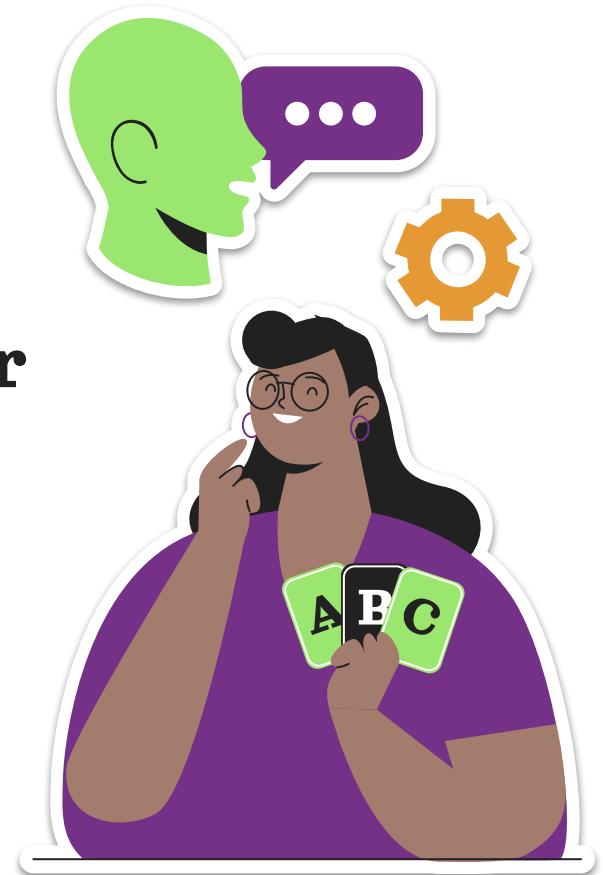


# **Investigating Efficiently Extending Transformers for Long Input Summarization**



Introduction to NLP Seminar



# Our team

Trần Anh Túc

19127651



Lê Tấn Đạt

19127353



# Table of contents



01

## Overview

Overview

02

## Introduction

Introduction to  
text summarisation

03

## Related work

Overview of long text  
summarisation



# Table of contents



**04**

## Architecture

Proposed model architecture

**05**

## Experiment and Conclusion

Experiment result and conclusion

**06**

## Reference

Reference used on seminar





# Overview

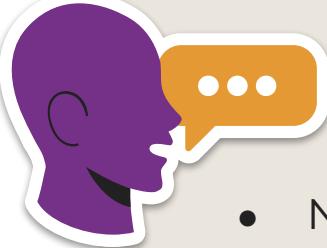
Tổng quan



## Natural Language Processing

Xử lý ngôn ngữ tự nhiên là một nhánh nghiên cứu của trí tuệ nhân tạo, được phát triển nhằm xây dựng các chương trình máy tính có khả năng phân tích, xử lý, và hiểu ngôn ngữ con người, dưới dạng tiếng nói (speech) hoặc văn bản (text).

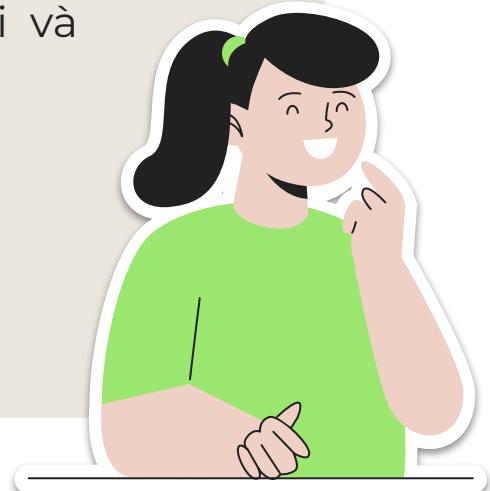




## Applications of NLP



- Nhận dạng giọng nói tự động
- Chuyển văn bản thành giọng nói và ngược lại
- Truy xuất, trích xuất thông tin
- Dịch máy
- Chatbot
- **Tóm tắt văn bản tự động**





A-Z

02

# Introduction

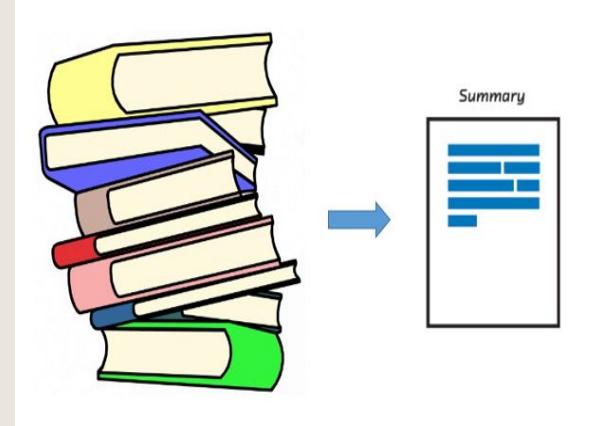
Giới thiệu bài toán





# What is text summarization?

- Tóm tắt văn bản là quá trình trích xuất những thông tin quan trọng nhất từ một hoặc nhiều nguồn văn bản để tạo ra một phiên bản cô đọng, súc tích để thực hiện một nhiệm vụ cụ thể nào đó
- Có thể thực hiện thủ công hoặc tự động.





