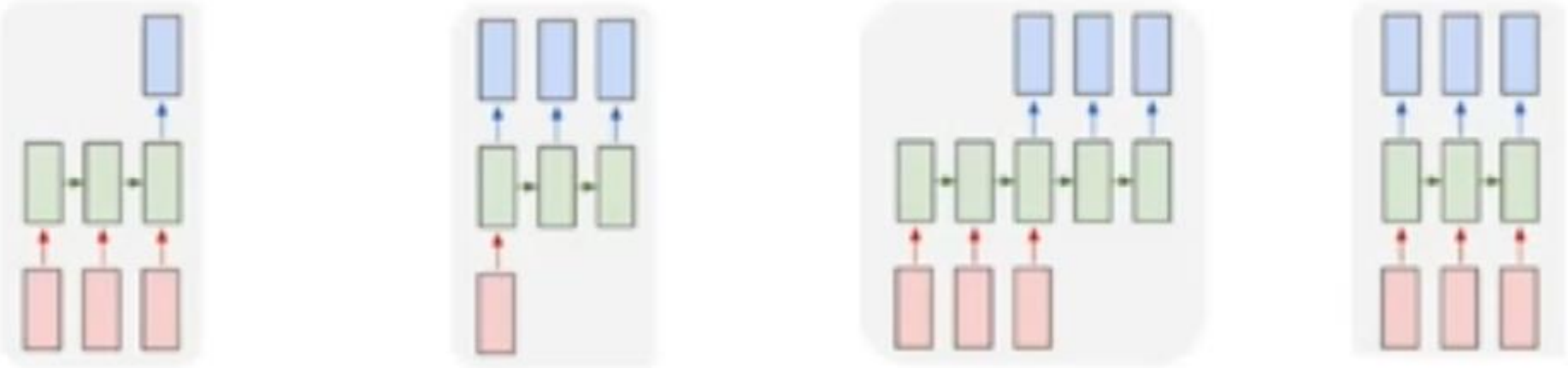
History of LLM

Dr. Muhammad Safyan

Sequence data type

- Synchronous and Asynchronous
  - Seq2Seq Models
  - Text summarization, Question answering, chatbot,



History of Sequence to Sequence Models

- Stage-1:
  - Encoder/Decoder
    - Machine Translation
- Stage-2:
  - Attention Model
- Stage-3:
  - Transformers
- Stage-4:
  - Transfer Learning
- Stage-5:
  - LLM
    - ChatGPT

Encoder/Decoder-Stage-1-2014

Claim seq2seq model does not work well.

Sequence to Sequence Learning with Neural Networks

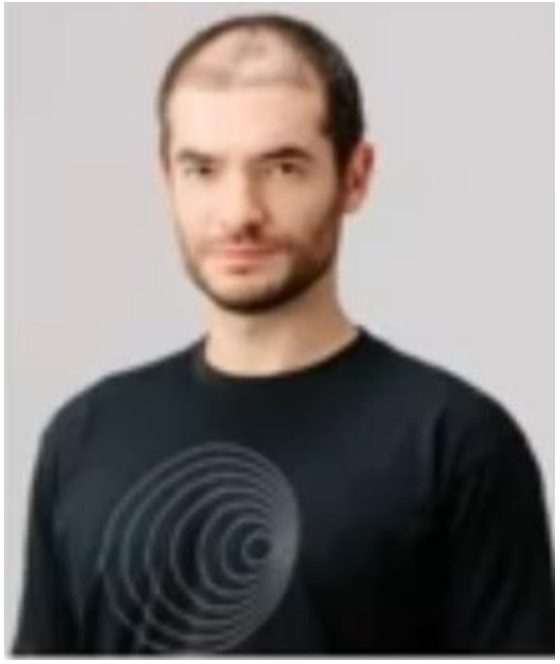
Ilya Sutskever
Google
ilyasu@google.com

Oriol Vinyals
Google
vinyals@google.com

Quoc V. Le
Google
qvl@google.com

Abstract

Deep Neural Networks (DNNs) are powerful models that have achieved excellent performance on difficult learning tasks. Although DNNs work well whenever large labeled training sets are available, they cannot be used to map sequences to sequences. In this paper, we present a general end-to-end approach to sequence learning that makes minimal assumptions on the sequence structure. Our method uses a multilayered Long Short-Term Memory (LSTM) to map the input sequence to a vector of a fixed dimensionality, and then another deep LSTM to decode the target sequence from the vector. Our main result is that on an English to French translation task from the WMT-14 dataset, the translations produced by the LSTM achieve a BLEU score of 34.8 on the entire test set, where the LSTM's BLEU



