AI Text Classification

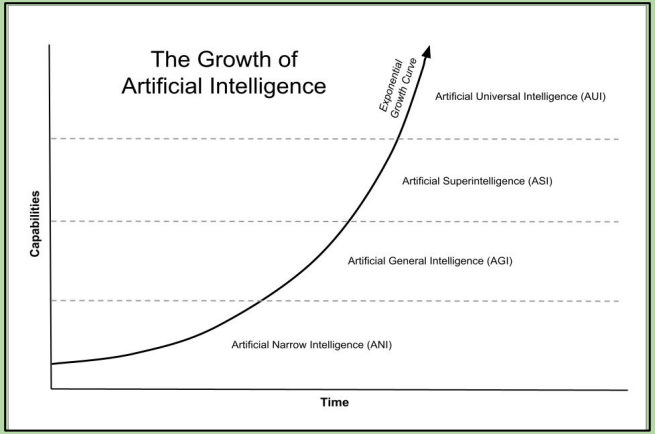
AI Group 1: Samyuktha Nair, Arnav Nair, Advay Dinesh, Advait Baijulal

How many of you follow news articles on Social Media?




Introduction and Research Question

- Indistinguishability between AI and Human written text leads to:
  - **Plagiarism**
  - **Deep fake** messages
  - Loss of **trust** in online platforms



(The Science of Machine Learning and AI)

? Who should be held **accountable** for false AI-generated content on social media or in research articles?

  ? Solely with **users** or with the **AI companies** that they use to produce the content?

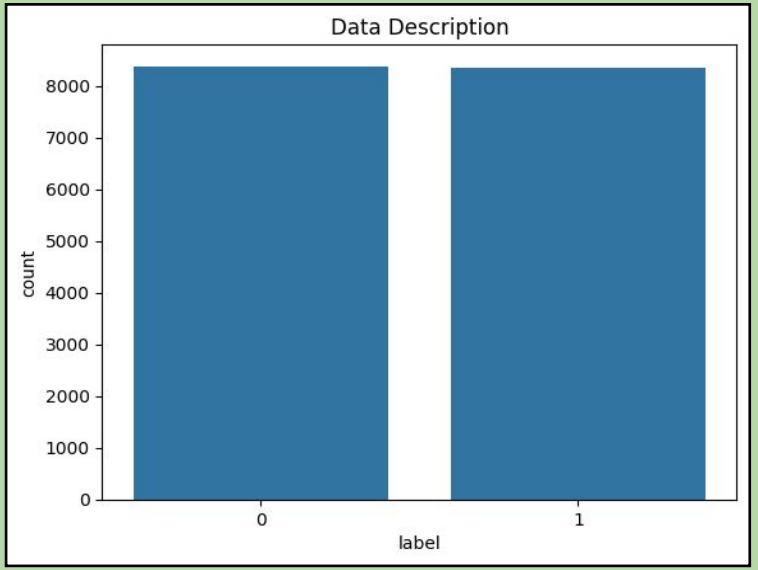
✔ Explore **methods to detect AI** generated content.

✔ Improve **transparency** and **accountability** in online creation.

Dataset

Data	Label
Human Text	0
AI Text	1

Total Number of Samples = 20,283



Dataset label distribution against total count*

Preprocessing and Filtering

What's processed and filtered?

- **Lowercase** Ex: "Filter" vs "filter"
- **Punctuation and non-alphabetic tokens** Ex: Dashes, colons, parenthesis, numbers, etc.
- **Application of stop words** Ex: "of", "the", "is", etc.
- **Natural Language Toolkit** (NLTK)

These steps **highlight** meaningful **linguistic features** while **simplifying** the input for the model.

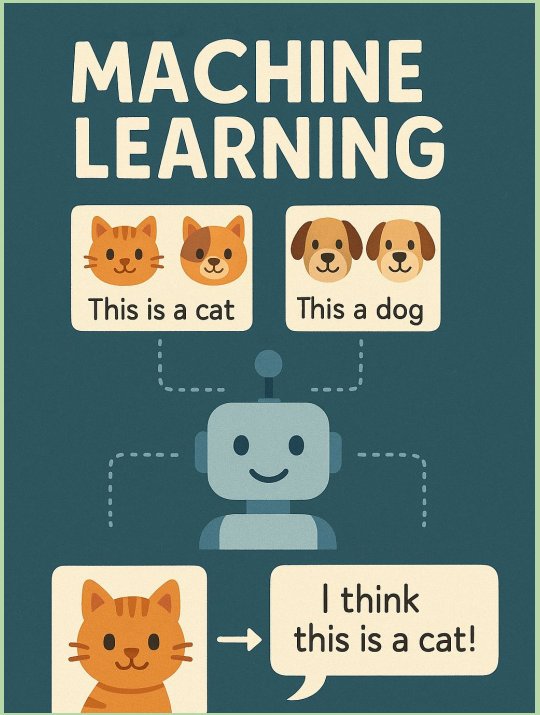
	abstract	label
0	OBJECTIVE: This retrospective chart review des...	0.0
1	Inflammatory diseases of the respiratory tract...	0.0
2	Surfactant protein-D (SP-D) participates in th...	0.0
3	Endothelin-1 (ET-1) is a 21 amino acid peptide...	0.0
4	Respiratory syncytial virus (RSV) and pneumoni...	0.0

Preprocessing + Filtering*

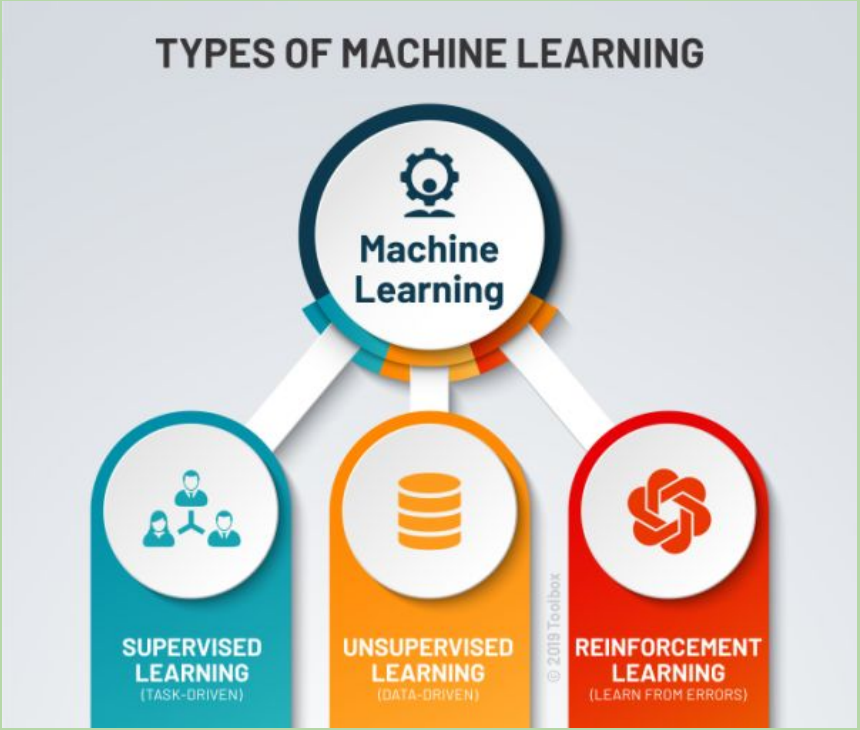
label	cleaned_text
0.0	objective retrospective chart review describes...
0.0	inflammatory diseases respiratory tract common...
0.0	surfactant proteind spd participates innate re...
0.0	amino acid peptide diverse biological activity...
0.0	respiratory syncytial virus rsv pneumonia viru...

Machine Learning

A branch of Artificial Intelligence focused on creating models and algorithms allowing computers to learn data patterns and make predictions without explicit coding.



Three Types of Learning



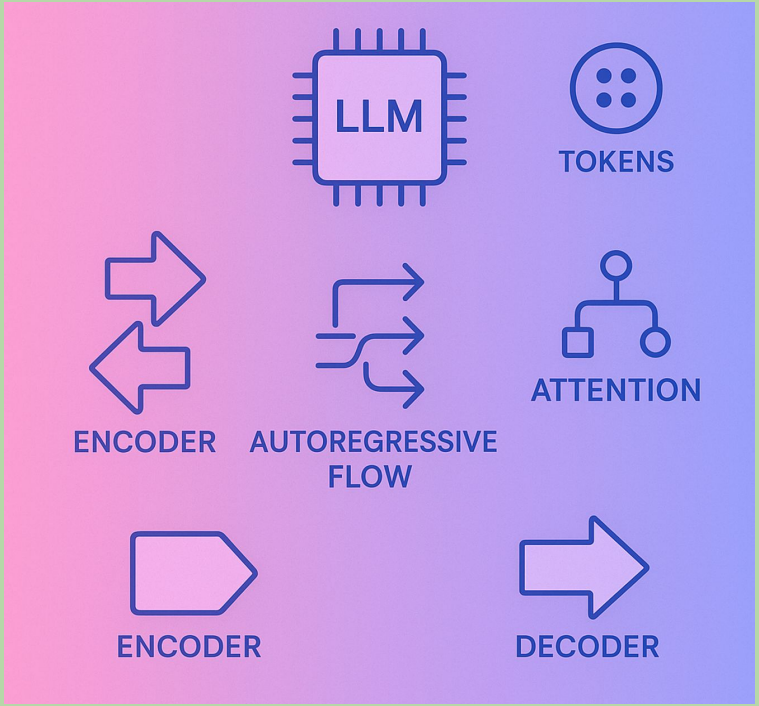
Supervised: Trains models using labeled data and comparisons between actual values and predictions.

Unsupervised: Trains model with unlabeled data and identifies hidden pattern to create groupings.

Reinforcement: Trains model through a decision making environment and a trial and error process.

Transformers

A deep learning neural network architecture that helps turn an input sequence into an output sequence.

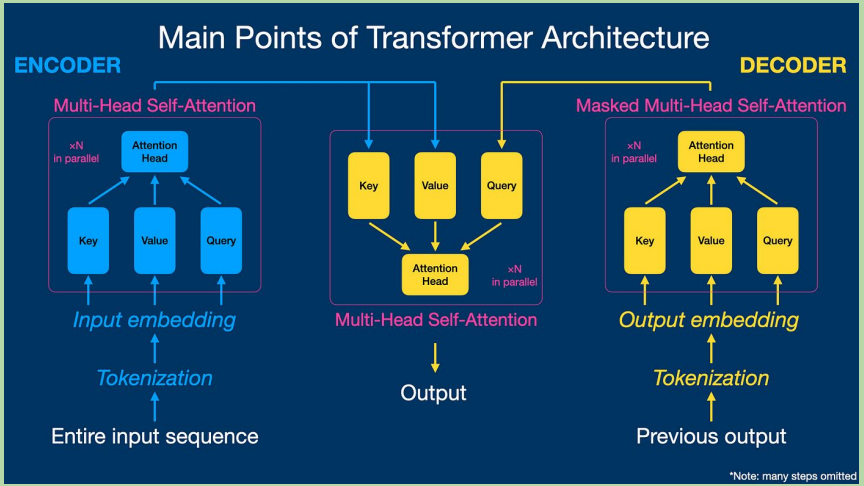


Transformers

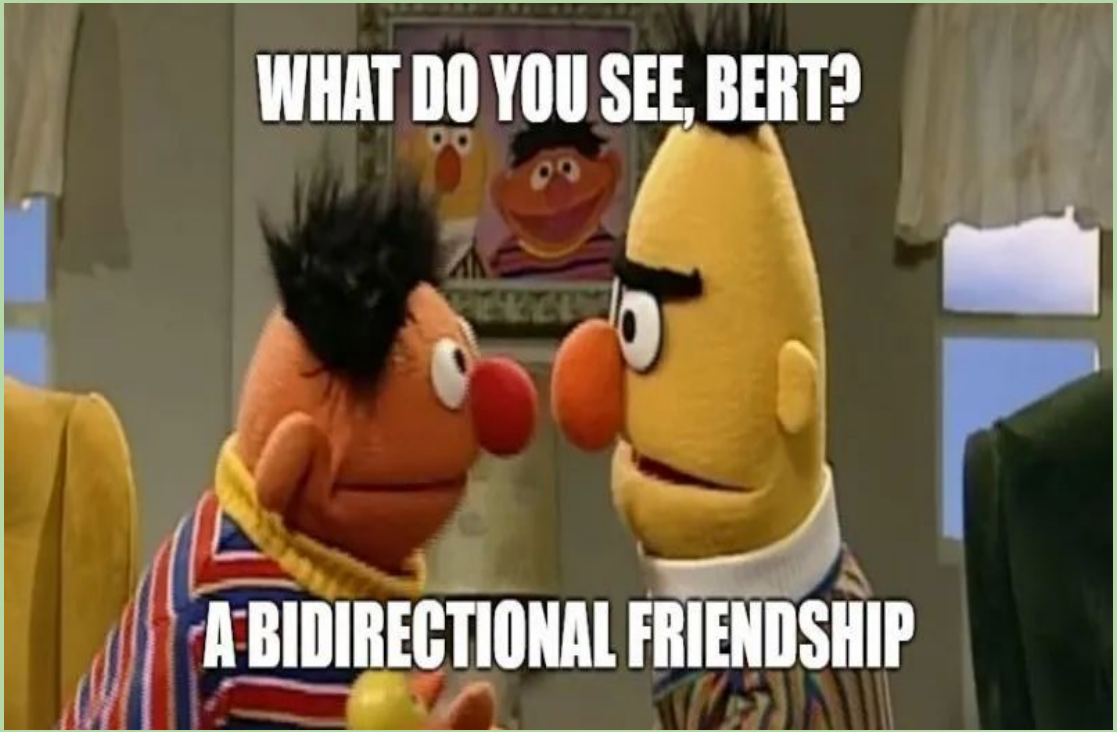
Encoder-only: Input sequence is read and contextualized / no new sequence is generated