# Advantages

- Tiết kiệm thời gian
- Thực hiện ngay lập tức
- Xử lý được đa ngôn ngữ

# Disadvantages

- Sự nhầm lẫn về từ, câu đặc biệt
- Thiếu những thông tin liên quan
- Không chỉ rõ đâu mới là từ quan trọng



## Application of Text Summarization

- Tóm tắt bài báo, tin tức
- Tóm tắt nội dung hội nghị
- Trả lời tự động (Chatbot)
- Tóm tắt nội dung âm thanh, phim ảnh
- Hỗ trợ bác sĩ khám/chữa bệnh
- Tự động tạo script cho video



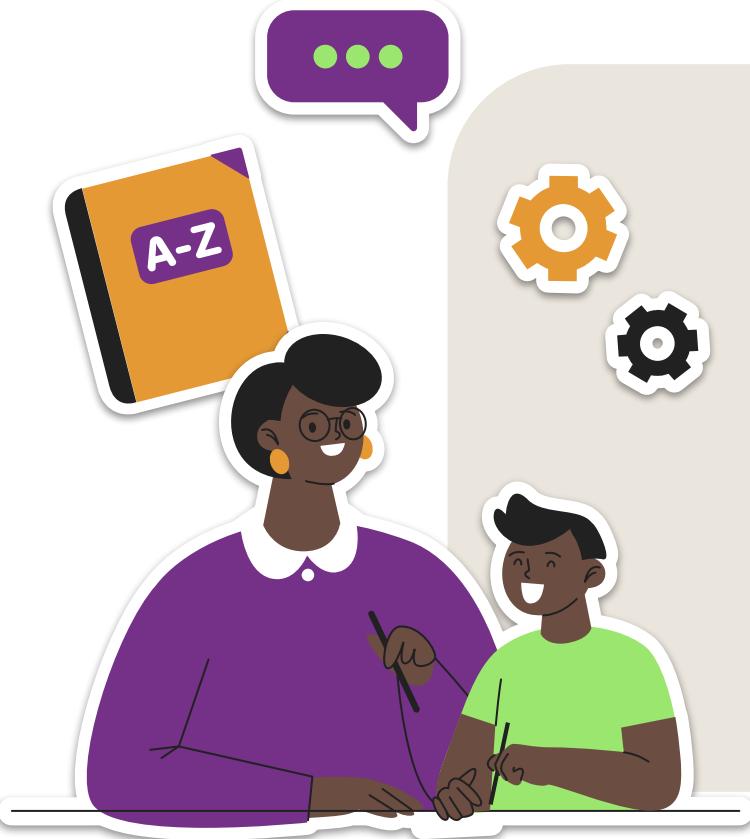


# Text Summarization

- Hiện nay, nhiều thuật toán tóm tắt văn bản đã được các công ty và các nhà nghiên cứu phát triển.
- Vào năm 2017, Google đã đăng tải bài báo "Attention Is All You Need" thông tin về Transformer - một kiến trúc không sử dụng kiến trúc hồi quy mà chỉ sử dụng một kỹ thuật gọi là **attention** để xử lý toán tóm tắt, dịch thuật văn bản.

# Backdraws

- Xử lý chuỗi đầu vào dài là một thách thức đáng kể.
- Các mô hình đào tạo để xử lý các chuỗi dài tốn kém cả về tính toán và bộ nhớ, đồng thời yêu cầu đào tạo và đánh giá trên dữ liệu chuỗi dài, việc thu thập dữ liệu khó khăn và yêu cầu lượng lớn data.





03



## Let's dive in

Cơ sở lý thuyết và các công trình nghiên cứu liên quan

# Text summarization



## Extraction based

Trích chọn các từ/câu chính rồi gộp lại với nhau thành đoạn tóm tắt. Chỉ chọn câu có độ quan trọng cao



## Abstraction based

Xem xét toàn bộ văn bản và tạo summary dựa trên ý chính của văn bản đó

Diễn đạt lại hoặc tái cấu trúc câu. Câu trong đoạn tóm tắt không xuất hiện trong văn bản



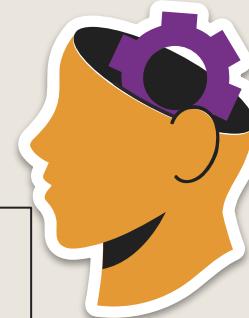
# Category



Purpose

Generic   Domain specific   Query-based

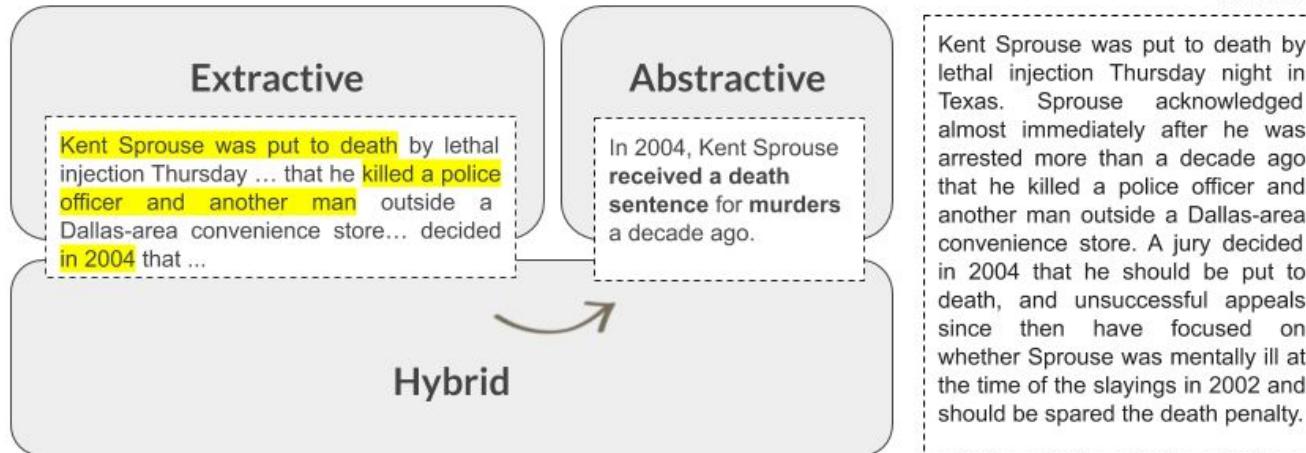
Dimensions



Input type

Single

Multi



Mô hình hybrid kết hợp tốc độ và độ chính xác cao của phương pháp rút trích và sự trôi chảy/ tổng quan cao của phương pháp trừu tượng

Nguồn: [Long document text summarization](#)

