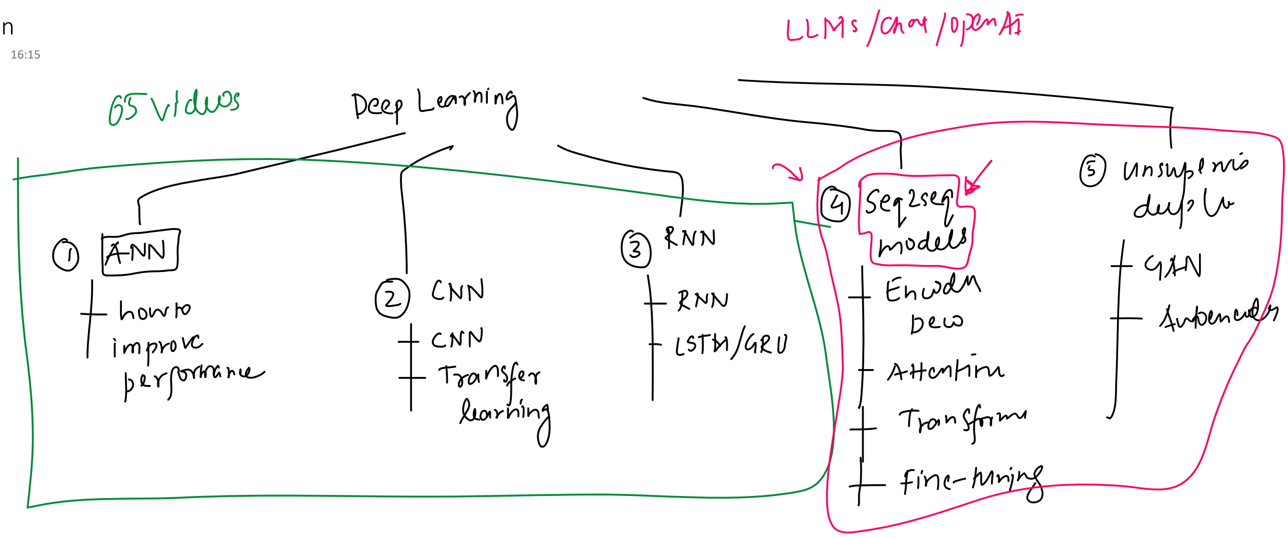
Introduction

16 November 2023 16:15

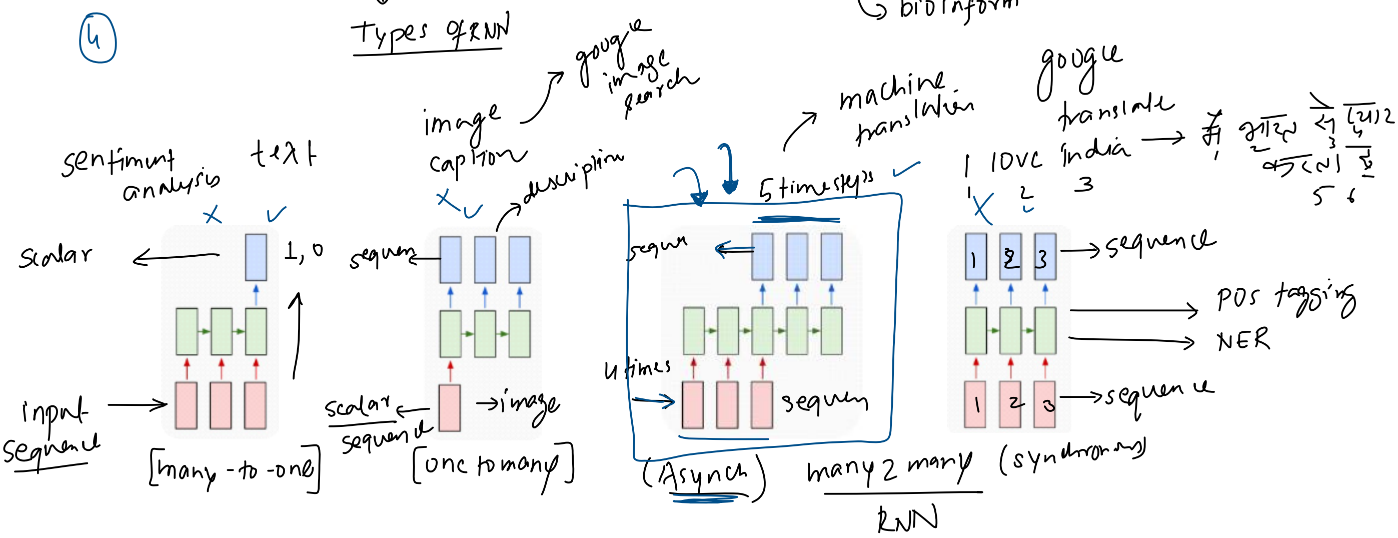


Sequence Tasks and its types

16 November 2023 16:16

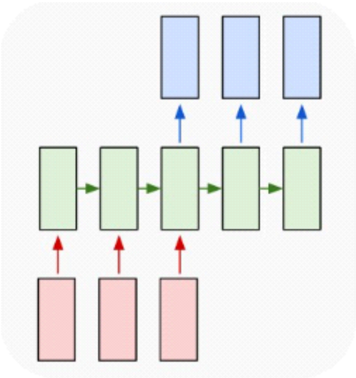
seq2seq → RNN → sequential → sequence → lang/sentence → time series → bioinform → seq2seq model