Encoder-Decoder: Combines encoder and decoder to compress input into a vector and then create an output

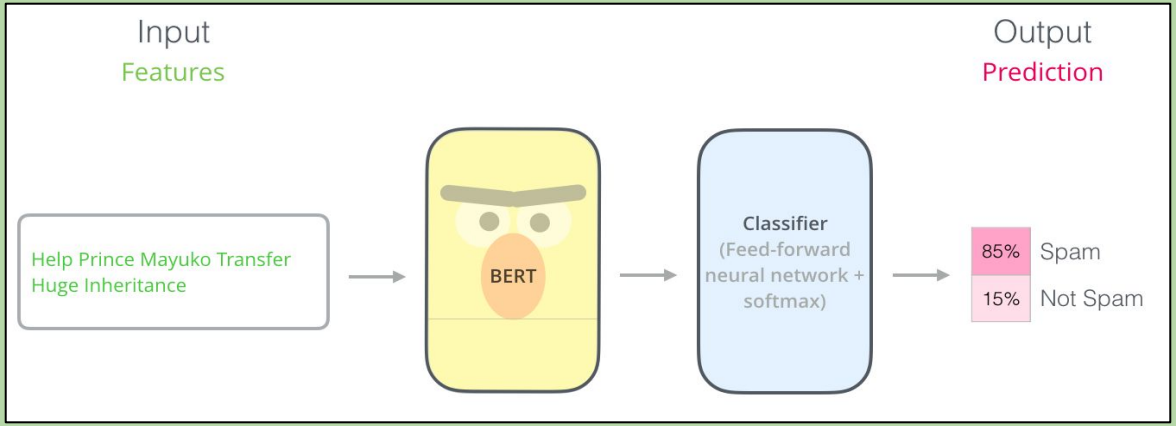
Decoder-only: Takes the sequence and generates a new token one at a time using self-attention



BERT



BERT



Model	No of layers
BERT-Base/RoBERTa-Base	12
BERT-Large/RoBERTa-Large	24
DistilBERT	6
TinyBERT	4

*(https://jalammar.github.io/a-visual-guide-to-using-bert-for-the-first-time/)

BERT Models

Model	Approx. Parameters	Relative Size	Relative Speed	Relative Performance
BERT-Base	**110M**	1x	1x	Baseline
RoBERTa-Base	125M	~1.1x	~1x	~3-5% > BERT
DistilBERT	66M	~0.6x	~1.6x Faster	~97% of BERT
TinyBERT (4L)	**14.5M**	~0.13x	~9.4x Faster	~96% of BERT

Model Evaluation

Model	Average Training Time for each epoch	Total number of epochs	Resources used	Accuracy
BERT	**01:22:17**	10	Tesla 4 GPU	0.9931
RoBERTa	00:39:04	10	Tesla 4 GPU	0.6516
DistilBERT	00:39:07	10	Tesla 4 GPU	0.9948
TinyBERT	**00:03:10**	5	Tesla 4 GPU	0.9876

Are You Smarter Than the Model?

Global warming is a serious threat today. We need to take serious action to promote sustainability and address climate change. In the long term, this could pose a serious threat to our environment and many species.

Are You Smarter Than the Model?

Global warming is a serious threat today. We need to take serious action to promote sustainability and address climate change. In the long term, this could pose a serious threat to our environment and many species.

Actual Classification	Model Prediction
Human	Human

Are You Smarter Than the Model?

Climate change has become an indigent topic in our day-to-day life. From bush fires in New South Wales and Victoria to the wildlife forest fires in California, it has become an inevitable subject that needs more attention. One of the vital causes of climate change is the disappearance or extinction of important species. The species could range from insects to wild mammals. Hence, it is crucial to trace the location species that play a vital role in the food chain and ecosystem to further study and conserve. We introduce a novel method to extract the geographical coordinates of localities and the distribution of species from georeferenced maps. This is done through a series of processing steps. We propose a software toolbox that extracts maps from the textbooks,occurrence points (distribution points of the species) from these maps and finally, georeferencing and postprocessing the maps to extract the geographical coordinates of the occurrence points. The data relating to the distribution of species, their habitats would pave the way for the study of functional traits or habitats of species with their abiotic properties of the environment (Zeuss, 2020).

Are You Smarter Than the Model?

Climate change has become an indigent topic in our day-to-day life. From bush fires in New South Wales and Victoria to the wildlife forest fires in California, it has become an inevitable subject that needs more attention. One of the vital causes of climate change is the disappearance or extinction of important species. The species could range from insects to wild mammals. Hence, it is crucial to trace the location species that play a vital role in the food chain and ecosystem to further study and conserve. We introduce a novel method to extract the geographical coordinates of localities and the distribution of species from georeferenced maps. This is done through a series of processing steps. We propose a software toolbox that extracts maps from the textbooks,occurrence points (distribution points of the species) from these maps and finally, georeferencing and postprocessing the maps to extract the geographical coordinates of the occurrence points. The data relating to the distribution of species, their habitats would pave the way for the study of functional traits or habitats of species with their abiotic properties of the environment (Zeuss, 2020).

Actual Classification	Model Prediction
Human	Human

Are You Smarter Than the Model?

Here in Offenbach, Germany, on a warm afternoon in early July, nature feels both immediate and serene. The sun is likely casting a warm, golden light over the city, coaxing the deepest greens from the leaves of the trees lining the Main River. This time of year, parks like the Büsing-Park or Dreieichpark are in their summer glory, with flowerbeds bursting in vibrant color and the air filled with the gentle hum of bees. The experience of nature here isn't one of vast, untamed wilderness, but of its intimate integration with city life. It's found in the shade of a mature linden tree, the gentle flow of the river, and the chorus of birdsong that persists over the urban rhythm—a welcome, living counterpoint to the stone and asphalt of the city

Are You Smarter Than the Model?

Here in Offenbach, Germany, on a warm afternoon in early July, nature feels both immediate and serene. The sun is likely casting a warm, golden light over the city, coaxing the deepest greens from the leaves of the trees lining the Main River. This time of year, parks like the Büsing-Park or Dreieichpark are in their summer glory, with flowerbeds bursting in vibrant color and the air filled with the gentle hum of bees. The experience of nature here isn't one of vast, untamed wilderness, but of its intimate integration with city life. It's found in the shade of a mature linden tree, the gentle flow of the river, and the chorus of birdsong that persists over the urban rhythm—a welcome, living counterpoint to the stone and asphalt of the city

Actual Classification	Model Prediction
AI	AI

Are You Smarter Than the Model?

The CPU (Central Processing Unit), often referred to as the "brain" of the computer, is a critical hardware component that performs most of the processing inside a computer. It executes instructions from programs by performing basic arithmetic, logic, control, and input/output (I/O) operations.

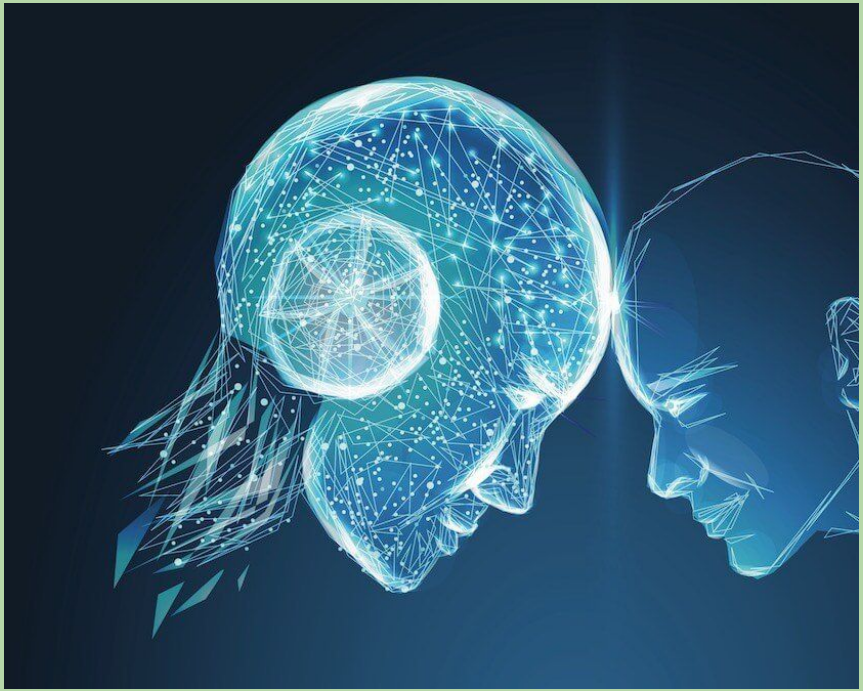
Are You Smarter Than the Model?

The CPU (Central Processing Unit), often referred to as the "brain" of the computer, is a critical hardware component that performs most of the processing inside a computer. It executes instructions from programs by performing basic arithmetic, logic, control, and input/output (I/O) operations.

Actual Classification	Model Prediction
AI	Human

Concluding Thoughts

- ✔ Longer text is associated with greater accuracy than shorter text
- ✔ Goal of making AI watermark detector given more time (Phase 1 Complete)
- ✔ AI generation’s future
- ✔ Our mission: Promote unique thought and original ideas



Acknowledgements

- Ms. Madhuvanthi Venkatesh
- Dr. Latha Nair
- NAMAM and MBN Foundation
- Data Nova Coordinators

THANK YOU

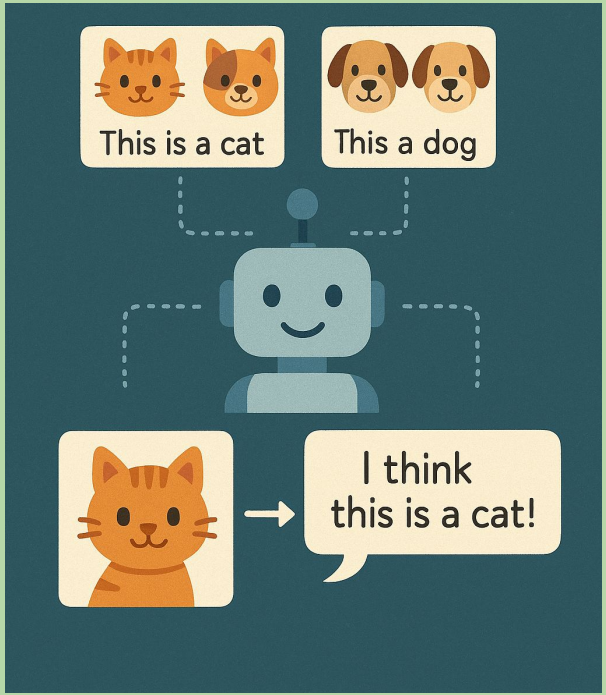
ANY QUESTIONS?