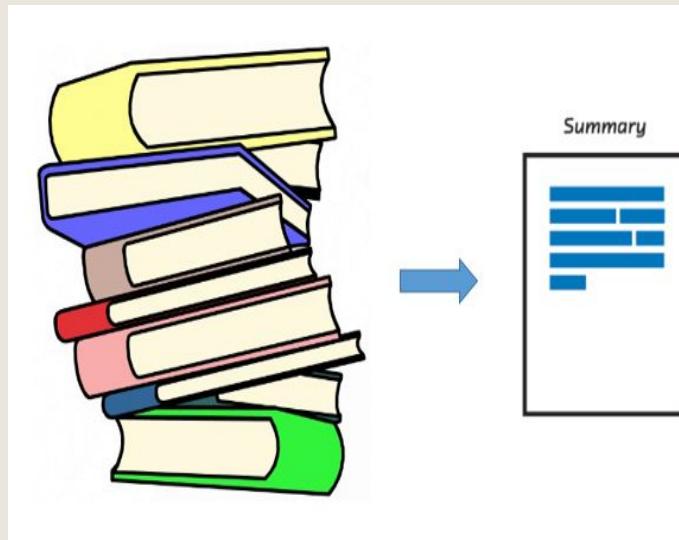
Phương pháp rút trích dễ dàng và hiệu quả cao, tuy nhiên phương pháp trừu tượng cung cấp một giải pháp tóm tắt văn bản tổng quan và mạch lạc hơn.



# Deep learning in Text Summarisation

Các hướng tiếp cận trong Tóm tắt văn bản:

- Rút trích (Extractive summarization)
- Topic representation approach:  
Frequency-driven approach
- Knowledge-based and automatic summarisation
- Context summarisation
- Indicator representation approach:  
Graph Methods for Summarization,  
Machine Learning for  
Summarization





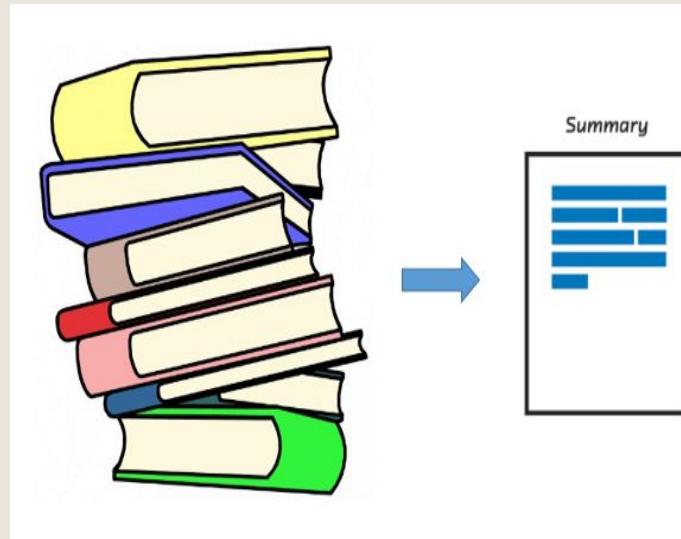
Nội dung tìm hiểu của nhóm  
về phương hướng tiếp cận  
dùng Deep learning cho bài  
toán tóm tắt văn bản (trường  
hợp đặc biệt “long input” )



# Deep learning in Text Summarisation

Tóm tắt văn bản dùng học sâu:

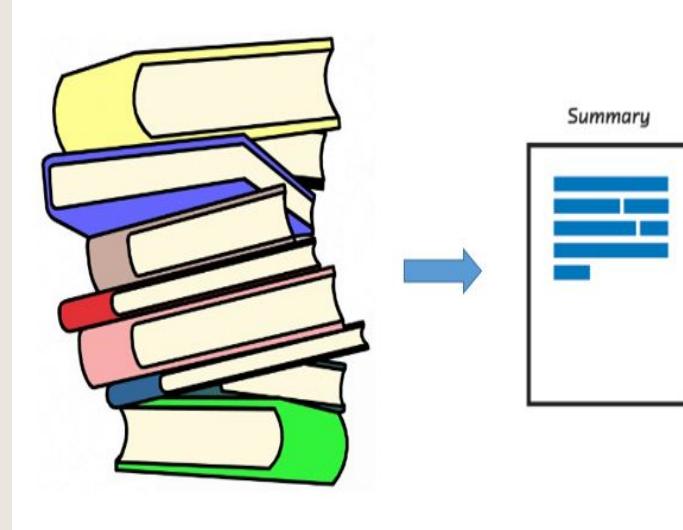
- Các mô hình deep learning có thể xem là một phương pháp trừu tượng hóa, văn bản tóm tắt được sinh ra từ một language generation model từ input văn bản gốc.
- Các mô hình học sâu mặc dù chưa tối ưu bằng phương pháp trích xuất nhưng đem lại cạnh tranh cho các giải pháp trừu tượng hóa.



# Deep learning in Text Summarisation

Tóm tắt văn bản dùng học sâu:

- Ưu điểm của các mô hình học sâu là nó hoàn toàn có thể train end-to-end mà không cần bước chuẩn bị dữ liệu đặc biệt hay mô hình con. Mô hình sẽ dựa hoàn toàn vào văn bản mà không cần chuẩn bị dữ liệu chuyên ngành hay tiền xử lý trên dữ liệu gốc.