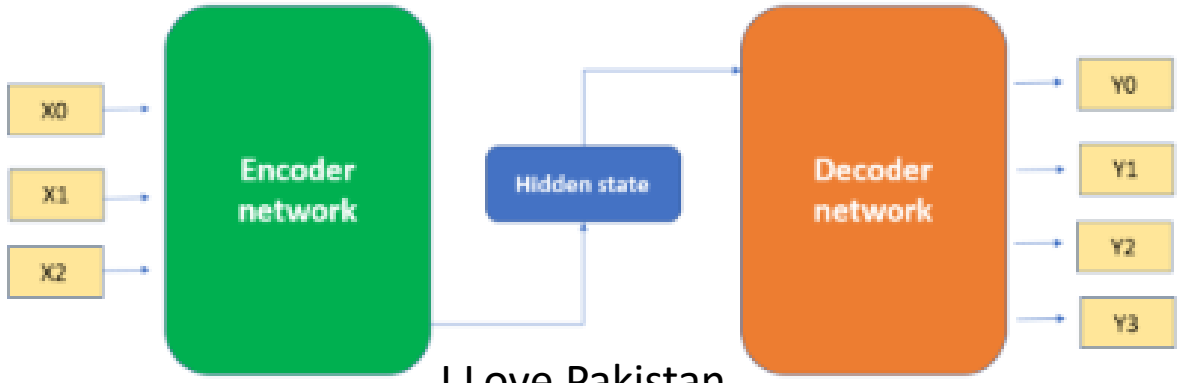
I Love Pakistan

میں پاکستان سے محبت کرتا ہوں۔

Encoder/Decoder Architecture

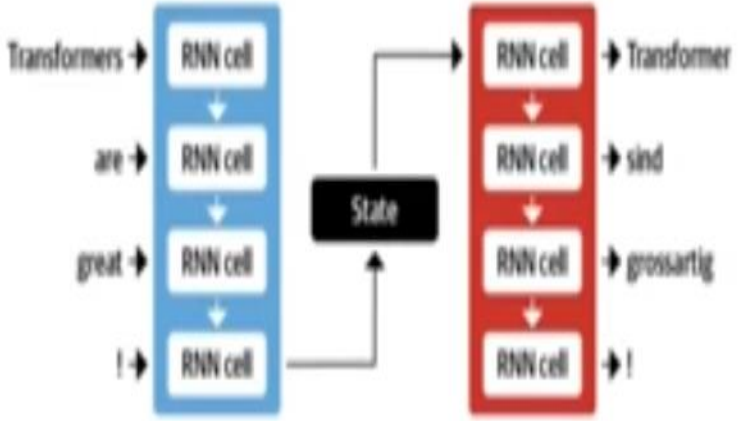
- Cofounder of OpenAI
- Propose a solution for Seq2Seq Model with different idea Encoder/Decoder Architecture.
- Language Translation
- Encoder has cell state(Ct,ht), represent your sentence in compress representation.

“ , he realized that the job offer was actually an incredible opportunity that would lead to significant personal and professional growth."