Lecture Slides

AI Group 1: Samyuktha Nair, Arnav Nair, Advay Dinesh, Advait Baijulal

Mentor: Madhuvanthi Venkatesh

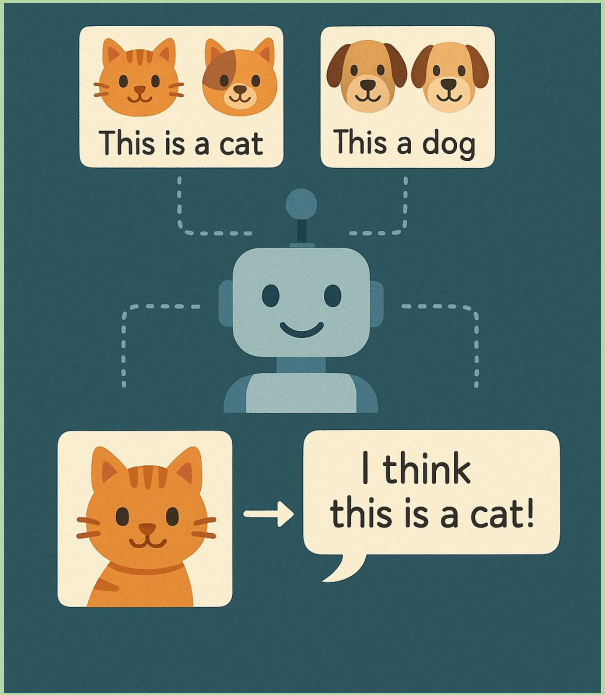
Machine Learning



Machine Learning

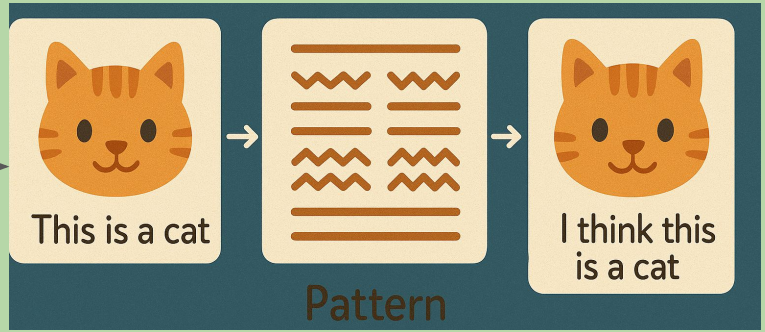
Machine Learning is when computers learn from examples instead of being told exactly what to do.

Data → **Training** → **Prediction**



Types of Data

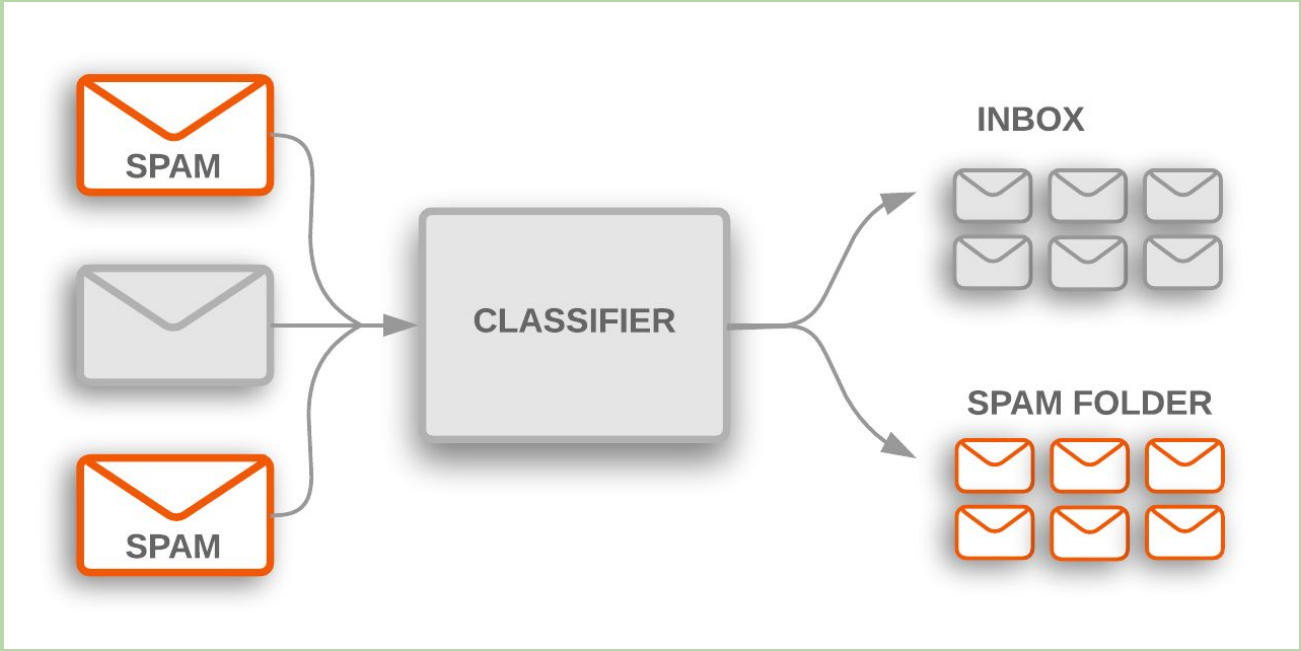
Image



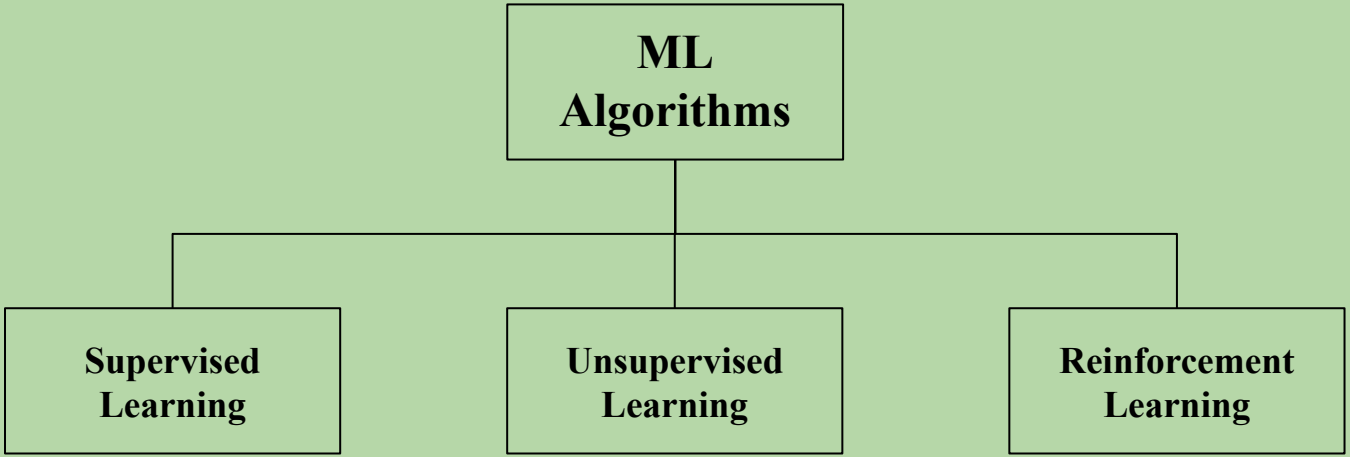
Text

"This game is awesome!" → Happy 😊
"I love ice cream!" → Happy 😊
"I hate doing homework." → Not Happy 😠
"This is the worst movie ever." → Not Happy 😠

Machine Learning



ML Basics


ML Algorithms
Supervised Learning
Unsupervised Learning
Reinforcement Learning

ML Basics

Sentence: "I love chocolate!" → Label: Happy 😊
Sentence: "This is boring." → Label: Not happy 😐

ML Basics

Supervised Learning

Sentence: "I love chocolate!" → Label: Happy 😊
Sentence: "This is boring." → Label: Not happy 😐

ML Basics

Supervised Learning

Sentence: "I love chocolate!" → Label: Happy 😊
Sentence: "This is boring." → Label: Not happy 😐

A computer controlling a game character:

- If it wins → +10 points
- If it falls → -5 points

ML Basics

Supervised Learning

Sentence: "I love chocolate!" → Label: Happy 😊
Sentence: "This is boring." → Label: Not happy 😐

Reinforcement Learning

A computer controlling a game character:

- If it wins → +10 points
- If it falls → -5 points

ML Basics

Supervised Learning

Sentence: "I love chocolate!" → Label: Happy 😊
Sentence: "This is boring." → Label: Not happy 😐

Reinforcement Learning

A computer controlling a game character:

- If it wins → +10 points
- If it falls → -5 points

"I love pizza."
"Pizza is the best!"
"I am so tired."
"I'm sleepy."

ML Basics

Supervised Learning

Sentence: "I love chocolate!" → Label: Happy 😊
Sentence: "This is boring." → Label: Not happy 😐

Reinforcement Learning

A computer controlling a game character:

- If it wins → +10 points
- If it falls → -5 points

Unsupervised Learning

"I love pizza."
"Pizza is the best!"
"I am so tired."
"I'm sleepy."

ML Basics

Text Classfication

Supervised Learning

Sentence: "I love chocolate!" → Label: Happy 😊
Sentence: "This is boring." → Label: Not happy 😐

Reinforcement Learning

A computer controlling a game character:

- If it wins → +10 points
- If it falls → -5 points

Unsupervised Learning

"I love pizza."
"Pizza is the best!"
"I am so tired."
"I'm sleepy."

Example 1

A robot is shown hundreds of fruits. Each one is labeled as either “apple,” “banana,” or “orange.” The robot learns to recognize fruits.

Example 1

A robot is shown hundreds of fruits. Each one is labeled as either “apple,” “banana,” or “orange.” The robot learns to recognize fruits.

Supervised Learning

Example 2

A computer program plays a maze game. If it reaches the goal, it gets points. If it hits a wall, it loses points. It learns the best path by trying over and over.

Example 2

A computer program plays a maze game. If it reaches the goal, it gets points. If it hits a wall, it loses points. It learns the best path by trying over and over.

Reinforcement Learning

Example 3

A robot is given a big box of LEGOs without any labels. It groups them based on size and color.

Example 3

A robot is given a big box of LEGOs without any labels. It groups them based on size and color.

Unsupervised Learning

Example 4

An AI sees many pictures of animals. Each picture says what the animal is. It learns to recognize cats, dogs, and elephants.

Example 4

An AI sees many pictures of animals. Each picture says what the animal is. It learns to recognize cats, dogs, and elephants.

Supervised Learning

Example 5

A music app groups songs that “sound similar,” without knowing anything about what the songs are called or what genre they are.

Example 5

A music app groups songs that “sound similar,” without knowing anything about what the songs are called or what genre they are.

Unsupervised Learning

Example 6

A robot watches people play chess, but it’s not told who wins. It tries to find different types of strategies people use.

Example 6

A robot watches people play chess, but it’s not told who wins. It tries to find different types of strategies people use.

Unsupervised Learning

Example 7

A robot dog is trained to sit, jump, and roll over. Every time it does the trick right, it gets a treat!

Example 7

A robot dog is trained to sit, jump, and roll over. Every time it does the trick right, it gets a treat!

Reinforcement Learning

Example 8

A spam filter is trained using emails marked “spam” or “not spam” to block junk messages.

Example 8

A spam filter is trained using emails marked “spam” or “not spam” to block junk messages.

Supervised Learning

Example 9

A self-driving car gets a reward every time it follows traffic rules, and a penalty when it crashes or makes a mistake.

Example 9

A self-driving car gets a reward every time it follows traffic rules, and a penalty when it crashes or makes a mistake.