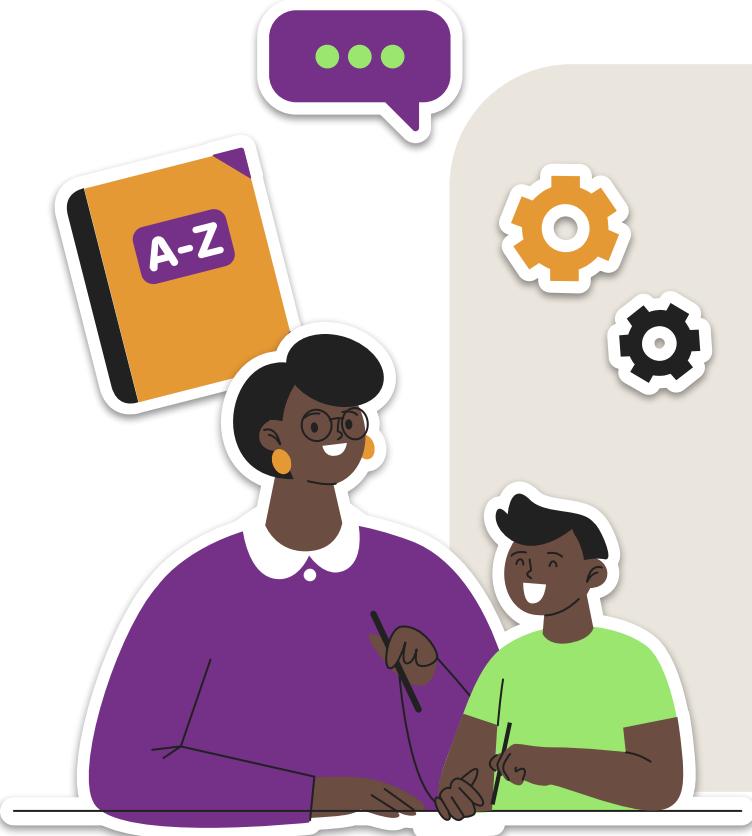
# “Long Document”

Một document được coi là dài (“long document”) khi các hệ thống SOTA xử lý các document ngắn (“short document”) không thể mở rộng để xử lý các document dài hơn rất nhiều.

Long document (e.g., scientific papers, theses, and novels).

# “Long Document”

Mặc dù định nghĩa này có thể nồng gâý ra vài nhầm lẫn, tuy nhiên nó đảm bảo được sự thúc đẩy phát triển các kiến trúc mô hình vượt qua các hạn chế của phần cứng hơn là các giải pháp “bản sao” của các nghiên cứu trước đó.



[1] An Empirical Survey on Long Document Summarization:  
Datasets, Models and Metrics

## Short document

vs.

## Long document

CNN/Daily Mail
- Avg. doc len: 656 (CNN), 693 (Daily Mail)
- Avg. summary len: 42 (CNN), 53 (Daily Mail)
NY Times
- Avg. doc len: 530
- Avg. summary len: 38

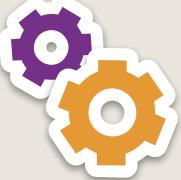
arXiv/PubMed
- Avg. doc len: 4938 (arXiv), 3016 (PubMed)
- Avg. summary len: 220 (arXiv), 203 (Daily Mail)
BIGPATENT
- Avg. doc len: 3573
- Avg. summary len: 117

Comparison of datasets for short document summarization (left)  
and long document summarization (right)

	Short Document Datasets					Long Document Datasets					Long vs. Short
	CNN-DM	NWS	XSum	WikiHow	Reddit	ArXiv	PubMed	BigPatent	BillSum	GovReport	Avg. Ratio
# doc-summ.	278K	955K	203K	231K	120K	215K	133K	1.34M	21.3K	19.5K	-
summ tokens	55	31	24	70	23	242	208	117	243	607	6.9x
doc tokens	774	767	438	501	444	6446	3143	3573	1686	9409	8.3x
summ sents	3.8	1.5	1	5.3	1.4	6.3	7.1	3.6	7.1	21.4	3.7x
doc sents	29	31	19	27	22	251	102	143	42	300	6.5x
Compression <sub>token</sub>	14.8	31.7	19.7	7.2	18.4	41.2	16.6	36.3	12.2	18.7	1.4x
Compression <sub>sent</sub>	8.3	22.4	18.9	3.3	14.5	44.3	15.6	58.7	9.7	18.1	2.2x
Coverage	0.890	0.855	0.675	0.610	0.728	0.920	0.893	0.861	0.913	0.942	1.2x
Density	3.6	9.8	1.1	1.1	1.4	3.7	5.6	2.1	6.6	7.7	1.5x
Redundancy	0.157	0.088	-	0.324	0.078	0.144	0.146	0.223	0.163	0.124	1.0x
Uniformity	0.856	0.781	0.841	0.813	0.777	0.894	0.896	0.922	0.903	0.932	1.2x

Table 1. Comparison of Short and Long Document Summarization Datasets. Intrinsic characteristics are computed based on the average result of test samples. Average Ratios are computed based on the average long over short document statistics.

# Disadvantages of long document

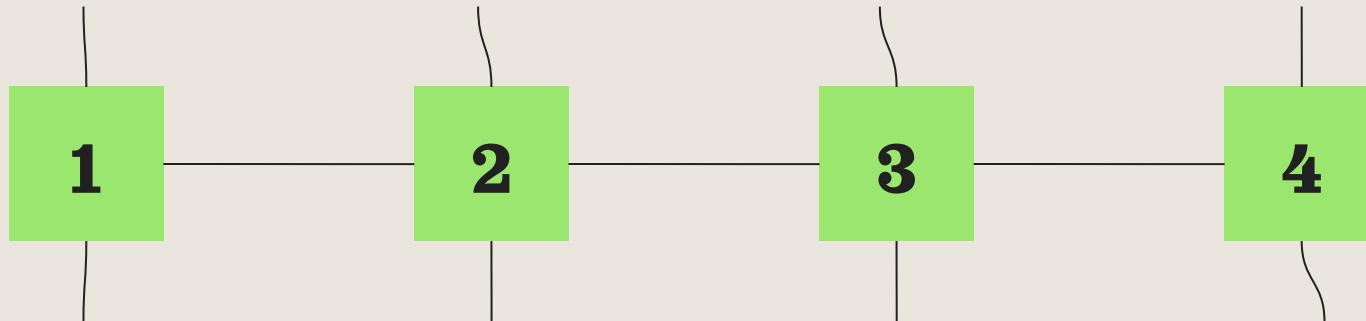


- Nhiều từ nhiều hơn với khả năng cao xuất hiện nhiều lần
- Những nội dung quan trọng phân bố rải rác
- Cần nhiều tài nguyên phần cứng (lưu trữ các vector embedding dài..)