I Love Pakistan

میں پاکستان سے محبت کرتا ہوں۔



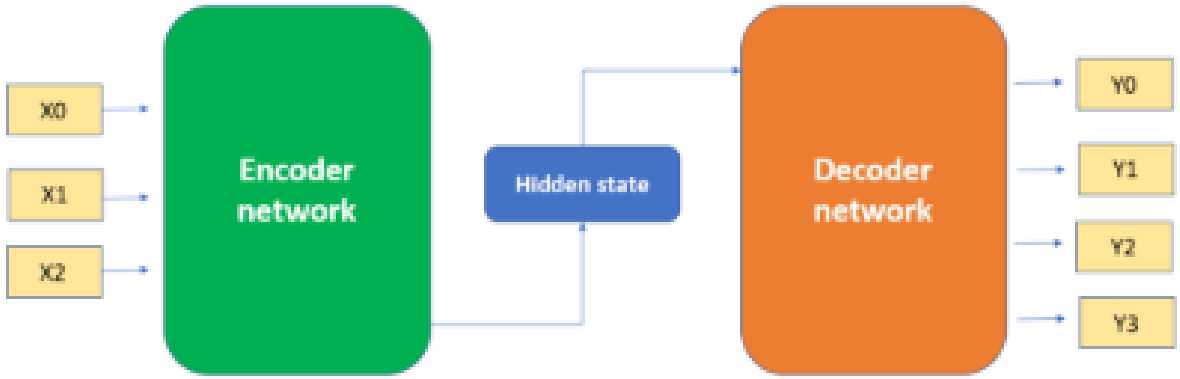
Compressed Representation

Limitation of Encoder/Decoder

- Encoder/Decoder is good for small sentence.
- Perform poor on long sentence.
- Full load is on context vector that convert your all information into one vector(compressed).
- Decoder is not working well on compressed vector.

Stage-2 Attention Mechanism

- Upon reflection, he recognized that declining the job offer had been an error, as it represented a considerable opportunity for both personal and professional development.
- Perform poor if the sentence is greater than 30 words.