Reinforcement Learning

Example 10

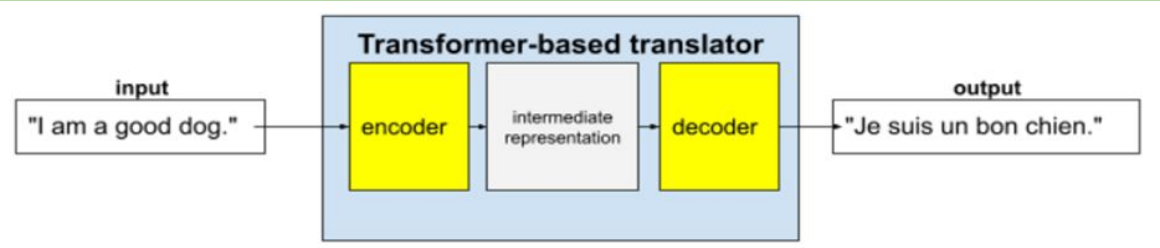
A system reads lots of product reviews labeled as “positive” or “negative” and learns to detect emotions in new reviews.

Example 10

A system reads lots of product reviews labeled as “positive” or “negative” and learns to detect emotions in new reviews.

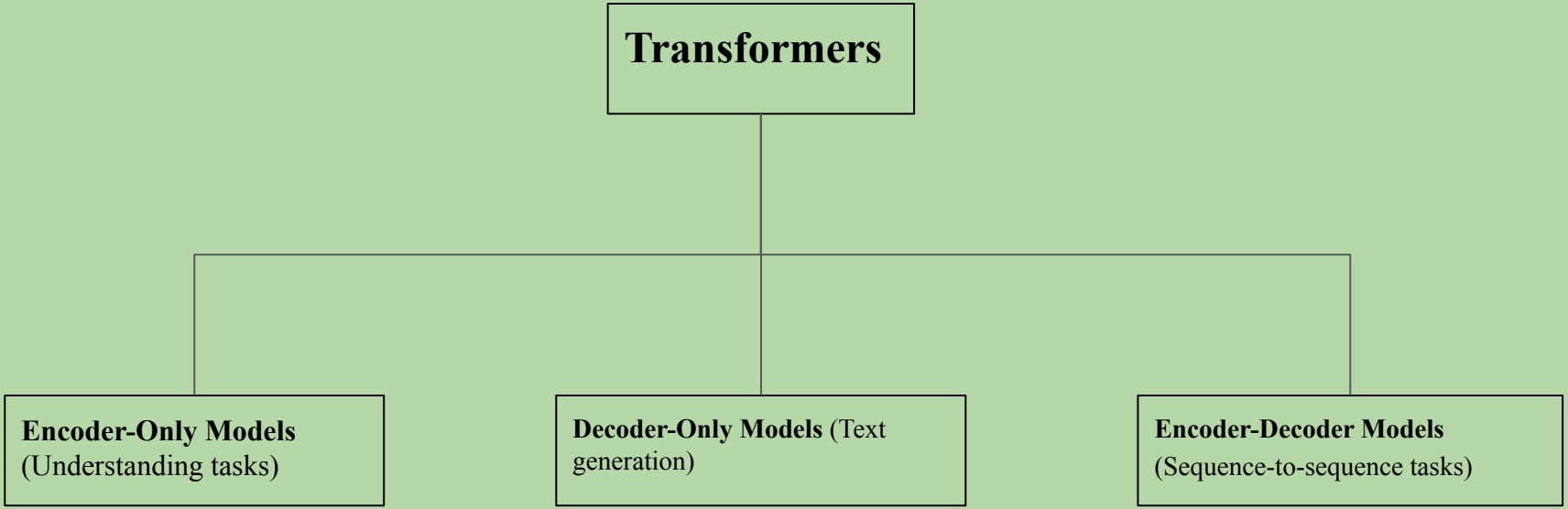
Supervised Learning

Transformers

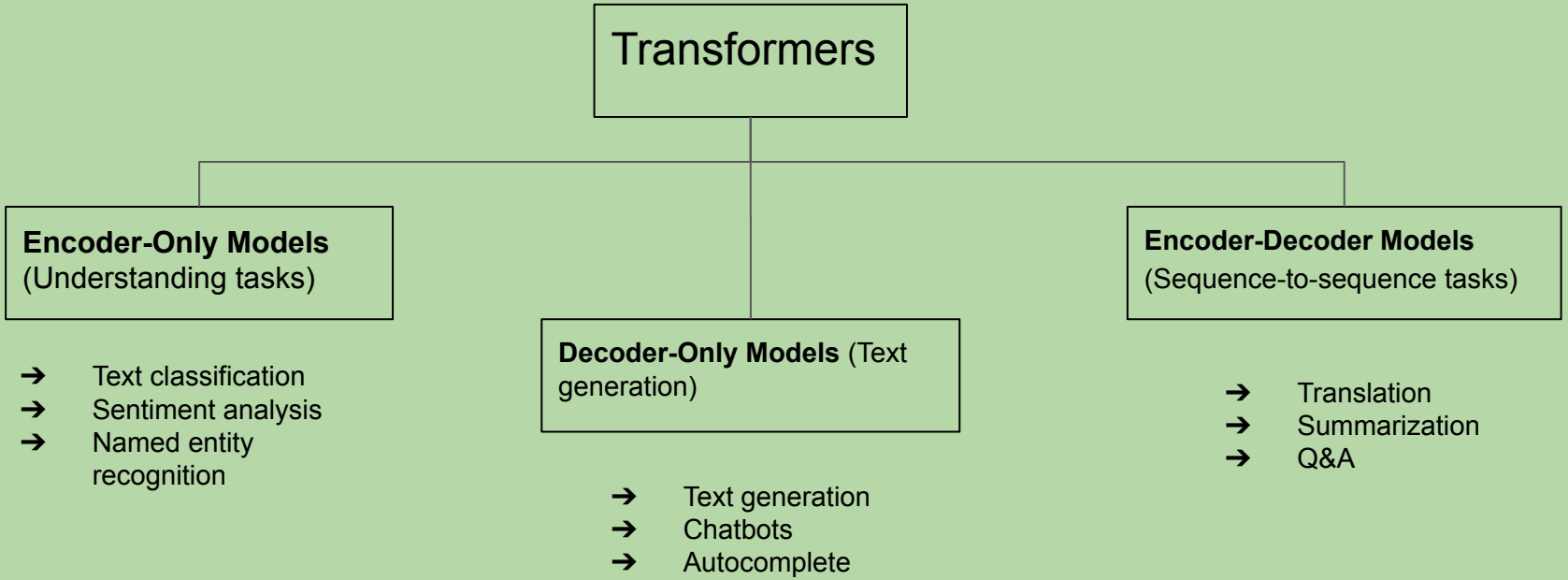


https://hanlab.mit.edu/projects/tinyml

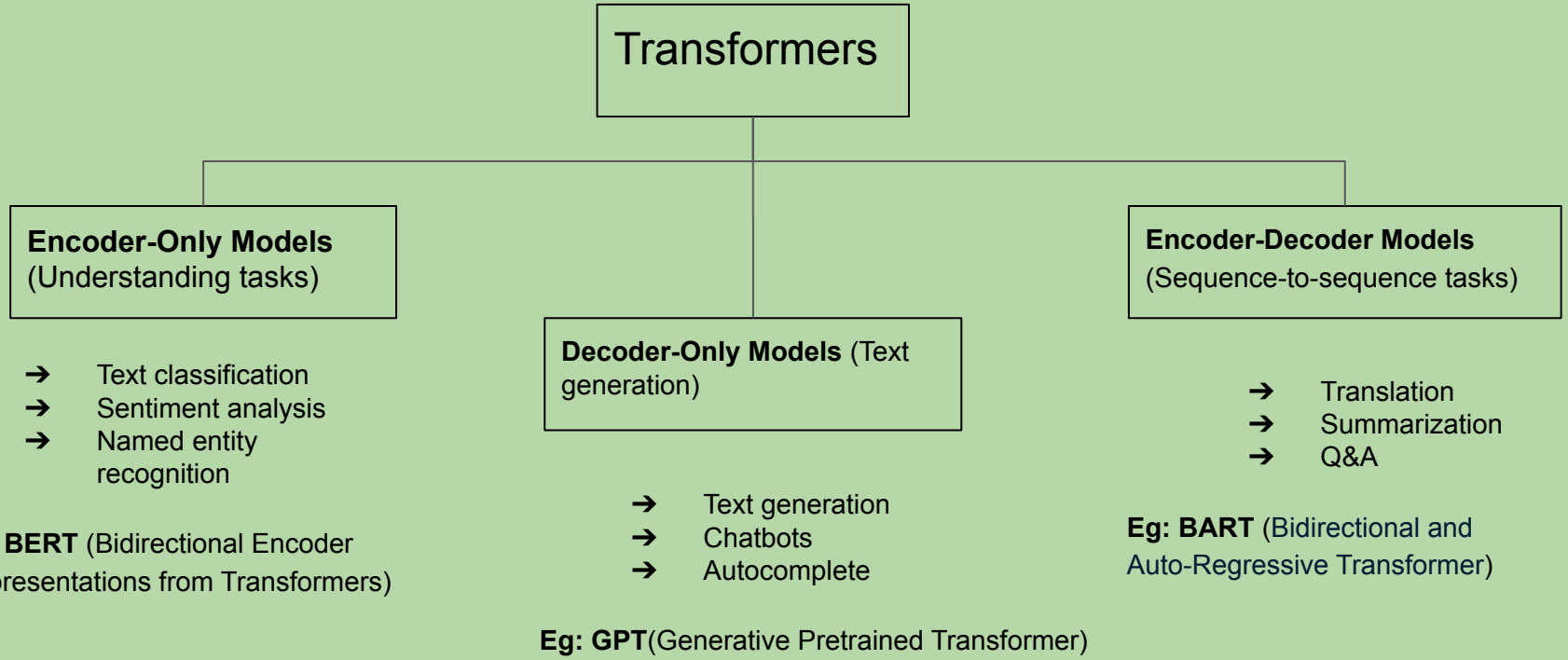
Transformers



Transformers



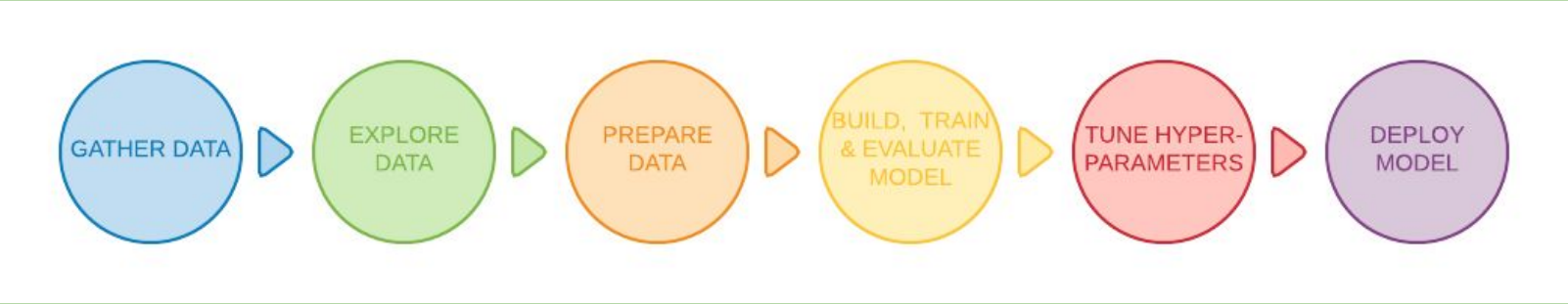
Transformers



Text Classification

Supervised Learning

Encoder- Based Transformer

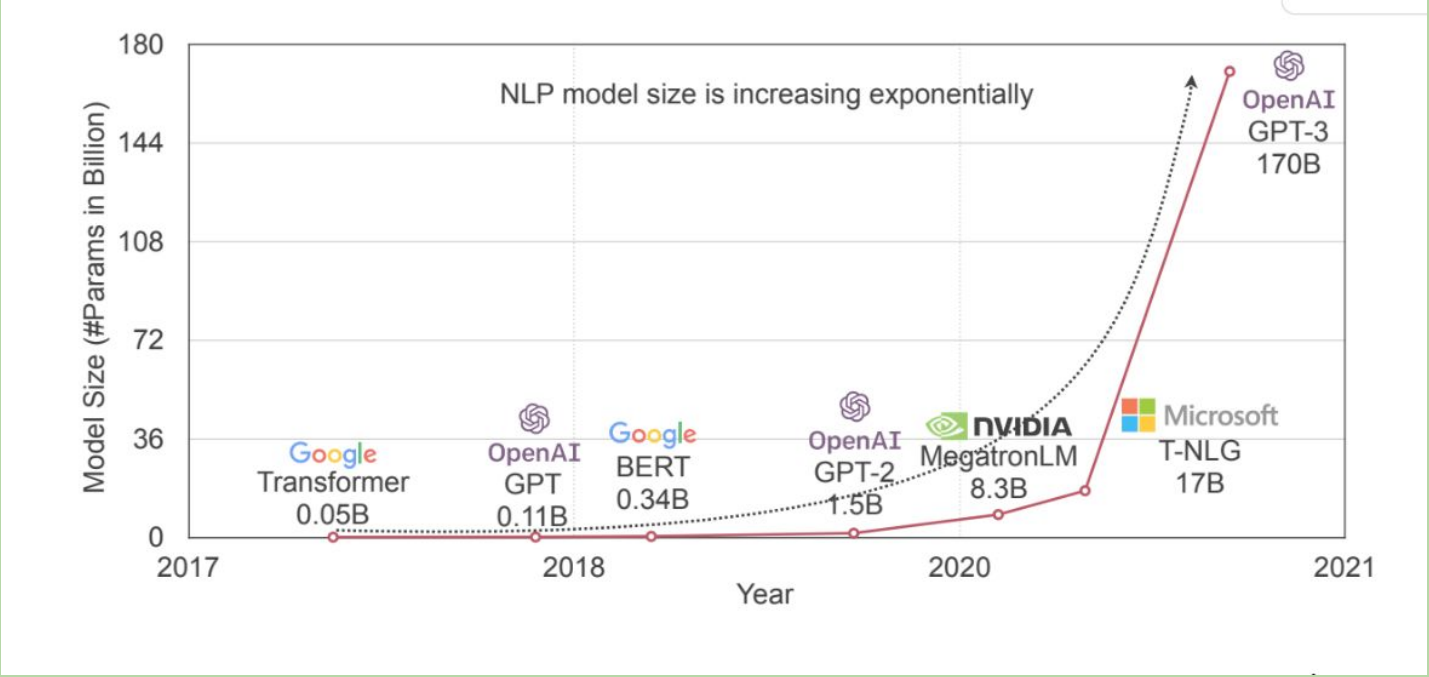


Examples of User Texts and AI generated Texts

BERT, RoBERTa, DistilBERT, TinyBERT

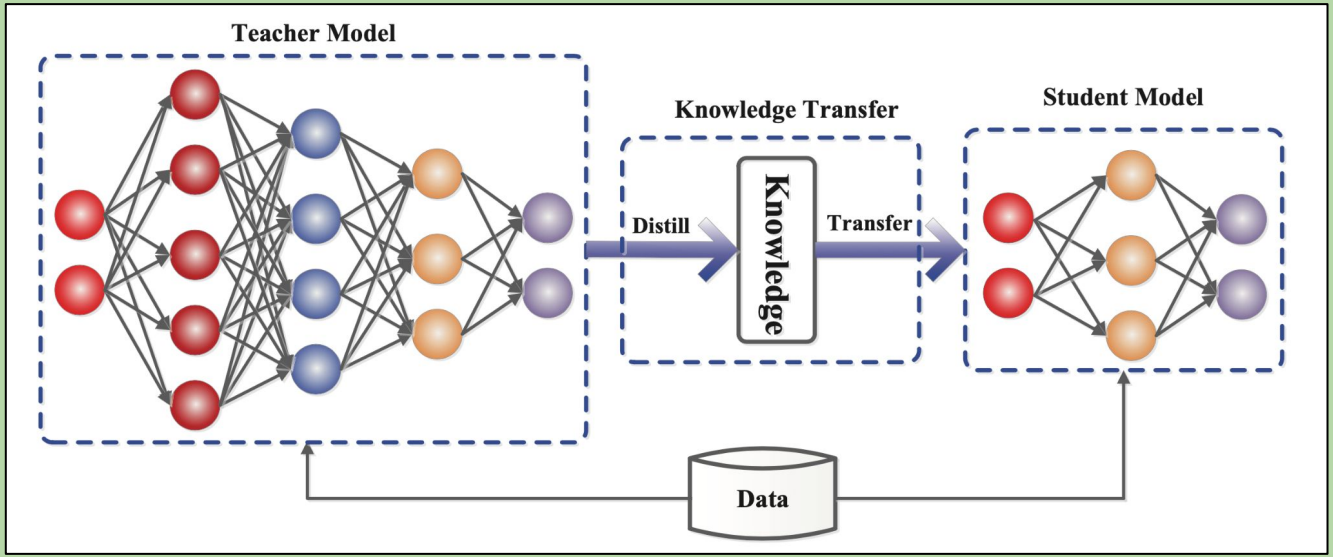
Distillation

AI Trends



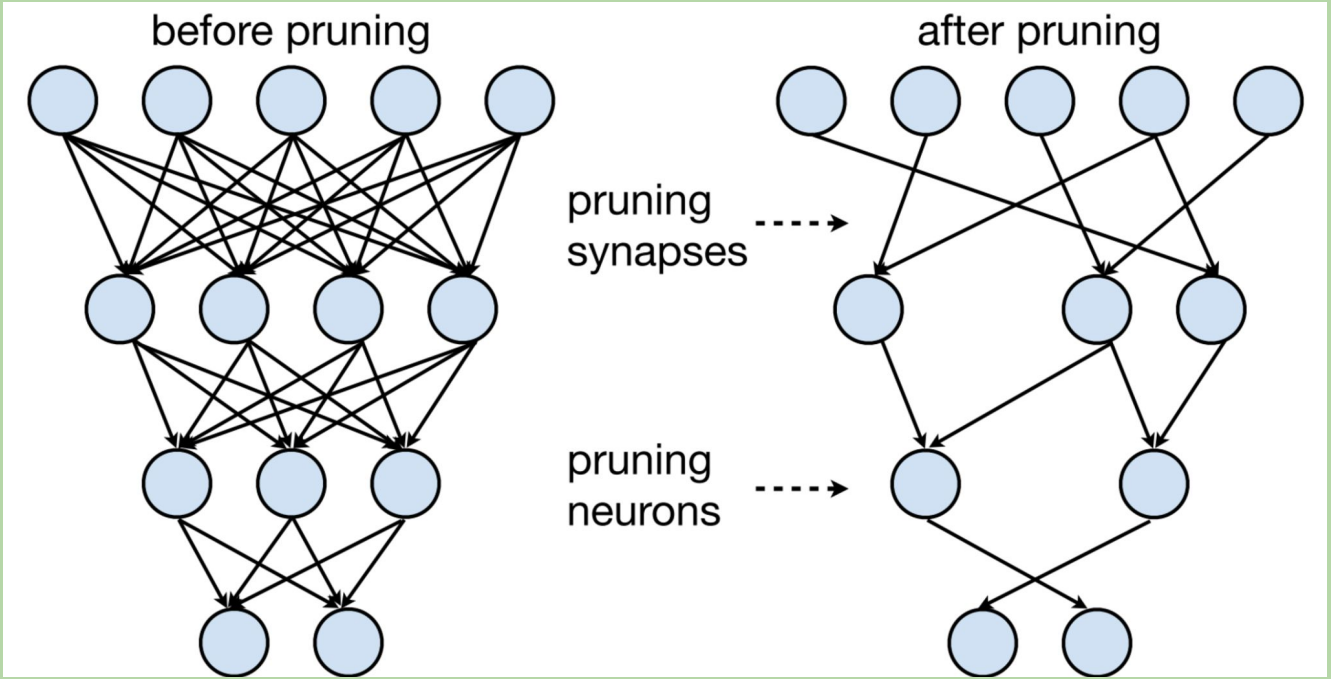
https://hanlab.mit.edu/projects/tinyml

Teacher-Student Architecture