Types of RNN



Seq2Seq Tasks

16 November 2023 16:16



NLP

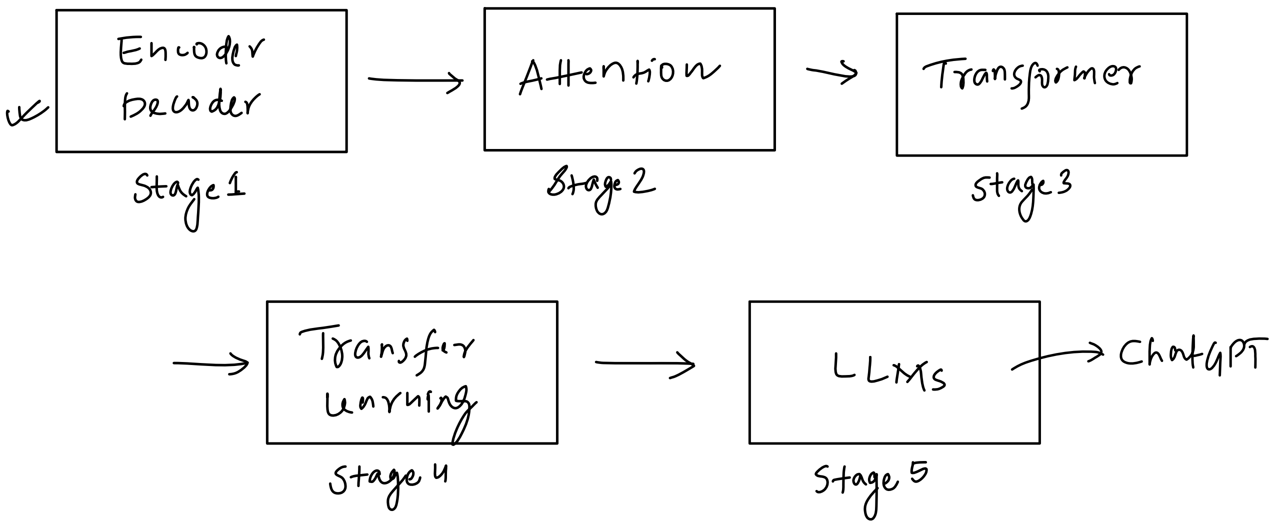
Seq2seq → machine trans

1) text summazi → seq → seq
2) Question answer → seq → seq
knowledge base
3) chatbot → input (text) → output (text)
4) speech-to-text → → seq2seq → chatgpt
seq seq

