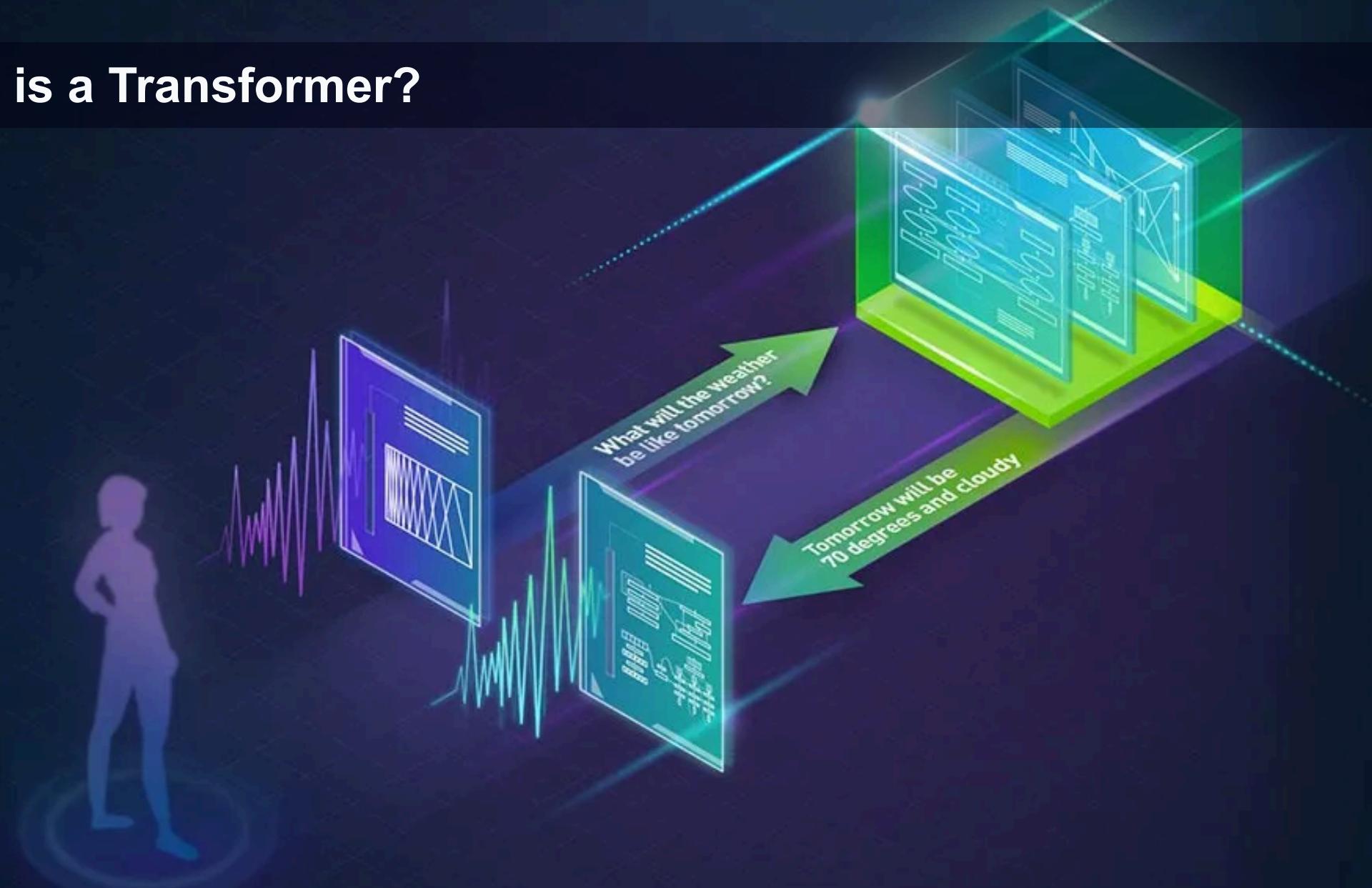
## **Session 12: Introduction to Transformers and Hugging Face**

Lecturer: Jordan Hill

## Learning Objectives

- Understand the basic concepts of Transformer models.
- Explore the architecture and applications of Transformers.
- Introduction to Hugging Face and its tools.
- Learn how to get started with Transformers using Hugging Face.

# What is a Transformer?



# What is a Transformer?

## Overview

- **Transformer**: A deep learning model designed for handling sequential data, introduced in the paper *"Attention is All You Need"* by Vaswani et al.
- **Core Component**: Attention mechanism which allows the model to focus on different parts of the input sequence.