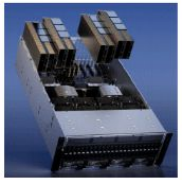
2006.05525

Sparse and Pruning (Spruning)



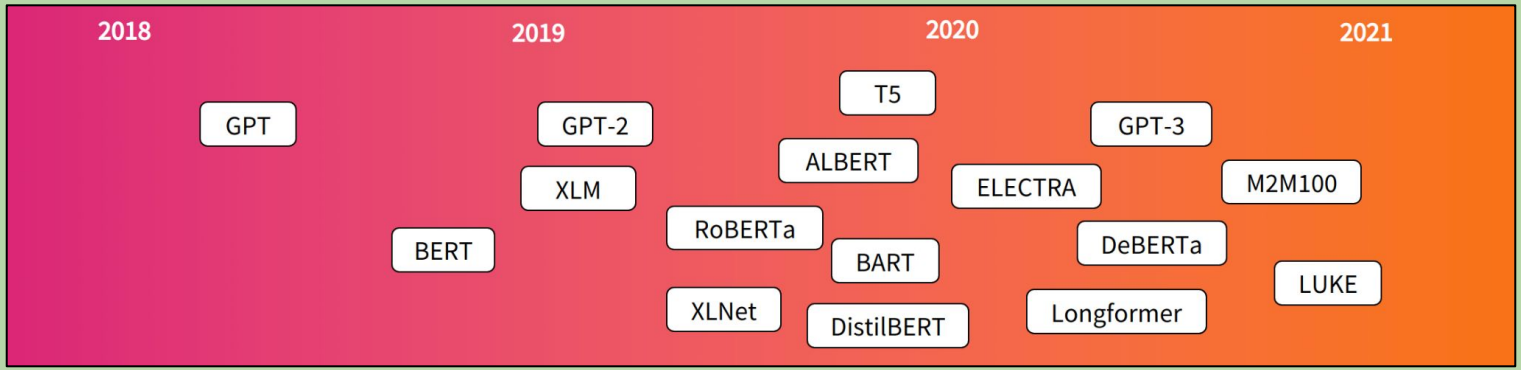
https://hanlab.mit.edu/projects/tinyml

AI to TinyAI

	Cloud AI	Mobile AI		Tiny AI
				
	Nvidia H100[1]	Apple M2 Ultra[2]	Qualcomm S8Gen2[3]	STM32F746NG[4]
Memory	80GB	64-192 GB	8-24 GB	320kB
Storage	~TB/PB	~GB/TB	~GB	1MB
Computing power	1,979 TOPS	31.6 TOPS	36 TOPS	462 MOPS

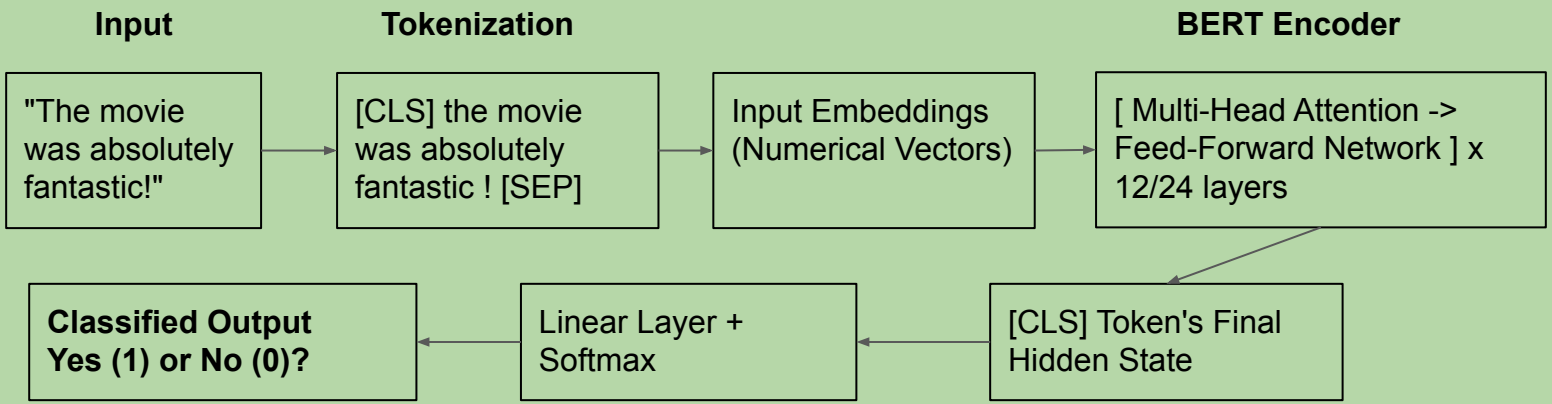
https://hanlab.mit.edu/projects/tinyml

BERT Timeline



BERT

Token	Example Embedding (truncated for clarity)
[CLS]	[0.11, -0.23, ..., 0.04]
the	[0.02, 0.14, ..., -0.03]
movie	[-0.25, 0.11, ..., 0.08]

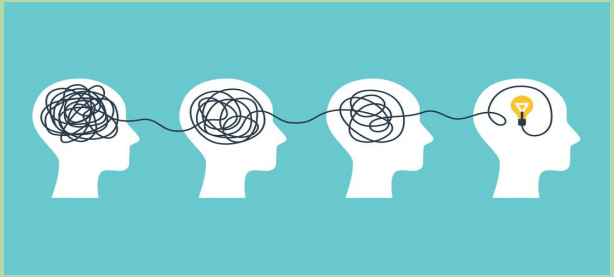


Class	Logit (pre-softmax)	Softmax Output
Negative (0)	-1.2	0.12
Positive (1)	2.4	0.88

[0.21, -0.35, ..., 0.18] (768-d vector)

Self-Attention Mechanism

- A mechanism that helps a model focus on relevant words in a sentence.
- Each word learns to "pay attention" to other words to understand context.