# Methods for long document input



**Truncating**      **Focus on**      **Hybrid**      **Divide-and-c**  
**input**      **informative parts**      **models**      **onquer**



Cắt ngắn tài liệu input đến độ dài chấp nhận được

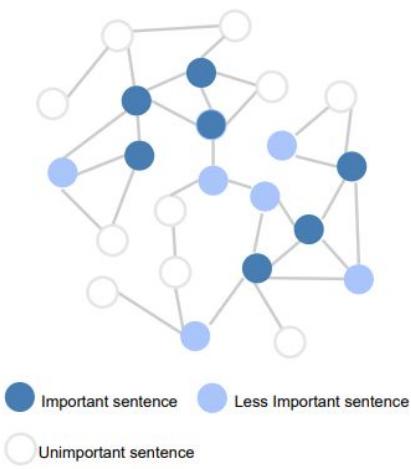
Thực hiện tóm tắt trên những phần nhỏ hơn được cho là quan trọng

Rút trích các phần quan trọng và thực hiện tóm tắt trên những phần đó, giảm không gian làm việc

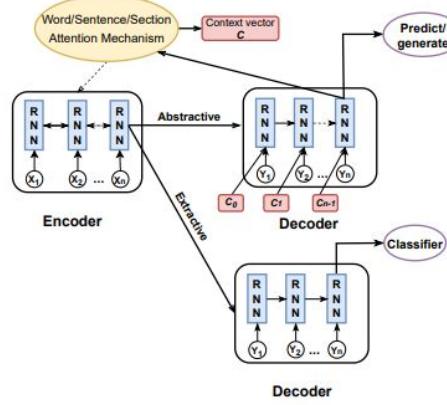
Tóm tắt các đoạn và tóm tắt văn bản từ các đoạn tóm tắt đó.



A - Classic Graph (Extractive)



B - RNN-based Model Architecture



C - Transformers

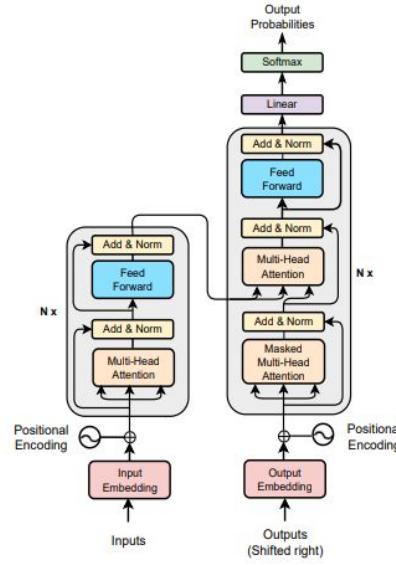


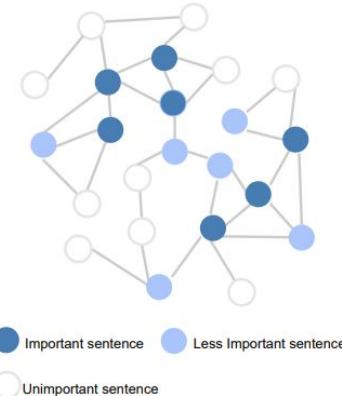
Fig. 2. Overview of Model Architectures.

Paper: An Empirical Survey on Long Document Summarization: Datasets, Models and Metrics

# Graph Architecture

Bao gồm hai giai đoạn: ánh xạ tài liệu vào trong không gian của mạng đồ thị (graph network), các nút đại diện cho các câu còn các cạnh thể hiện mối quan hệ tương đồng giữa hai nút trong mạnh. Rút trích top-K các câu có thứ hạng cao dự trên độ đo khoảng cách của nó đến trung tâm của đồ thị.

A - Classic Graph (Extractive)



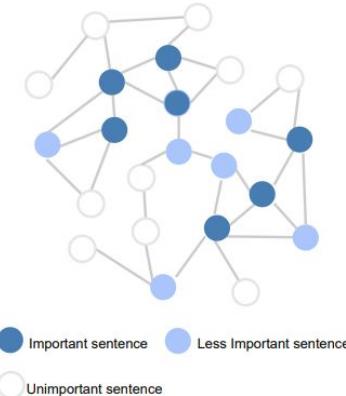


# Graph Architecture

Đạt được kết quả SOTA trong các phương pháp tóm tắt bằng trích xuất không giám sát tài liệu dài khi tích hợp với các mô hình SOTA hiện tại.



A - Classic Graph (Extractive)



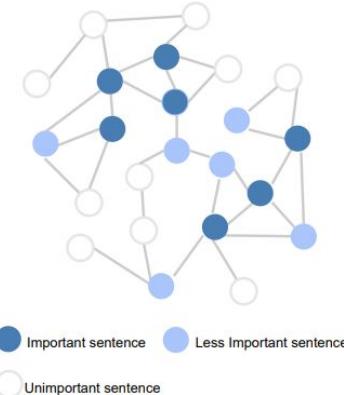


# Graph Architecture

Other than the multi-sentence compression approach that may be extended to long document summarization tasks, there has been no applicable work on classical graph-based architecture for long document abstractive summarization.



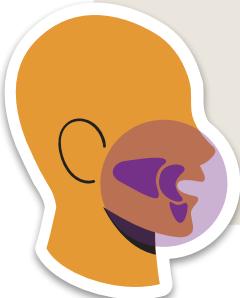
A - Classic Graph (Extractive)



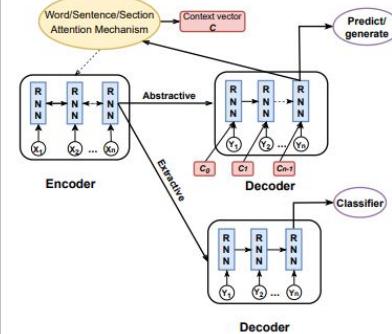


# Recurrent Neural Networks

RNN không có khả năng nắm bắt các phụ thuộc thời gian trong phạm vi dài trên một văn bản đầu vào dài (long document).