Attention-2015

Published as a conference paper at ICLR 2015

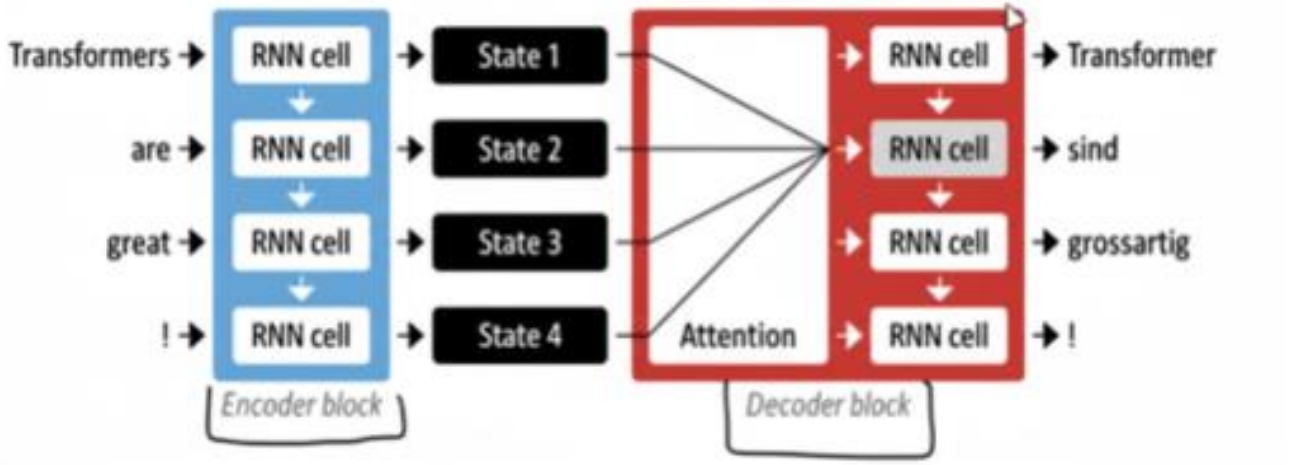
NEURAL MACHINE TRANSLATION BY JOINTLY LEARNING TO ALIGN AND TRANSLATE

Dzmitry Bahdanau
Jacobs University Bremen, Germany

KyungHyun Cho **Yoshua Bengio***
Université de Montréal

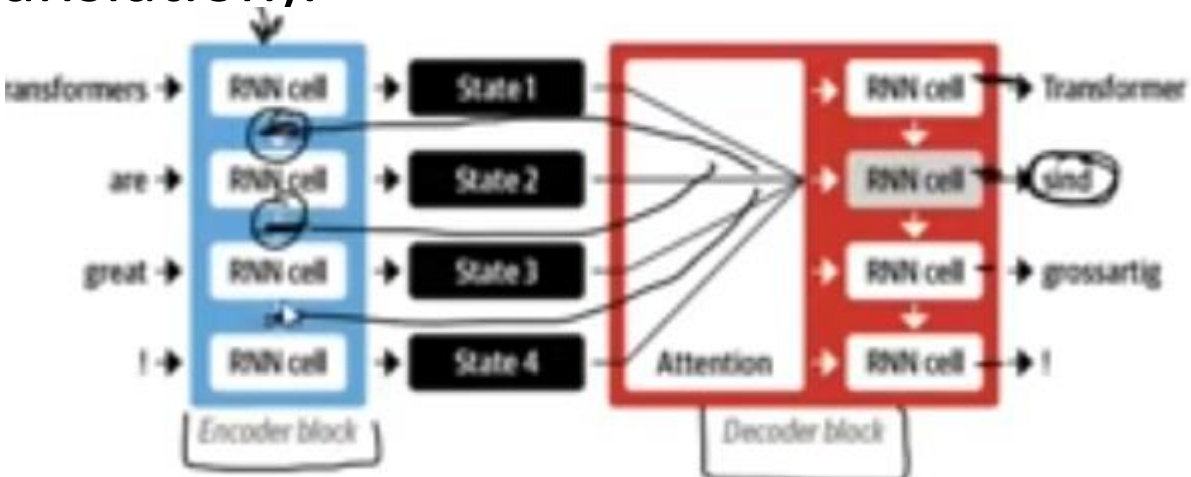
ABSTRACT

Neural machine translation is a recently proposed approach to machine translation. Unlike the traditional statistical machine translation, the neural machine translation aims at building a single neural network that can be jointly tuned to maximize the translation performance. The models proposed recently for neural machine translation often belong to a family of encoder–decoders and encode a source sentence into a fixed-length vector from which a decoder generates a translation. In this paper, we conjecture that the use of a fixed-length vector is a bottleneck in improving the performance of this basic encoder–decoder architecture, and propose to extend this by allowing a model to automatically (soft-)search for parts of a source sentence that are relevant to predicting a target word, without having to form these parts as a hard segment explicitly. With this new approach, we achieve a translation performance comparable to the existing state-of-the-art phrase-based system on the task of English-to-French translation. Furthermore, qualitative analysis reveals that the (soft-)alignments found by the model agree well with our intuition.



Attention

- Encoder is same , difference is in decoder
- We have the context of each word that can be used for decoder.
- But, which word should be used for which word in decoding process?
- Where attention come on screen.
- Attention is actually mechanism for dynamically searching the encoding word useful in decoding(Translation).
-



2015-2017

- **Drawback: calculate context vector at each step. (computational extensive) (n*m)→ complexity.**
- **Different Model invented to reduce this complexity.**
- **But actual problem is encoder →LSTM– work in sequential order.**