History of Seq2Seq Models

16 November 2023 16:16

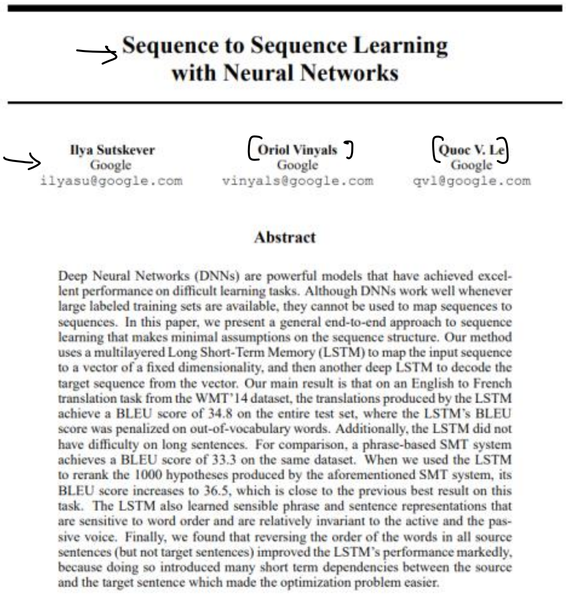
ChatGPT



Stage 1 - Encoder Decoder Architecture

18 November 2023 16:16

2014 Seminal



→ 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 score was penalized on out-of-vocabulary words. Additionally, the LSTM did not have difficulty on long sentences. For comparison, a phrase-based SMT system achieves a BLEU score of 33.3 on the same dataset. When we used the LSTM to rerank the 1000 hypotheses produced by the aforementioned SMT system, its BLEU score increases to 36.5, which is close to the previous best result on this task. The LSTM also learned sensible phrase and sentence representations that are sensitive to word order and are relatively invariant to the active and the passive voice. Finally, we found that reversing the order of the words in all source sentences (but not target sentences) improved the LSTM's performance markedly, because doing so introduced many short term dependencies between the source and the target sentence which made the optimization problem easier.