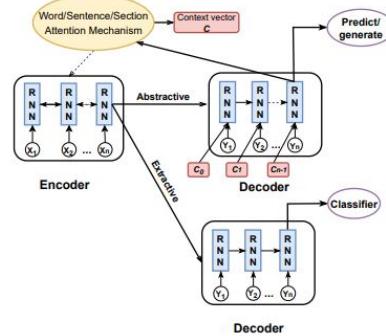
B - RNN-based Model Architecture



# Recurrent Neural Networks

Mãi cho đến LSTM-minus (một biến thể của RNN), Cohan et al. đã trình bày kiến trúc bộ encoder-decoderLSTM trong đó decoder tham gia vào từng phần(section) của tài liệu nguồn để xác định trọng số chú ý ở cấp độ phần trước khi tham gia vào từng từ, mới có thể đưa các giải pháp sử dụng kiến trúc RNN đi vào xử lý bài toán tóm tắt văn bản dài.

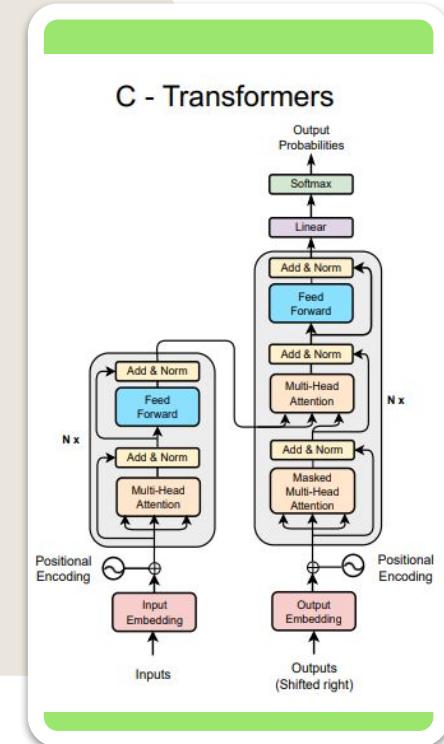
B - RNN-based Model Architecture





# Transformers

Các mô hình Transformer đã đạt được nhiều thành tựu cao trong các tác vụ xử lý ngôn ngữ seq2seq. Tuy nhiên, quy mô bậc hai của các yêu cầu bộ nhớ và tính toán cho cơ chế Attention trong Transformers đặt ra một thách thức để giải quyết các nhiệm vụ tóm tắt long document input.



	<b>Model</b>	<b>Architecture</b>	<b>Pre-Train</b>	<b>Long Document Mechanism</b>	<b>Max Token</b>
<i>Unsupervised Extractive</i>	PacSum [135]	Graph	BERT	Discourse Bias	-
	HipoRank [25]	Graph	BERT	Discourse Bias	-
	FAR [69]	Graph	BERT	Facet-Aware Scoring	-
	IBSumm [55]	Pipeline	SciBERT	Signal Guidance	-
<i>Supervised Extractive</i>	GlobalLocal [118]	RNN	-	Discourse Bias	-
	Sent-CLF/PTR [94]	RNN	-	-	-
	Topic-GraphSum [22]	GAT	BERT	Neural Topic Modelling	-
	SSM-DM [21]	DMN	BERT	-	-
<i>Supervised Abstractive</i>	Discourse-Aware [20]	RNN	-	Discourse Bias	-
	Longformer [2]	Transformer	BART	Efficient Attention	16,384
	BigBird [127]	Transformer	PEGASUS	Efficient Attention	4,096
	GSUM [26]	Transformer	BART	Signal Guidance	4,096
	CRTLSum [45]	Transformer	BART	Prompt Engineering	1,024
	HAT-BART [99]	Transformer	BART	Hierarchical Attention	1,024
	HEPOS [49]	Transformer	BART	Efficient Attention	10,240
<i>Supervised Hybrid</i>	TLM+Ext [94]	Transformer	-	Content Selection + Discourse Bias	-
	DANCER [37]	Transformer	PEGASUS	Content Selection + Discourse Bias	-
	SEAL [134]	Transformer	-	Content Selection w/ Segment-wise Scorer	-
	LoBART [77]	Transformer	BART	Content Selection + Efficient Attention	-

Table 2. Long Document Summarization Models in Chronological Order. Max token represents the maximum input sequence length that the model can process and any text that exceeds this cutoff point will be truncated.

# What you will learn



- Trong khi các pretrained Transformer cho thấy khả năng cao trong xử lý các tác vụ xử lý ngôn ngữ tự nhiên, thì việc xử lý các kiểu văn bản dài vẫn là một trong những khó khăn cần giải quyết (do hạn chế phần cứng chi phí,...).
- Tóm tắt văn bản dài là một trong những tác vụ có đầu vào là các chuỗi dài hơn độ dài đầu vào tối đa của hầu hết các mô hình pretrained hiện tại.
- Tuy nhiên, qua một số thử nghiệm như trong bài báo: Với một số điều chỉnh và thay đổi trong kiến trúc, các pretrained Transformer vẫn có thể đáp ứng được nhiệm vụ tóm tắt văn bản cho “long input document”.