Input X

↓

Dimension Size -> d (768 for BERT Small, 1024 for BERT Large)

Number of Heads -> h

d_k = d/h

Linear → Q (Queries)

Linear → K (Keys)

Linear → V (Values)

↓

$\text{Attention}(Q, K, V) = \text{softmax}(QK^T / \sqrt{d\_k}) \times V$

↓

Output (Attention X V)

The FBI is chasing a criminal on the run .

The FBI is chasing a criminal on the run .

The FBI is chasing a criminal on the run .

The FBI is chasing a criminal on the run .

The FBI is chasing a criminal on the run .

The FBI is chasing a criminal on the run .

The FBI is chasing a criminal on the run .

The FBI is chasing a criminal on the run .

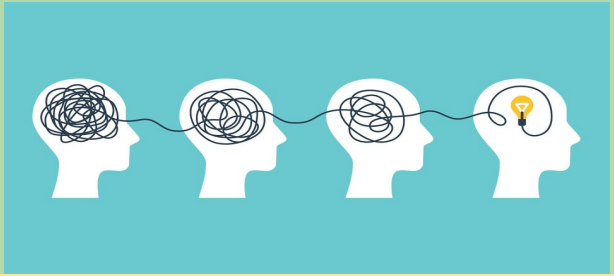
The FBI is chasing a criminal on the run .

https://blog.paperspace.com/seq-to-seq-attention-mechanism-keras/

https://www.linkedin.com/pulse/metacognition-self-regulation-perplexing-phrase-classroom-allen/

Self-Attention Mechanism

- A mechanism that helps a model focus on relevant words in a sentence.
- Each word learns to "pay attention" to other words to understand context.



For each word:

- **Query (Q)**: What am I looking for?
- **Key (K)**: What do I offer?
- **Value (V)**: What information do I carry?

Input X

↓

Dimension Size -> d (768 for BERT Small, 1024 for BERT Large)

Number of Heads -> h

d_k = d/h

Linear → Q (Queries)

Linear → K (Keys)

Linear → V (Values)

↓

$$\text{Attention}(Q, K, V) = \text{softmax}(QK^T / \sqrt{d\_k}) \times V$$

↓

Output (Attention X V)

The FBI is chasing a criminal on the run .

The FBI is chasing a criminal on the run .

The FBI is chasing a criminal on the run .

The FBI is chasing a criminal on the run .

The FBI is chasing a criminal on the run .

The FBI is chasing a criminal on the run .

The FBI is chasing a criminal on the run .

The FBI is chasing a criminal on the run .

The FBI is chasing a criminal on the run .

https://blog.paperspace.com/seq-to-seq-attention-mechanism-keras/
https://www.linkedin.com/pulse/metacognition-self-regulation-perplexing-phrase-classroom-allen/

Self-Attention Mechanism

1. Input → Token Embeddings
2. Linear projections → Q, K, V vectors
3. Compute scores: $Q \times K^T / \sqrt{d\_k}$
4. Apply softmax → get attention weights
5. Multiply weights by V → get new word vector
6. Output: **contextualized embeddings**

Input X

↓

Dimension Size -> d (768 for BERT Small, 1024 for BERT Large)

Number of Heads -> h

d_k = d/h

Linear → Q (Queries)

Linear → K (Keys)

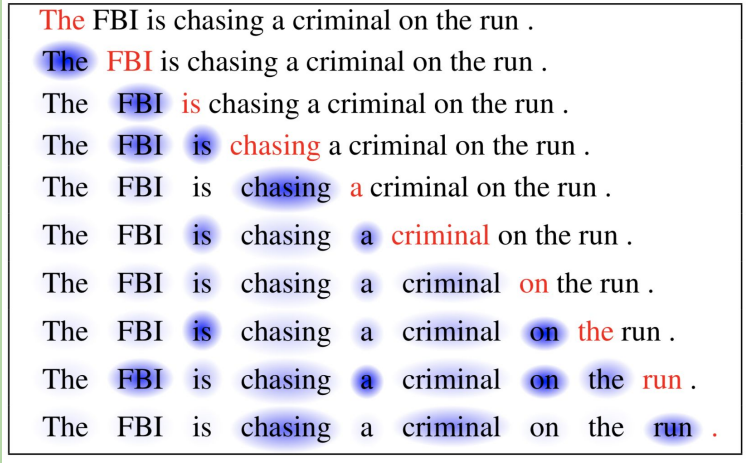
Linear → V (Values)

↓

$\text{Attention}(Q, K, V) = \text{softmax}(QK^T / \sqrt{d\_k}) \times V$

↓

Output (Attention X V)



https://blog.paperspace.com/seq-to-seq-attention-mechanism-keras/

https://www.linkedin.com/pulse/metacognition-self-regulation-perplexing-phrase-classroom-allen/

Self-Attention Mechanism

1. Input → Token Embeddings
2. Linear projections → Q, K, V vectors
3. Compute scores: $Q \times K^T / \sqrt{d\_k}$
4. Apply softmax → get attention weights
5. Multiply weights by V → get new word vector
6. Output: **contextualized embeddings**

Input X

↓

Dimension Size -> d (768 for BERT Small, 1024 for BERT Large)

Number of Heads -> h

d_k = d/h

Linear → Q (Queries)

Linear → K (Keys)

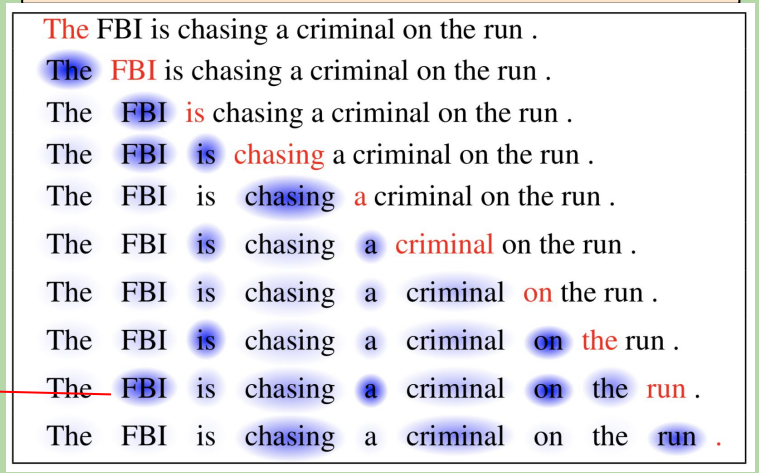
Linear → V (Values)

↓

$\text{Attention}(Q, K, V) = \text{softmax}(QK^T / \sqrt{d\_k}) \times V$

↓

Output (Attention X V)



They are chasing criminal on a run

https://blog.paperspace.com/seq-to-seq-attention-mechanism-keras/

https://www.linkedin.com/pulse/metacognition-self-regulation-perplexing-phrase-classroom-allen/

Self-Attention Mechanism

1. Input → Token Embeddings
2. Linear projections → Q, K, V vectors
3. Compute scores: $Q \times K^T / \sqrt{d\_k}$
4. Apply softmax → get attention weights
5. Multiply weights by V → get new word vector
6. Output: **contextualized embeddings**

Enables parallel computation, better performance than RNNs!

They are chasing criminal on a run

Input X

↓

Dimension Size -> d (768 for BERT Small, 1024 for BERT Large)

Number of Heads -> h

d_k = d/h

Linear → Q (Queries)

Linear → K (Keys)

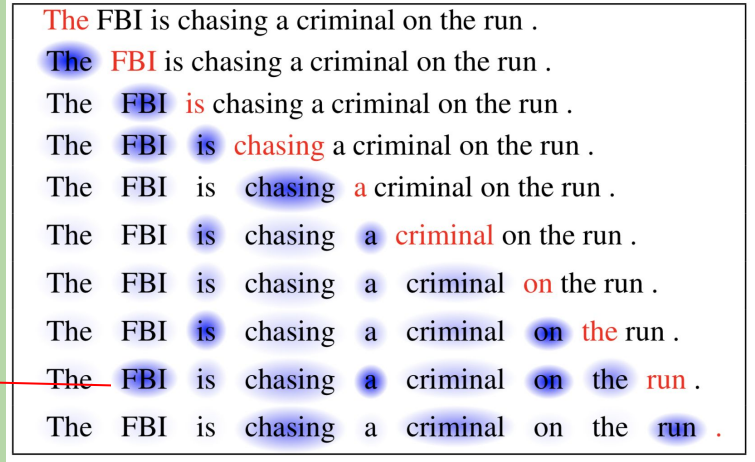
Linear → V (Values)

↓

$\text{Attention}(Q, K, V) = \text{softmax}(QK^T / \sqrt{d\_k}) \times V$

↓

Output (Attention X V)



https://blog.paperspace.com/seq-to-seq-attention-mechanism-keras/

https://www.linkedin.com/pulse/metacognition-self-regulation-perplexing-phrase-classroom-allen/

Multi-Head Attention Mechanism

Multi-head attention **uses multiple self-attention mechanisms (heads)** in parallel. Each head has its own learned linear transformations for Q, K, and V.