[Ilya Sutskever] → cofounder OpenAI

seq2seq
↓
diff
↳ encoder decoder

encoder/decoder 100 words

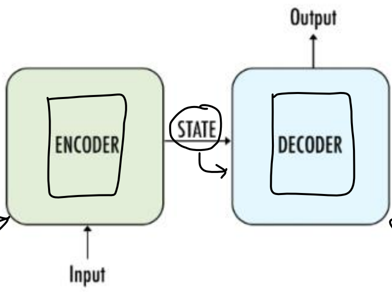
small sentence

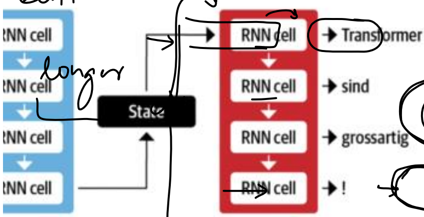
longer

→ RNN/LSTM/GRU

context vector

LSTM →





internal C_t / h_t update → summarize

compressed repr

seq2seq → input → out

machine trans

eng → hindi

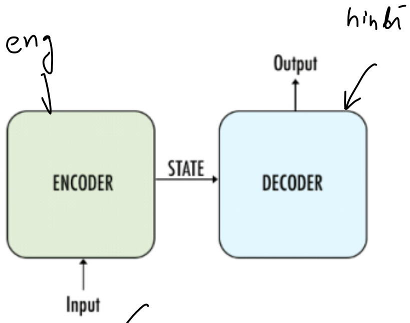
I love India → मुझे भारत से प्यार है

← compress ← process word by word

Attention mechanism

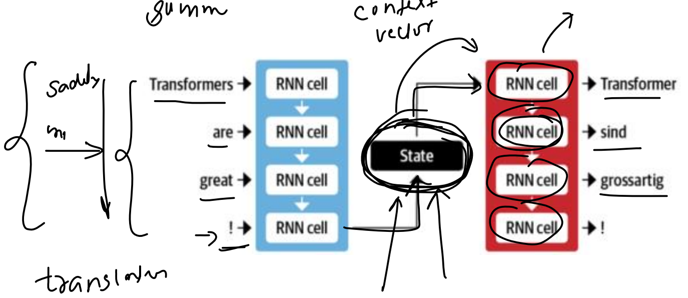
Stage 2 - Attention Mechanism

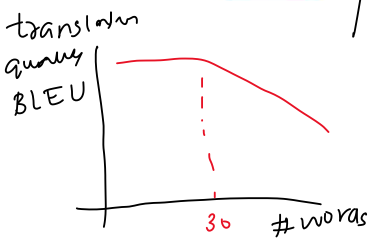
20 November 2023 10:59



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

Long input → 30 words





→ encoder decoder

2015

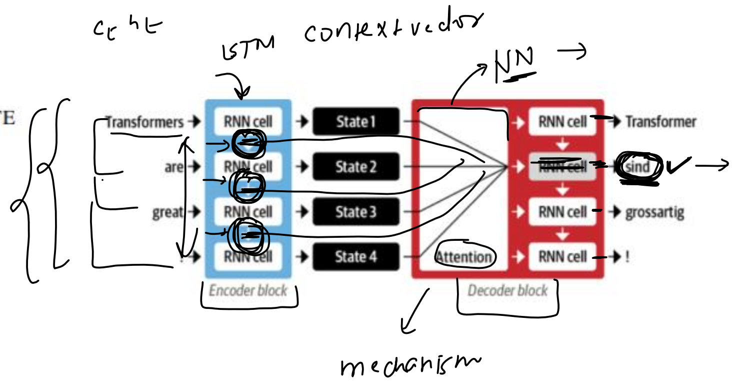
Attention

NEURAL MACHINE TRANSLATION BY JOINTLY LEARNING TO ALIGN AND TRANSLATE

Dzmitry Bahdanau
Jacobs University Bremen, Germany

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

ct ht



ABSTRACT

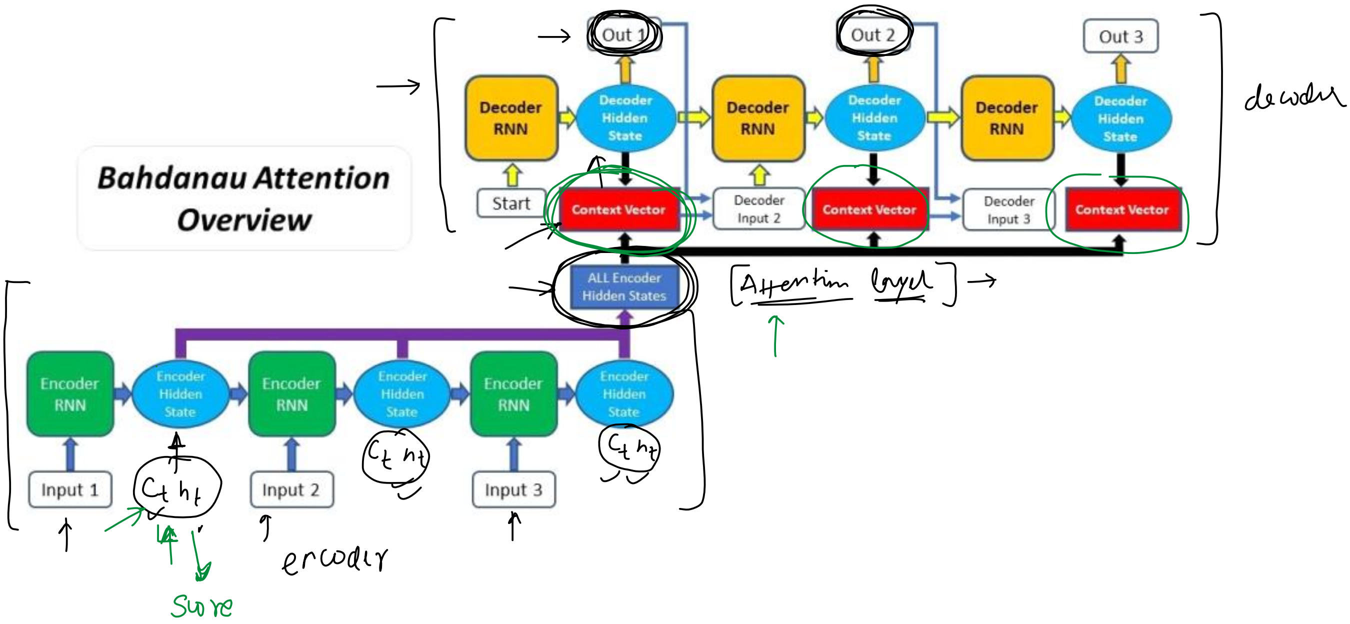
Neural machine translation is a recently proposed approach to machine translation. Unlike the traditional statistical machine translation, the neural machine translation aims at building a single neural network that can be jointly tuned to maximize the translation performance. The models proposed recently 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.