A-Z

04

# Architecture

Sơ lược kiến trúc của bài báo



# Paper propose



- Thực nghiệm trên một số Transformer architectures trên long input text summarisation
- Trình bày cách thức để áp dụng model của short document cho long document, pre-trained Transformer model cho long-context, và đưa ra PEGASUS, kiến trúc cải thiện hiệu quả text summarisation cho longer input

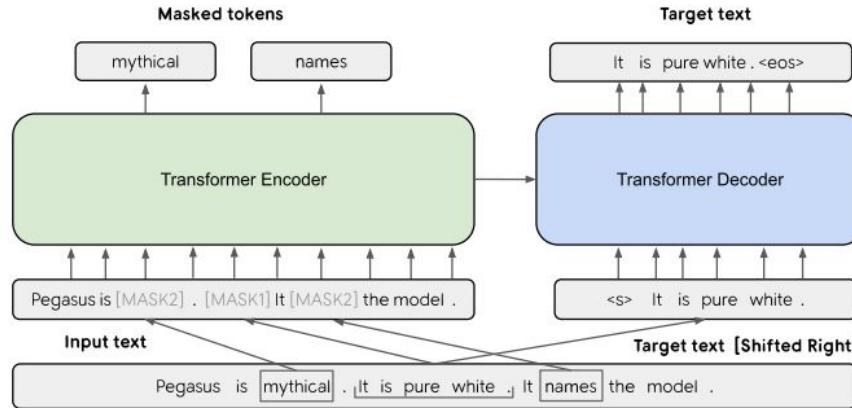
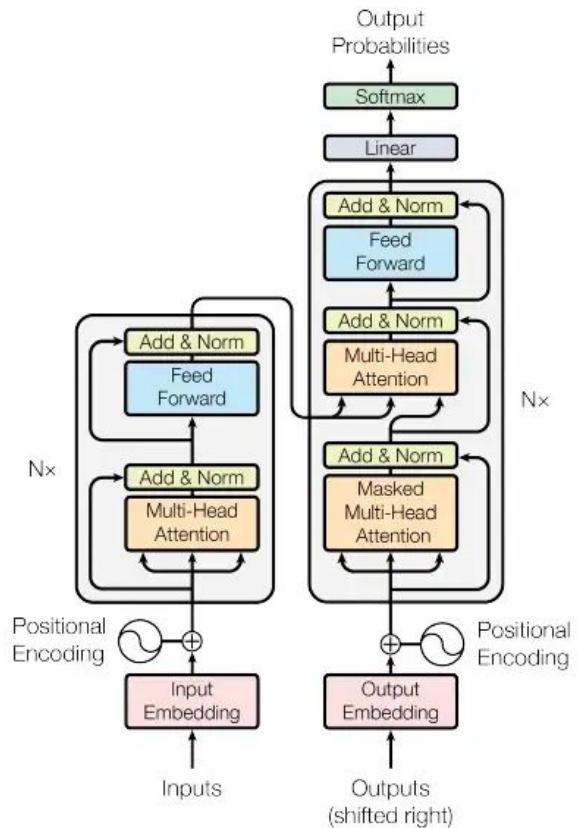


Figure 1: The base architecture of PEGASUS is a standard Transformer encoder-decoder. Both GSG and MLM are applied simultaneously to this example as pre-training objectives. Originally there are three sentences. One sentence is masked with `[MASK1]` and used as target generation text (GSG). The other two sentences remain in the input, but some tokens are randomly masked by `[MASK2]` (MLM).