Transformers

Attention is all you need

- No need of LSTM/RNN → ditch
- , Google brain→ countless citations

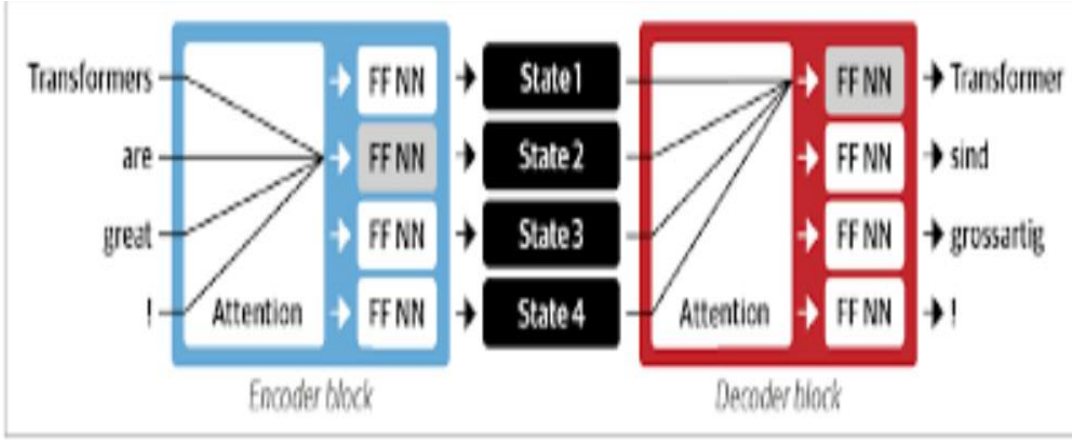


Figure 1-6. Encoder-decoder architecture of the original Transformer

Attention Is All You Need

Ashish Vaswani* Google Brain avaswani@google.com

Noam Shazeer* Google Brain noam@google.com

Niki Parmar* Google Research nikip@google.com

Jakob Uszkoreit* Google Research usz@google.com

Llion Jones* Google Research llion@google.com

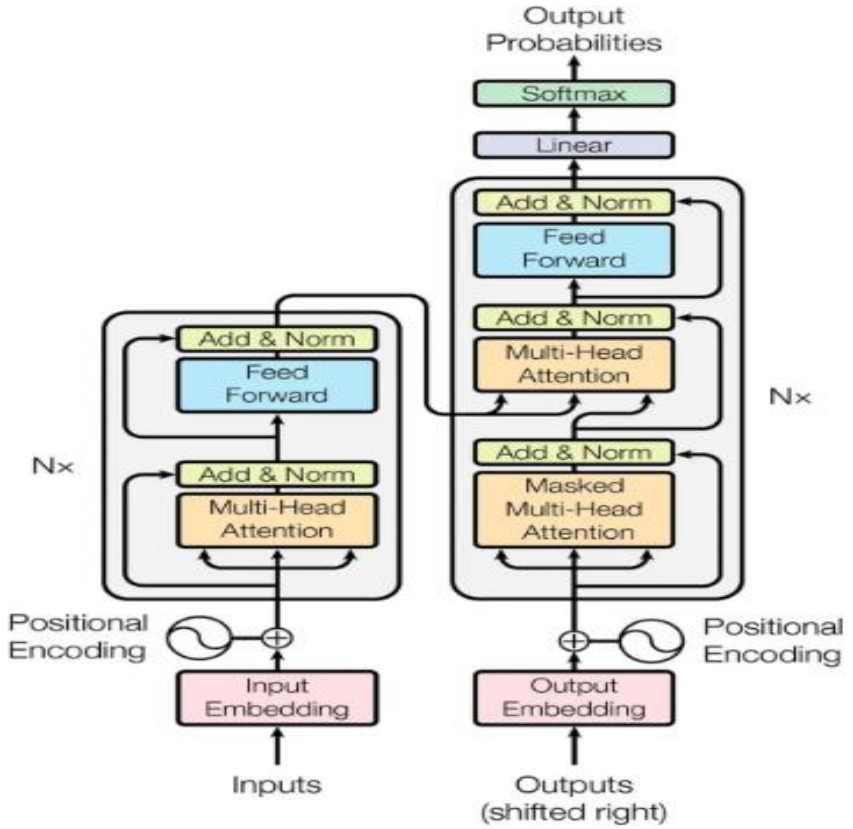
Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu

Łukasz Kaiser* Google Brain lukaszkaiser@google.com

Illia Polosukhin* ‡ illia.polosukhin@gmail.com

Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-



Attention is all you need

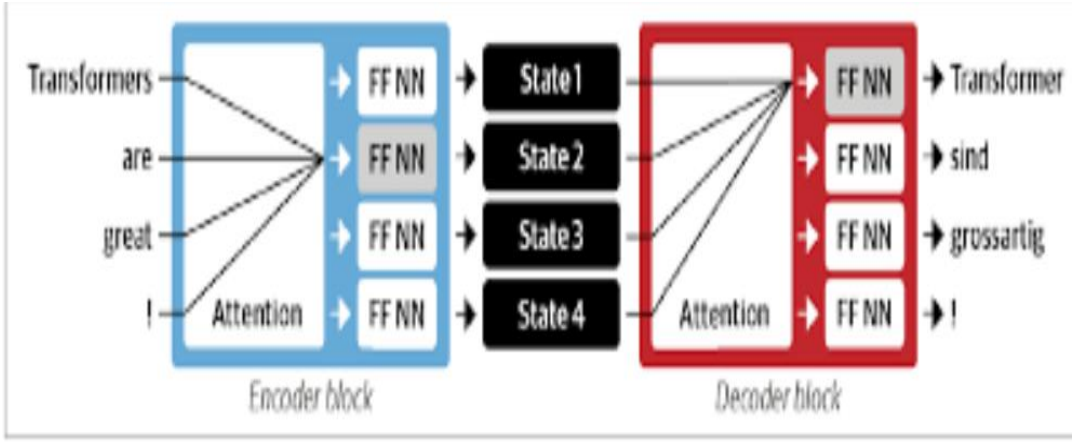


Figure 1-6. Encoder-decoder architecture of the original Transformer

- Wipe out LSTM
- No need of LSTM/RNN →
- Attention is all you need → self attention
- LSTM/RNN take one words at a time.
- Transformer can take all the words at a time → Training become Fast.
- Transformer change the Future of NLP.
- Easy to parallelize and Train the model in a fraction of time that Encode/Decoder Needed.
- Hardwre Requirement (GPU) also reduced.

Stage-4: Transfer Learning

- Transformer was revolutionary.
- Always produce the State of the Art results.
- Very Difficult to train from Scratch.
  - Hardware – GPU.
  - Time
  - Data
    - Every Time you don’t have the enough data.
    - That’s why Transformer are not used by every one.

Universal Language Modeling-2018

Universal Language Model Fine-tuning for Text Classification

Jeremy Howard*
fast.ai
University of San Francisco
j@fast.ai

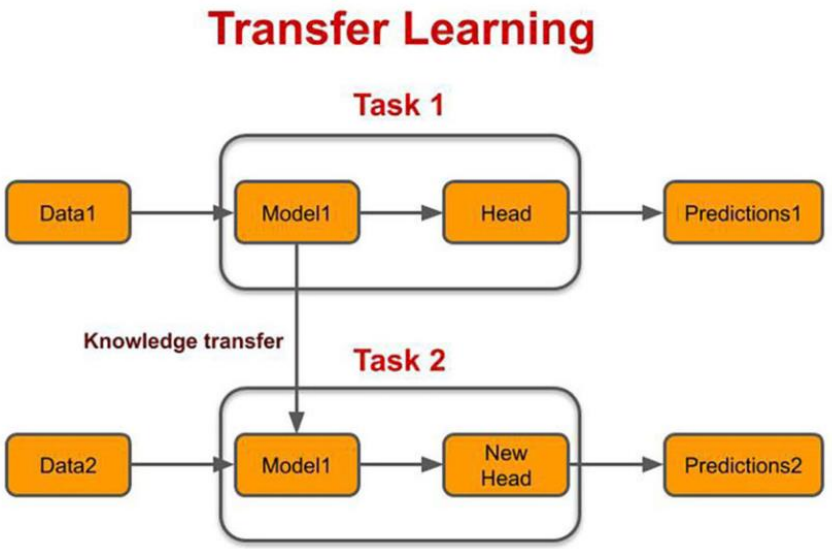
Sebastian Ruder*
Insight Centre, NUI Galway
Aylien Ltd., Dublin
sebastian@ruder.io

Abstract

Inductive transfer learning has greatly impacted computer vision, but existing approaches in NLP still require task-specific modifications and training from scratch. We propose Universal Language Model Fine-tuning (ULMFiT), an effective transfer learning method that can be applied to any task in NLP, and introduce techniques that are key for fine-tuning a language model. Our method significantly outperforms the state-of-the-art on six text classification tasks, reducing the error by 18-24% on the majority of datasets. Furthermore, with only 100 labeled examples, it matches the performance of training from scratch on 100× more data. We open-

While Deep Learning models have achieved state-of-the-art on many NLP tasks, these models are trained from scratch, requiring large datasets, and days to converge. Research in NLP focused mostly on *transductive* transfer (Blitzer et al., 2007). For *inductive* transfer, fine-tuning pre-trained word embeddings (Mikolov et al., 2013), a simple transfer technique that only targets a model's first layer, has had a large impact in practice and is used in most state-of-the-art models. Recent approaches that concatenate embeddings derived from other tasks with the input at different layers (Peters et al., 2017; McCann et al., 2017; Peters et al., 2018) still train the main task model from scratch and treat pretrained embeddings as fixed parameters, limiting their usefulness.