1 INTRODUCTION

Neural machine translation is a newly emerging approach to machine translation, recently proposed by Kalchbrenner and Blunsom (2013), Sutskever *et al.* (2014) and Cho *et al.* (2014b). Unlike the traditional phrase-based translation system (see, e.g., Koehn *et al.*, 2003) which consists of many small sub-components that are tuned separately, neural machine translation attempts to build and train a single, large neural network that reads a sentence and outputs a correct translation.

Most of the proposed neural machine translation models belong to a family of *encoder–decoders* (Sutskever *et al.*, 2014; Cho *et al.*, 2014a), with an encoder and a decoder for each language, or involve a language-specific encoder applied to each sentence whose outputs are then compared (Hermann and Blunsom, 2014). An encoder neural network reads and encodes a source sentence into a fixed-length vector. A decoder then outputs a translation from the encoded vector. The whole encoder–decoder system, which consists of the encoder and the decoder for a language pair, is jointly trained to maximize the probability of a correct translation given a source sentence.

A potential issue with this encoder–decoder approach is that a neural network needs to be able to compress all the necessary information of a source sentence into a fixed-length vector. This may make it difficult for the neural network to cope with long sentences, especially those that are longer than the sentences in the training corpus. Cho *et al.* (2014b) showed that indeed the performance of a basic encoder–decoder deteriorates rapidly as the length of an input sentence increases.


Bahdanau Attention Overview
Out 1
Out 2
Out 3
Decoder RNN
Decoder Hidden State
Start
Context Vector
Decoder Input 2
Decoder Input 3
ALL Encoder Hidden States
Encoder RNN
Encoder Hidden State
Input 1
Input 2
Input 3
decoder
[Attention layer]
encoder
$C_t\ h_t$
score

Stage 3 - Transformers

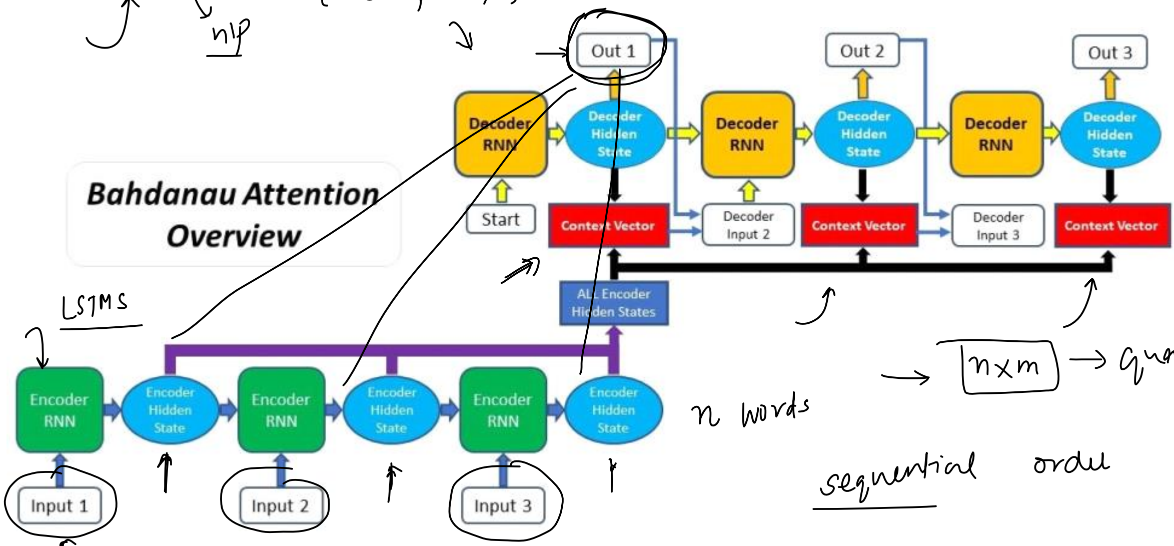
20 November 2023 12:18

nlp

{ computation complexity }

m words

2015-2017



atten

encoder decoder

parallel processing

LSTMs

n words

$\boxed{n \times m}$ → quadratic

sequential order

2017

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-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