That is an encoder-decoder transformer

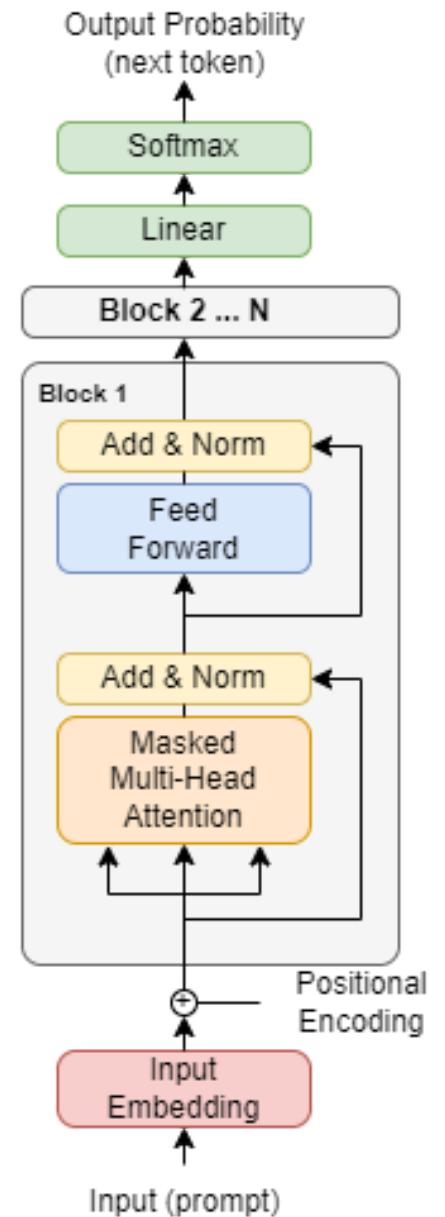
Most LLMs use self-attention

This is to say Decoder-only

What??

Reference: this [stackexchange response](#) is actually really good summary!

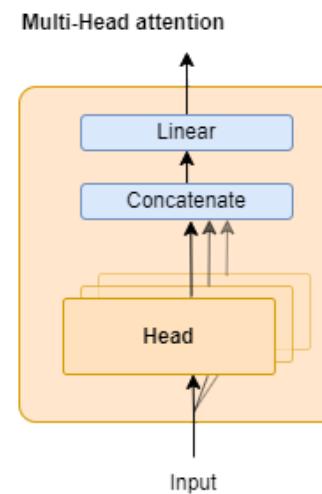
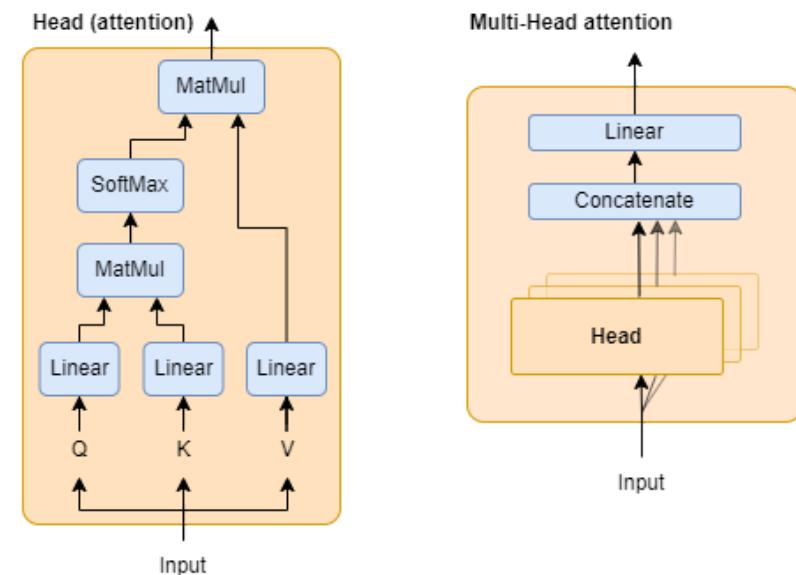
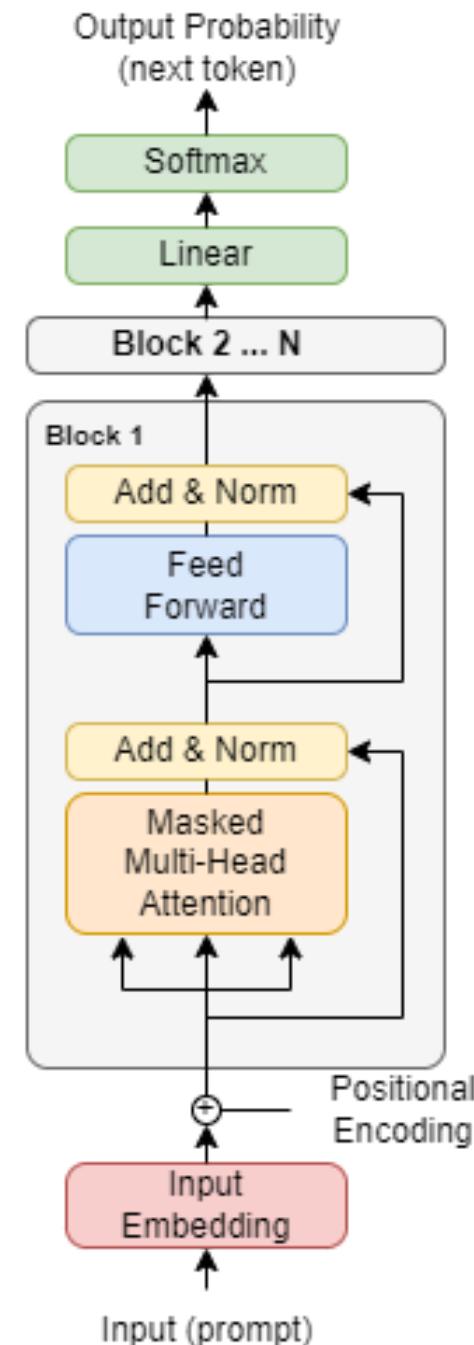
We'll talk more about this attention stuff next week...