In light of the benefits of pretraining (Erhan et al., 2010), we should be able to do better than



Transfer Learning

- Transfer learning can be used in NLP domain.(UNLFIT)
  - Before that Transfer Learning can only be used in Vision Domain.
- **Transfer learning (TL)** is a technique in which **knowledge learned** from a task is re-used in order to boost performance on a related task.
- For example, for image classification, knowledge gained while learning to recognize cars could be applied when trying to recognize trucks.
- **Two- Steps**
- Step-1: Pre-training:
  - Train on large data set

Transfer Learning

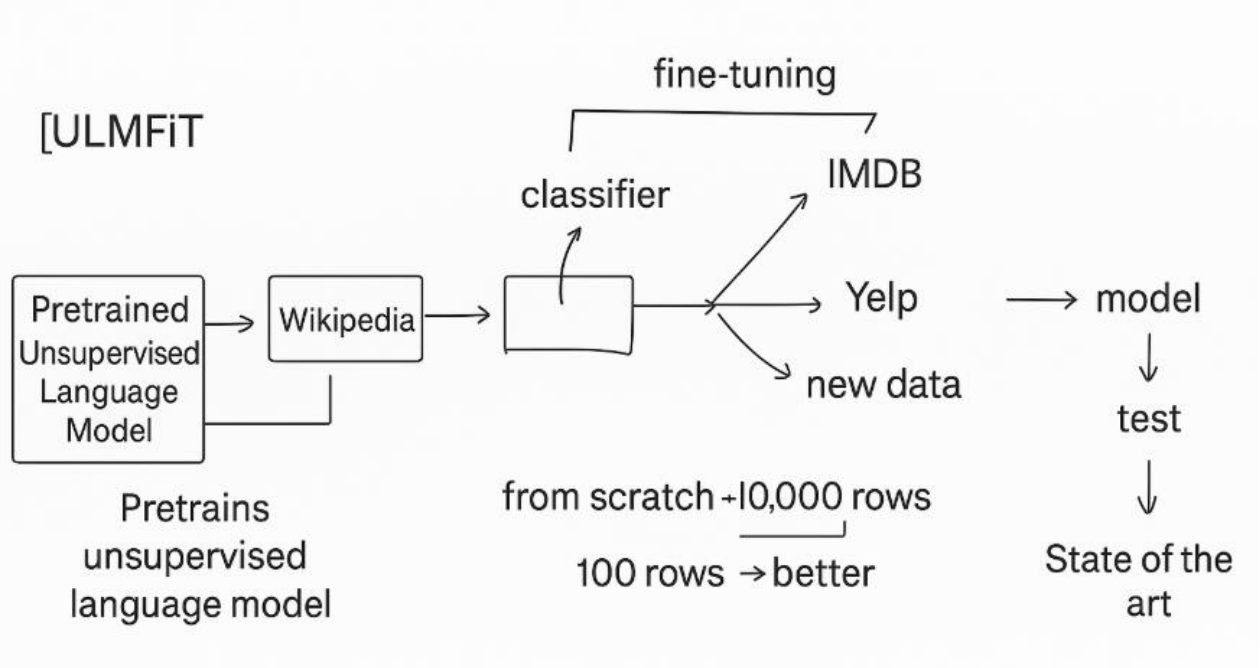
- Step-2: Fine Tuning
  - Take the Pre-trained Model .
  - Drop the later stage weights. And assign new random weights
  - And then train on you specific data.
  - Image Net:
    - Take a CNN Model , like RestNet
    - Pretrain the model on Image Net data.
      - it will learn all the edges and other features
    - Fine tune that Cat vs Dog.
- Why it could not applied in NLP domain.
  - **Task specific-various task**
    - Sentiment analysis, NER, PoS, Machine Trasnlation
    - Lack of data
- ULMFit solve this problem

Transfer Learning

- They did not use the Pre-train Machine Translation
- Used Language Modeling
  - LM is NLP task in which learn the machine to predict next word.
    - I live in Pakistan , capital of Pakistan is -------
- Benefits of Language Modeling:
  - Rich Feature Learning (learn feature , semantic, context , world knowledge)
    - The hotel was exceptionally clean yet the service was ------
  - If a model learn Language Modeling. It can be use for many other purposes.
    Like Text classification/QA/ Text summarization/ NER/PoS
- Huge availability of data
  - Englis | Urdu
    - You need Label data