LSTM ×

LSTM

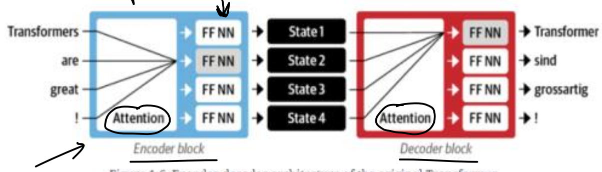


Figure 1-6. Encoder-decoder architecture of the original Transformer

LSTM / RNN cell

Attention

Self-attention

stage

arch

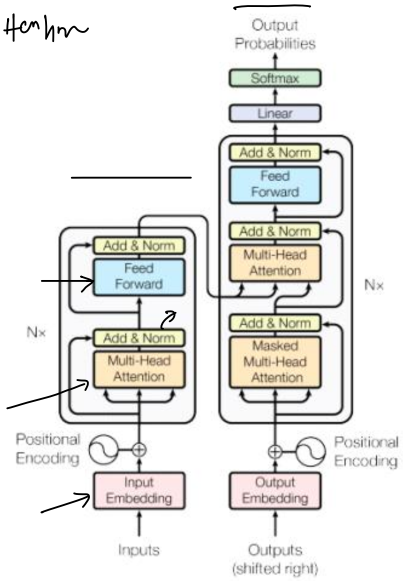


Figure 1: The Transformer - model architecture.

Stage 4 - Transfer Learning

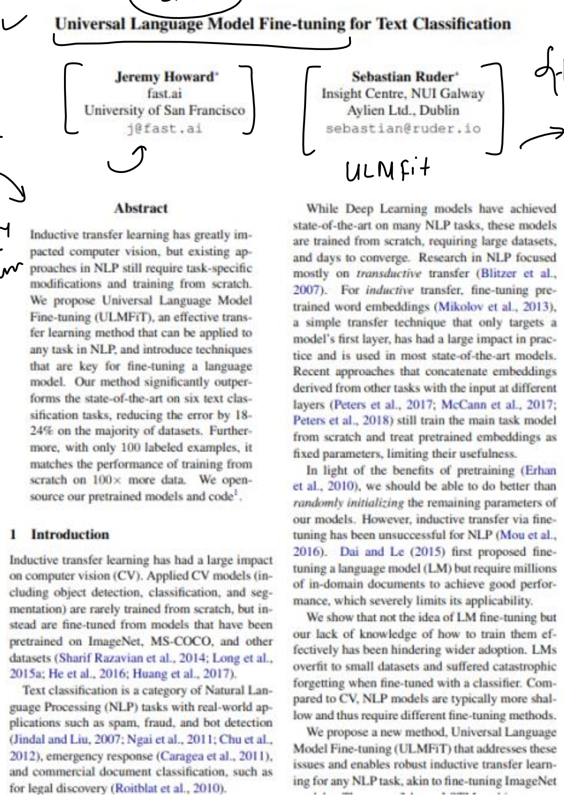
20 November 2023 15:39

1) Hardware → GPUs
2) Time →
3) Data → a lot

→ 100 , 1000

transformer → Transfer Learning

ULMFiT



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-source our pretrained models and code[1].

1 Introduction

Inductive transfer learning has had a large impact on computer vision (CV). Applied CV models (including object detection, classification, and segmentation) are rarely trained from scratch, but instead are fine-tuned from models that have been pretrained on ImageNet, MS-COCO, and other datasets (Sharif Razavian et al., 2014; Long et al., 2015a; He et al., 2016; Huang et al., 2017).

Text classification is a category of Natural Language Processing (NLP) tasks with real-world applications such as spam, fraud, and bot detection (Jindal and Liu, 2007; Ngai et al., 2011; Chu et al., 2012), emergency response (Caragea et al., 2011), and commercial document classification, such as for legal discovery (Roitblat et al., 2010).

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 pretrained 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 *randomly initializing* the remaining parameters of our models. However, inductive transfer via fine-tuning has been unsuccessful for NLP (Mou et al., 2016). Dai and Le (2015) first proposed fine-tuning a language model (LM) but require millions of in-domain documents to achieve good performance, which severely limits its applicability.

We show that not the idea of LM fine-tuning but our lack of knowledge of how to train them effectively has been hindering wider adoption. LMs overfit to small datasets and suffered catastrophic forgetting when fine-tuned with a classifier. Compared to CV, NLP models are typically more shallow and thus require different fine-tuning methods.