Input X

↓

Dimension Size → d (e.g., 768 for BERT Small, 1024 for BERT Large)

Number of Heads → h (e.g., 12)

d_k = d / h (e.g., 64 if d = 768 and h = 12)

Split into h heads

↓

For each head $i \in \{1, ..., h\}$

Linear → Q_i (Query for head i)

Linear → K_i (Key for head i)

Linear → V_i (Value for head i)

$\text{Attention}_i(Q_i, K_i, V_i) = \text{softmax}(Q_i K_i^T / \sqrt{d\_k}) \times V_i$

↓

Concatenate all head outputs: [head_1; head_2; ...; head□]

↓

Final Linear Projection (W^o)

↓

Multi-Head Attention Output

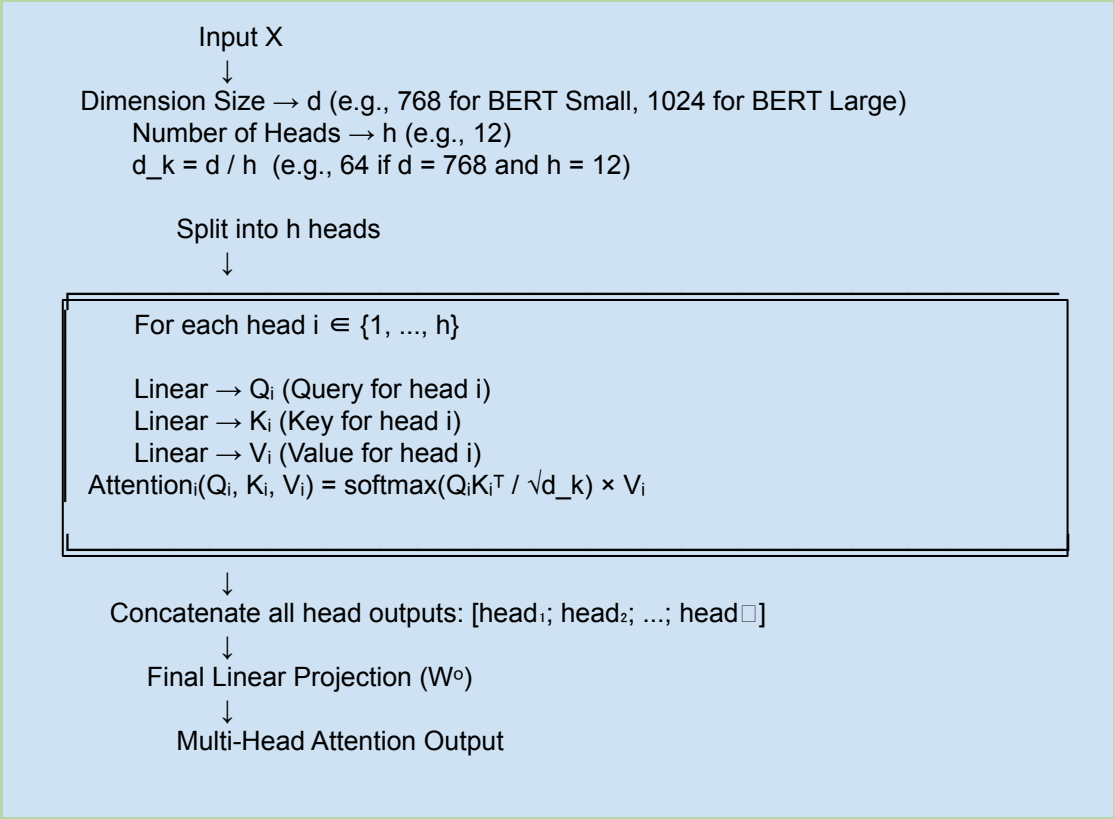
https://blog.paperspace.com/seq-to-seq-attention-mechanism-keras/

https://www.linkedin.com/pulse/metacognition-self-regulation-perplexing-phrase-classroom-allen/

Multi-Head Attention Mechanism

Multi-head attention **uses multiple self-attention mechanisms (heads)** in parallel. Each head has its own learned linear transformations for Q, K, and V.

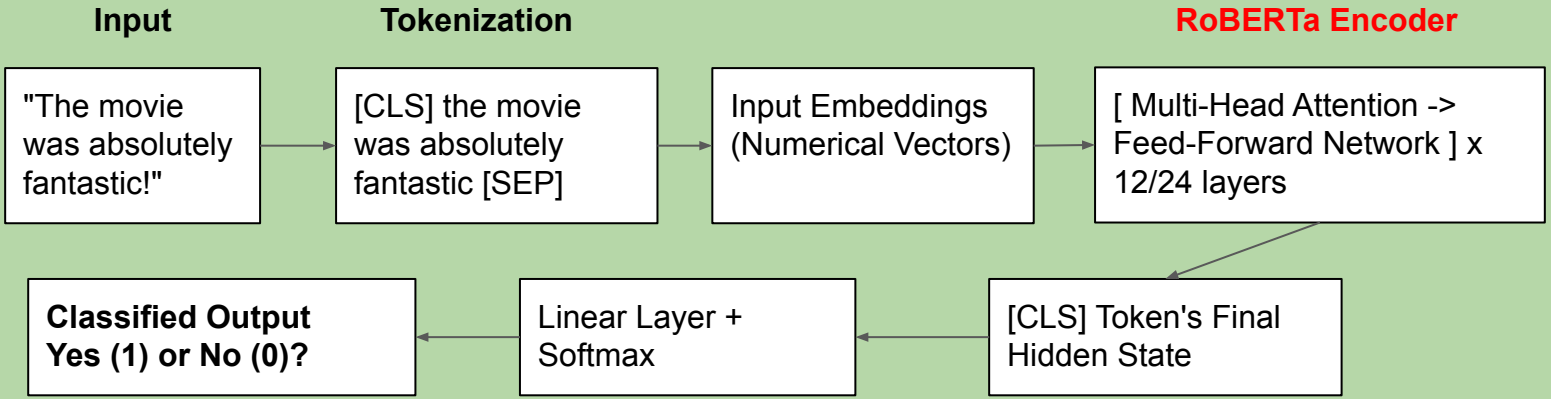
1. **Input** → Token embeddings
2. **Linear Projections** → → Q, K, V vectors
3. **Per Head Attention**: For each head separately:
   - ➔ Compare each token with others using Q and K
   - ➔ Apply softmax to get attention weights
   - ➔ Use these weights to combine V vectors → result is the output for that head
4. **Concatenate Outputs** → Combine the outputs from all heads into one long vector.
5. **Final Linear Layer**
6. **Output**



https://blog.paperspace.com/seq-to-seq-attention-mechanism-keras/
https://www.linkedin.com/pulse/metacognition-self-regulation-perplexing-phrase-classroom-allen/

RoBERTa

Token	Example Embedding (truncated for clarity)
[CLS]	[0.11, -0.23, ..., 0.04]
the	[0.02, 0.14, ..., -0.03]
movie	[-0.25, 0.11, ..., 0.08]



Class	Logit (pre-softmax)	Softmax Output
Negative (0)	-1.2	0.12
Positive (1)	2.4	0.88

Pooling

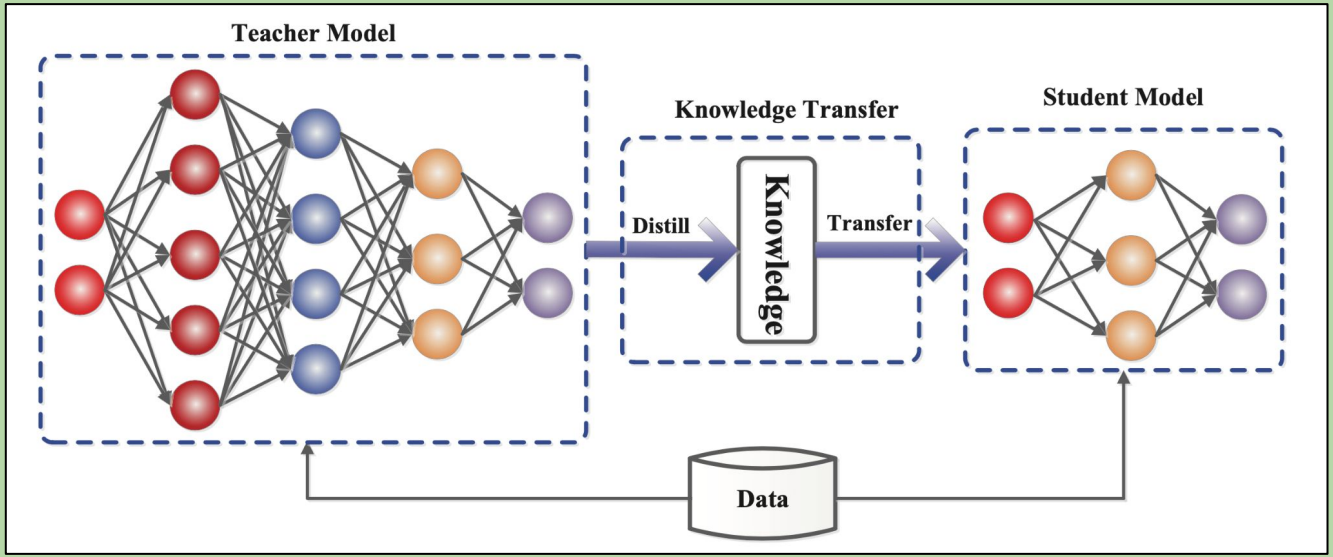
[0.21, -0.35, ..., 0.18] (768-d vector)

BERT Vs RoBERTa

Model	BERT (Bidirectional Encoder Representations from Transformers)	RoBERTa (A Robustly Optimized BERT Approach)
Training Data	Trained on BookCorpus & English Wikipedia (~16GB).	Trained on a much larger corpus including BookCorpus, Wikipedia, CC-News, OpenWebText, & Stories (~160GB).
Pre-training Tasks	1. Masked Language Model (MLM) 2. Next Sentence Prediction (NSP)	1. Masked Language Model (MLM) only.The NSP task was removed.
Masking Strategy	Static Masking: Data is masked once during preprocessing and never changed.	Dynamic Masking: Data is duplicated and masked 10 times. A different mask is chosen for each training epoch.
Batch Size	Trained with a batch size of 256 sequences.	Trained with a much larger batch size of 8,000 sequences.
Tokenizer	WordPiece vocabulary of 30,000 tokens.	Byte-level Byte-Pair Encoding (BPE) vocabulary of 50,000 tokens.
Performance	Strong baseline performance.	Significantly outperforms BERT on nearly all NLP benchmarks (like GLUE and SQuAD).

DistilBERT

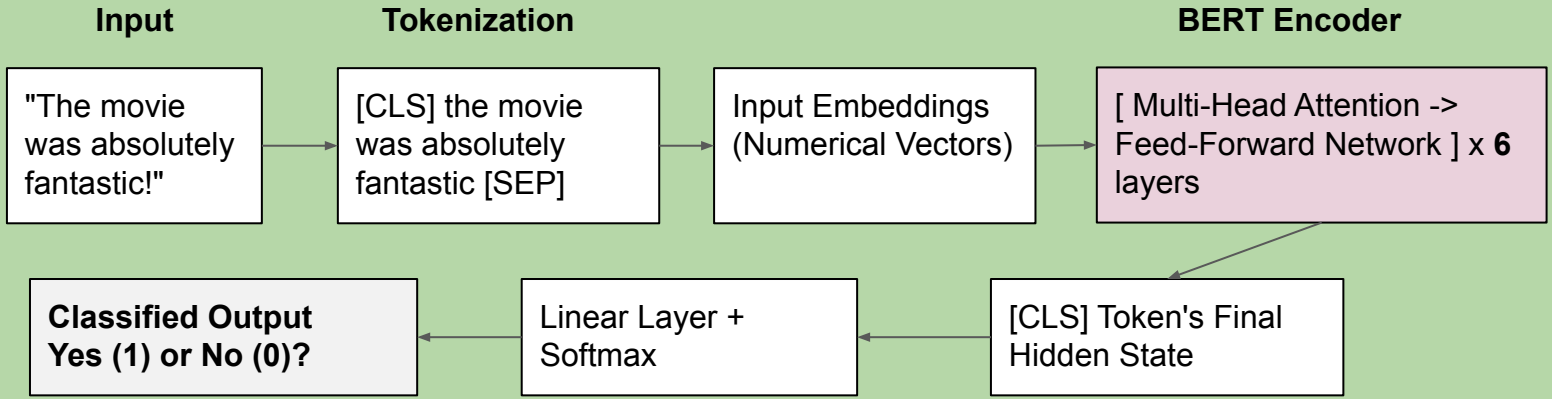
BERT - Base **DistilBERT**



2006.05525

DistilBERT

Token	Example Embedding (truncated for clarity)
[CLS]	[0.11, -0.23, ..., 0.04]
the	[0.02, 0.14, ..., -0.03]
movie	[-0.25, 0.11, ..., 0.08]