This is a **Decoder-Only** model.

These simply pass all the information one way through multiple blocks (Attention & FNN)

You just start with the input sequence (tokenized/embedded) and get the next token out the other end.



Back to the encoder-decoder...

These 'pay attention' to the input and another sequence and uses the other sequence (Output embedding) to influence **the logits** / Next token.

```
<iframe width="560" height="315" src="https://www.youtube.com/embed/0_4KEb08xrE?si=Bz7nyD53aY4D46ed" title="YouTube video player" frameborder="0" allow="accelerometer;autoplay; clipboard-write; encrypted-media; gyroscope; picture-in-picture; web-share" referrerpolicy="strict-origin-when-cross-origin" allowfullscreen></iframe>
```

# Transformer Architecture

## Key Components

- **Encoder-Decoder Structure:**
  - **Encoder:** Processes the input data and generates a representation.
  - **Decoder:** Uses the representation from the encoder to generate the output.
- **Attention Mechanisms:**
  - **Self-Attention:** Allows the model to weigh the importance of different words in a sentence relative to each other.
  - **Multi-Head Attention:** Multiple attention mechanisms run in parallel to capture different aspects of the data.

```
<iframe width="100%" height="580" src="https://www.youtube.com/embed/wjZofJX0v4M"  
title="But what is a GPT? Visual intro to transformers | Chapter 5, Deep Learning"  
frameborder="0" allow="accelerometer; autoplay; clipboard-write; encrypted-media;  
gyroscope; picture-in-picture; web-share" referrerpolicy="strict-origin-when-cross-origin"  
allowfullscreen></iframe>
```

# Applications of Transformers

## Real-World Use Cases

Domain	Applications
Natural Language Processing (NLP)	<ul style="list-style-type: none"><li>• Machine Translation</li><li>• Text Summarization</li><li>• Question Answering</li><li>• Sentiment Analysis</li></ul>
New Domains	<ul style="list-style-type: none"><li>• Protein Folding (AlphaFold) — This is a really big deal!</li><li>• Image Recognition (Vision Transformers - ViTs)</li><li>• DNA Sequence Analysis</li></ul>

# Hugging Face!

## Overview

- **Hugging Face**: An open-source AI research organization known for its NLP library called Transformers.
- **Transformers Library**: Provides pre-trained models and tools to easily integrate transformers into various applications.



# Introduction to Hugging Face

## Key Features

- **Model Hub:** Repository of pre-trained models.
- **Datasets:** Access to a wide range of datasets for different tasks.
- **Inference API:** Easily deploy models for prediction.



# Exploring Hugging Face

## First Stop The Documentation

A huge ecosystem of opensource client libraries!  
(Hint: the source code is a massive resource for learning more)

## Next Huggingface Learn

## Next Stop MODELS!

## Finally, Datasets

...also Datasets (the client library)



## More than utility

Huggingface has more than directly useful resources—it has a community!

[Daily Papers](#)

[Blog Articles & News](#)

A new feature is also [Spaces](#)

[Check out this example app](#)



## Getting Started with Hugging Face

**The Transformers library gives us access to a huge library of open source model architectures**

**Install the Transformers Library (pre-installed on Kaggle):**

```
pip install transformers
```

**It makes it really easy to work with models locally**

## Let's Start with a simple one

```
from transformers import pipeline  
  
# Load a pre-trained model for sentiment analysis  
classifier = pipeline("sentiment-analysis")
```

### Make Predictions :

```
result = classifier("I love using the Hugging Face library!")  
print(result)
```

## A bit more fun

```
import torch
from transformers import pipeline

model_id = "meta-llama/Llama-3.2-1B"

pipe = pipeline(
    "text-generation",
    model=model_id,
    torch_dtype=torch.bfloat16,
    device_map="auto"
)

pipe("What is the meaning of life?")
```

**What does your model say is the meaning of life?**

# Break

Let's take a break before we dive into  
Training using the transformers library

# Intro to Transformers

**Getting Started:**

## 1. Huggingface Transformers Quicktour

**There are only really two kinds of things we might do with a pre-trained model:**

**Either Inference or Training**

**The Trainer API makes it really easy to train LLMs**

*But it also hides a lot of complexity from you!*

## Summary and Q&A

### Summary:

- Introduction to Transformer models and their architecture.
- Overview of Hugging Face and its tools.
- Hands-on guide to getting started with Transformers using Hugging Face.
- Application examples in NLP and other domains like DNA analysis in marine biology.

### Q&A:

- Any questions about today's topics?

Contact: [jordan.hill@nmtafe.wa.edu.au](mailto:jordan.hill@nmtafe.wa.edu.au)