Transfer Learning

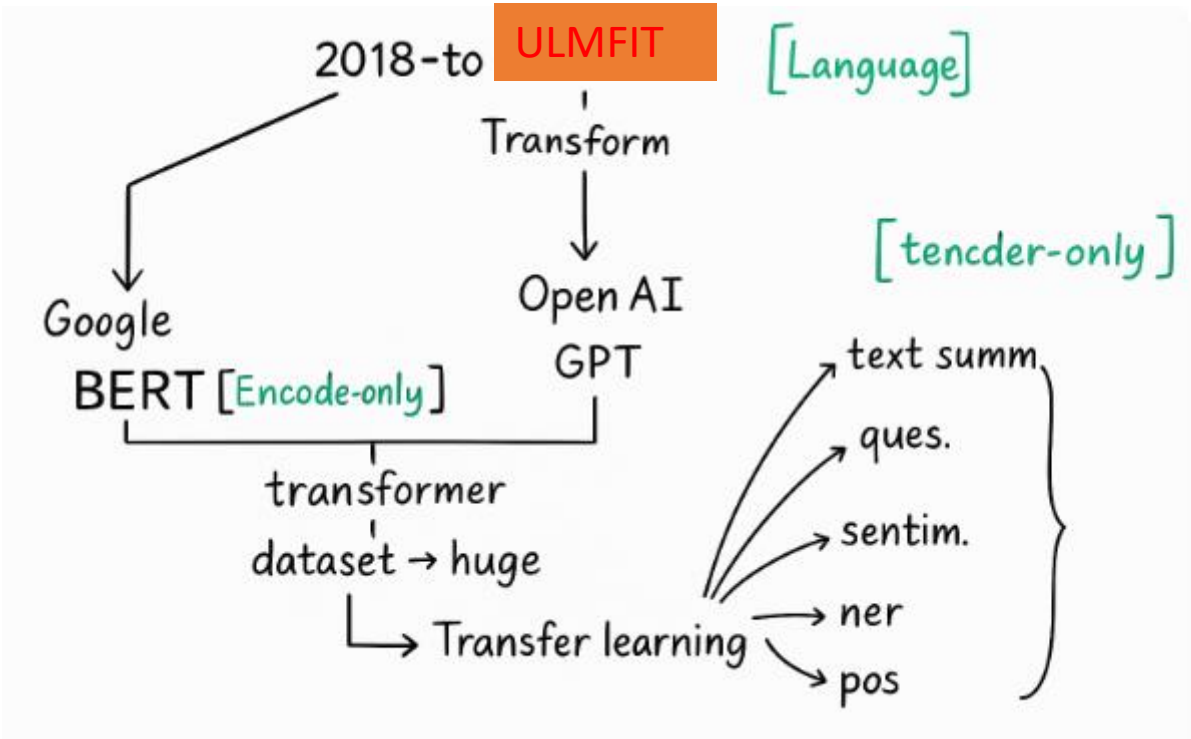
- You don’t need any labeled data.
  - Pick any pdf and extract data.
  - Training become unsupervised per training
- ULMFIT



- * Did not use Transformer

Transfer Learning

- Attention is you all need (Transformer)- 2017.
- ULMFit-2018
- Worked independently. But parallel
- One give the Transformer architecture
- And other give the Transfer Learning idea.



Stage-5 LLMs

- GPT→ GPT-2 → GPT3
- There so huge Number of parameter, dataset, People called in LLM.
- LM become LLMs
- Data Billions
  - GPT-3 was trained on 45 TBs
- Training Hard ware
  - Cluster of GPU
    - Super Computer → 1000s of NVIDIA GPU.
- Training Times:
  - Days →weeks
- Cost
  - Experts, electricity
- Trained by the:
  - Large companies, Govt, Institute
- Energy comsumptions

Chat-GPTs

- GPT is a model
- Chat-GPT is an application

- Hp→ chat gpt
- Gpt→ Intel
- Gpt→ Bard, Jasper

Present- GPT to ChatGPTs

- RLHF–
  - Supervised trainin on human conversation
  - Reinforcment
    - Ranking the answer of chatgpt

  - Apply language model and then Supervised fine tuning on conversation data → label data
  - Apply reinforcement Learning(Ranking the best answer)
- Incorporate safety and ethical guide line
  - Minimize biases
- Improvement in contextual points
  - Necessary for dialogue
- Training on conversational data
  - So know how to chat
- Continues improvement
  - Thumb signed
  - Two answer at the same time