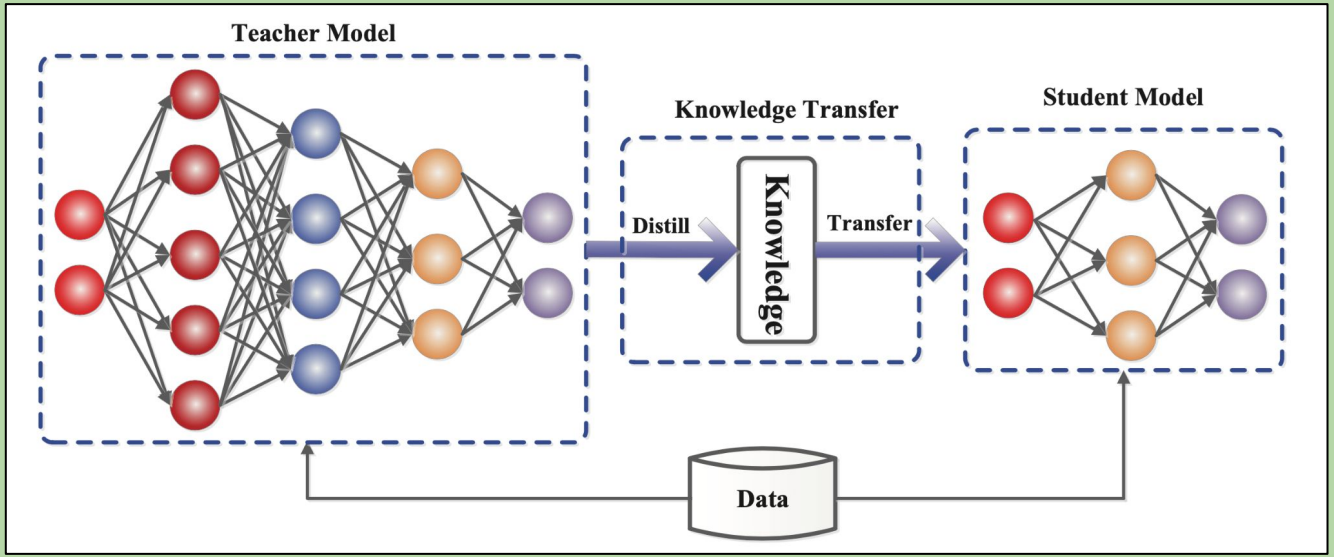
Class	Logit (pre-softmax)	Softmax Output
Negative (0)	-1.2	0.12
Positive (1)	2.4	0.88

Pooling

[0.21, -0.35, ..., 0.18] (768-d vector)

TinyBERT

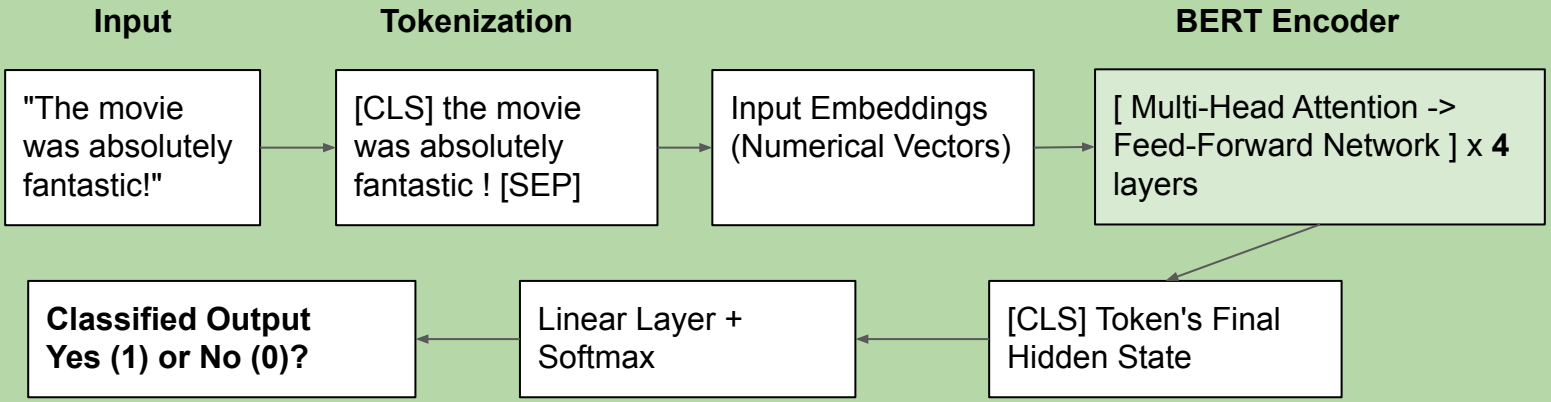
BERT - Base and Large **TinyBERT**



2006.05525

TinyBERT

Token	Example Embedding (truncated for clarity)
[CLS]	[0.11, -0.23, ..., 0.04]
the	[0.02, 0.14, ..., -0.03]
movie	[-0.25, 0.11, ..., 0.08]



Class	Logit (pre-softmax)	Softmax Output
Negative (0)	-1.2	0.12
Positive (1)	2.4	0.88

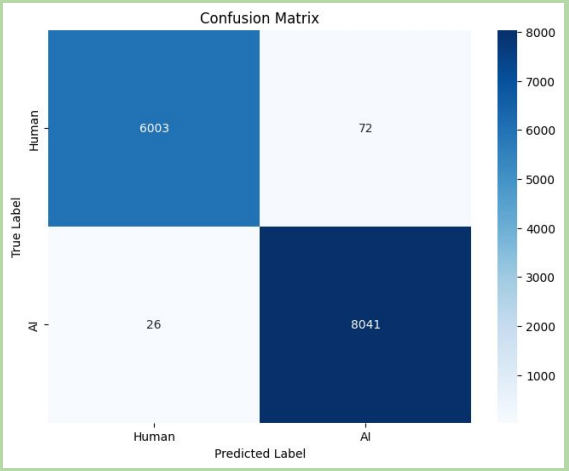
Pooling

[0.21, -0.35, ..., 0.18] (768-d vector)

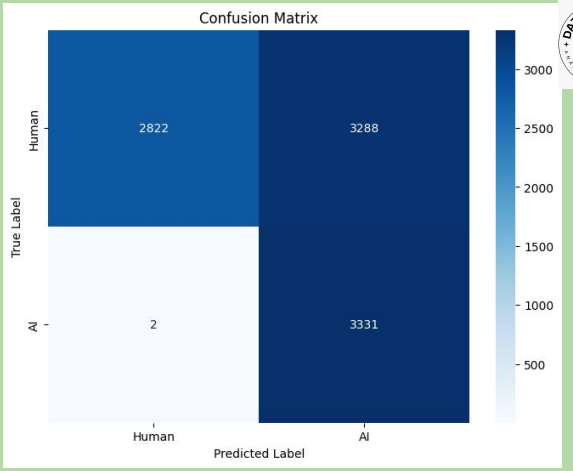
BERT Models

Model	Key Innovation	Approx. Parameters	Relative Size	Relative Speed	Relative Performance (GLUE)	Core Use Case
BERT-Base	Deep Bidirectionality (Masked Language Modelling + Next Sentence Prediction)	110M	1x	1x	Baseline	General Purpose NLP, Fine-tuning
RoBERTa-Base	Optimized Pre-training (Dynamic Mask, No NSP, More Data)	125M	~1.1x	~1x	~3-5% > BERT	State-of-the-Art Performance, Research
DistilBERT	Knowledge Distillation (Triple Loss)	66M	~0.6x	~1.6x Faster	~97% of BERT	Production Systems, Real-time APIs
TinyBERT (4L)	Two-Stage, Layer-wise Distillation	14.5M	~0.13x	~9.4x Faster	~96% of BERT	Edge Devices, Mobile Applications

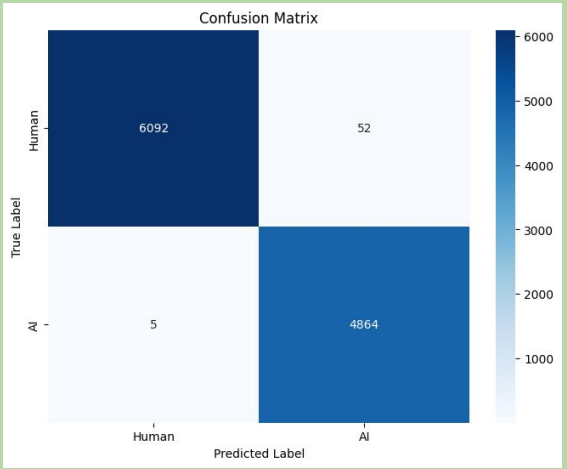
Confusion Matrix



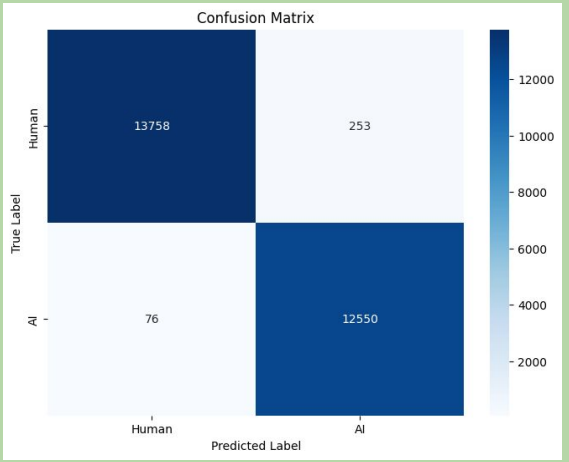
(a) BERT



(b) RoBERTa



(c) DistilBERT

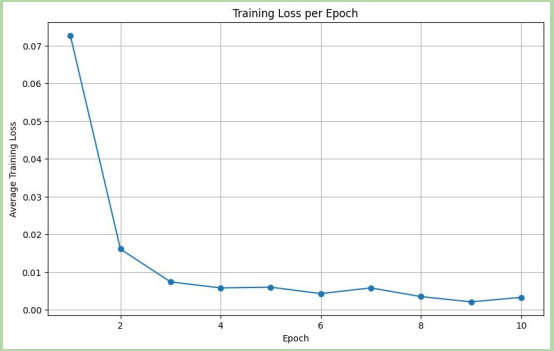


(d) TinyBERT

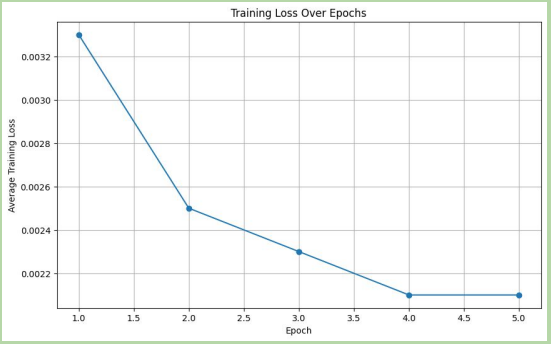
Training Loss

(a) BERT

(b) RoBERTa



(c) DistilBERT

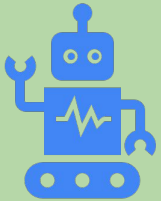


(d) TinyBERT

Abstract

The rise of generative AI in writing raises concerns about the difficulty of distinguishing AI generated from human written content. This brings about risks in trust, academic integrity, and skill development.

Our study investigates AI text detection using four transformer based classifiers: BERT, RoBERTa, DistilBERT, and TinyBERT. They are trained on a shared dataset and each evaluated for: Classification accuracy, efficiency, and inference speed.



We analyzed the trade-offs between model size and detection reliability, offering insights into selecting the right model for practical scalable AI detection.

References

Alla, S. (2021, April 9). *Attention Mechanisms With Keras | Paperspace Blog*. Paperspace by DigitalOcean Blog. https://blog.paperspace.com/seq-to-seq-attention-mechanism-keras/

Allen, L. (2023, July 5). *Metacognition and self-regulation. It sounds horribly complicated, doesn't it? It's the sort phrase, which when uttered during staff meetings, causes educators to yawn, panic or roll their eyes.* Linkedin.com. https://www.linkedin.com/pulse/metacognition-self-regulation-perplexing-phrase-classroom-allen/

Gou, J., Yu, B., Maybank, S., & Tao, D. (2021). Knowledge Distillation: A Survey. *Int. J. Comput. Vis.* https://doi.org/10.1007/s11263-021-01453-z

nirmalgaud. (2024, January 27). *Human vs AI Text*. Kaggle.com; Kaggle. https://www.kaggle.com/code/nirmalgaud/human-vs-ai-text

Theocharopoulos, P. C., Anagnostou, P., Tsoukala, A., Georgakopoulos, S. V., Tasoulis, S. K., & Plagianakos, V. P. (2023). *Detection of Fake Generated Scientific Abstracts*. https://doi.org/10.1109/bigdataservice58306.2023.00011