We propose a new method, Universal Language Model Fine-tuning (ULMFiT) that addresses these issues and enables robust inductive transfer learning for any NLP task, akin to fine-tuning ImageNet

primary pretrain

ULMFit

2018

transfer → vision

nlp tasks

NLP

useful

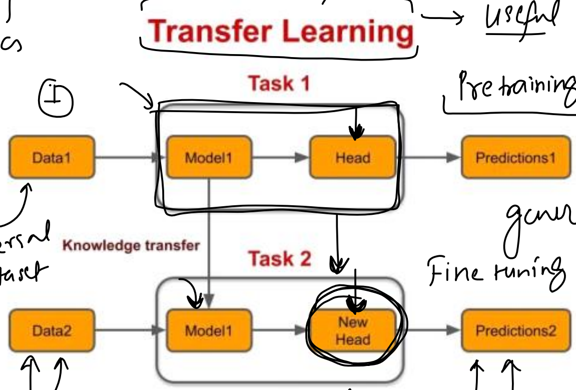
Pretraining

I

universal dataset

general features

Fine tuning



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.

imagenet → dataset → Millions of

↳ CNN → ResNet

↳ inception

↳ pretrain

edges

↳ shapes

↓

fine-tune → cat vs dog classify → 100

2018

1) Task specificity → NLP

Senti

NER

POS

ML

QA

summa

language

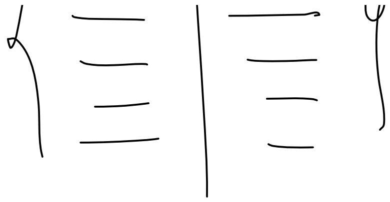
NLP → transfer learning

ML

2nd) Lack of data

eng | hin

NLP → transfer learning



Pretraining → Machine transl X

→ language modelling → pretraining → successful → (2)

NLP task → NLP/DL model   next word predi

I live in india. and the capital of ___ is New Delhi

Language modelling as a Pretraining task

→ unsupervised pretrain task

1) Rich feature learning

The hotel was exceptionally clean, yet the service was bad / pathetic

→ know trans

text classi / quest. | text summ | NER / POS tag

mt (lang → supervised labeled

eng | hin

→ unsupervised task

2) Huge avail of data

pdf → dataset labelling

fine tuning

[ULMFiT]

X transformer

AWD LSTM → wikipedia → classifier → imdb / yelp / new data → model → test → State of the art

unsupervised pretrain language modelling

Scratch → 10000 rows

100 row → better →

2012   2018

2012

2018

parallely

$\boxed{2018}$

archi

[transformer]

train

[transfer
leq] ?

Stage 5 - LLMs

20 November 2023 18:51

2018 → 10 ULMFit Jan Oct

[language]

Transformer

Google

BERT [Encoder only]

Open AI [decoder only model]

→ GPT

transformer

dataset → huge

↳ Transfer learning → txt summ, ques, sentim, ner, pos

download

limited dataset

finetune

SOTA

GPT → GPT2 → GPT3 → language model → LLM → LLMS

↳ parameter

↳ dataset

[LM → LLMs]

Large scale $\boxed{\text{LLMs}}$

Qualities of LLMs

1) Data → billions → GPT3 → 45 TBs

↳ book, websites, internet

↳ diversity → bias

2) Hardware → Cluster of GPU → GPT3 → Supercomputer → 1000s NVIDIA GPU

3) Training → days to weeks

4) Cost → hardware + elec + infra + experts → individual, companies, govt, institutes

↳ millions

4) energy consump → (gpt3) →

↳ energy consump

↳ gpt3 → small town

↳ month

The Grand Finale - ChatGPT

22 November 2023 10:13

GPT3 → chatGPT
↓
[diff]

GPT → model
[ChatGPT] → application
Chatbot

Nov

hp → chatgpt
intel → GPT → Bard, Jasper
→ dell
→ ASUS

GPT3 → [ChatGPT]

1) RLHF → Reinfo lear from human feed
   - 1 Supervised fine tuning → dataset } labelled
   - 2 Reinforce → prompt produce
     - responses
     - human → response rank

2) Incorporate safety and ethical guideline
   - minize biases

3) improvent in contextual point
   context retain → maintain context } dialogue Conver

4) Dialogue specific training
   - conversation
   - better under → dialogue lang → patterns

5) ChatGpt continous imp → human feedback
   user

train → refining

[GPT4] → [GPT3]