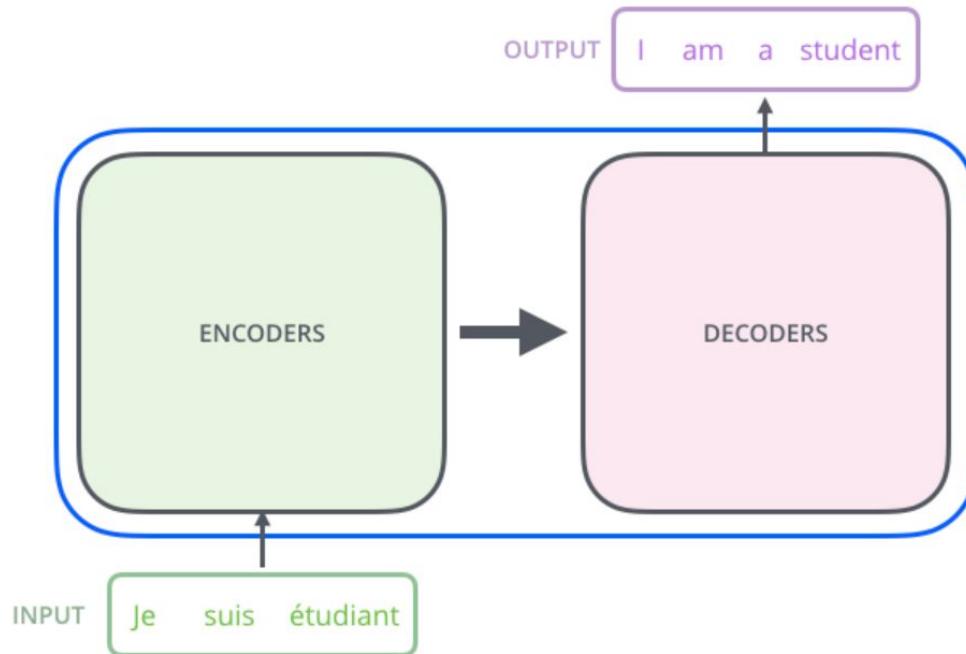
# Transformer architecture



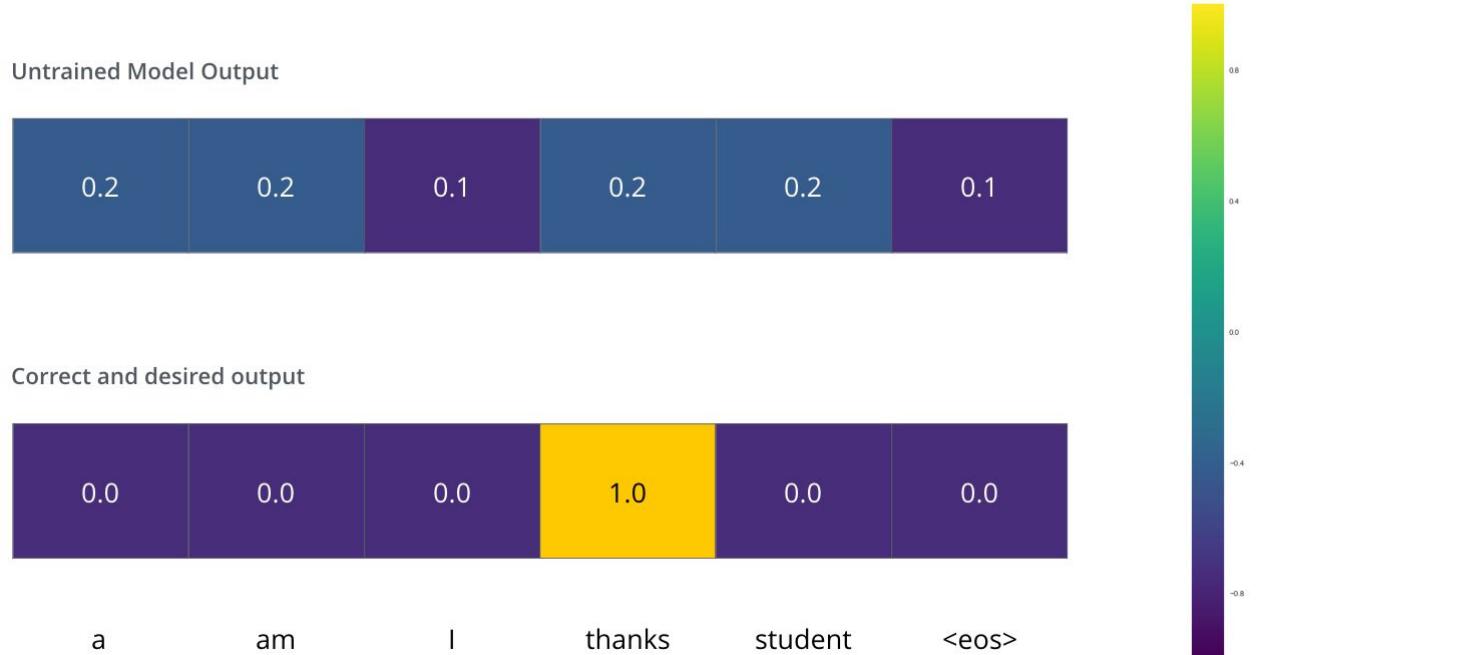
Pretrained PEGASUS được xây dựng trên kiến trúc Transformer với encoder trả về các mask tokens và decoder trả về các gap sentences tạo nên tóm tắt cho văn bản đầu vào.



Paper: Attention is all you need  
[illustrated-transformer](#)



Paper: Attention is all you need  
[illustrated-transformer](#)



Cross Entropy loss

Paper: Attention is all you need  
[illustrated-transformer](#)

# Pretrain Pegasus



- Pegasus là mô hình Transformer encoder-decoder cho tác vụ abstractive summarisation sequence-to-sequence.
- Pegasus là giải pháp SOTA trong tóm tắt văn bản trên 12 tập dữ liệu.
- Data training:
  - Mask-1: che các câu được chọn
  - Mask-2: che các tokens được chọn
- Mô hình được pretrain với data được che Mask-1 (GSG) và sau đó train model encoder trên ngữ liệu được che Mask-2 (MLM)

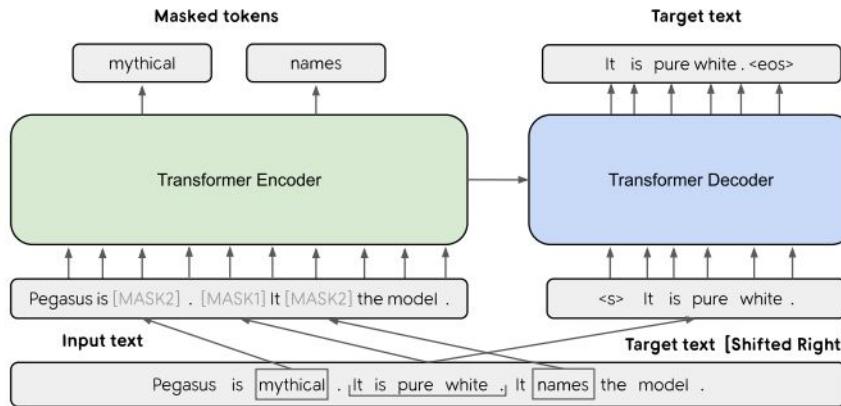


Figure 1: The base architecture of PEGASUS is a standard Transformer encoder-decoder. Both GSG and MLM are applied simultaneously to this example as pre-training objectives. Originally there are three sentences. One sentence is masked with [MASK1] and used as target generation text (GSG). The other two sentences remain in the input, but some tokens are randomly masked by [MASK2] (MLM).

Paper: [PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization](#)

<b>GovReport</b>	<i>Summarization</i>
<b>Input</b>	<p>Introduction</p> <p>The United States has an abundance of natural resources. For much of the nation's history, energy availability was not a concern as commerce and industry needs could be met by domestic supplies. However, industrialization and population growth, and the continuing development of a consumer-oriented society, led to growing dependence...</p>
<b>Output</b>	<p>Energy is crucial to the operation of a modern industrial and services economy. Concerns about the availability and cost of energy and about environmental impacts of fossil energy use have led to the establishment of...</p>
<b>SummScreenFD</b>	<i>Summarization</i>
<b>Input</b>	<p>Ted's kitchen</p> <p>Ted from 2030: Kids, when it comes to love, the best relationships are the ones that just come naturally.</p> <p>Ted: My first solo batch.</p> <p>Victoria: Um, I think those need to stay in the oven a while longer. Here's a professional tip. If it's still runny, it's not a cupcake. It's a beverage...</p>
<b>Output</b>	<p>Just as things are going well between Ted and Victoria, the latter is offered a surprising but incredible opportunity to be a fellow at a culinary institute in Germany. As the couple discuss the viability of long-distance...</p>
<b>QMSum</b>	<i>Query-Based Summarization</i>
<b>Input</b>	<p>What did the team discuss during the product evaluation about its feature to solve customers' concerns?</p> <p>Project Manager: Yep. Soon as I get this. Okay. This is our last meeting. Um I'll go ahead...</p>
<b>Output</b>	<p>Generally speaking, the team agreed that the product was intuitive and had successfully incorporated main aims that the team had. The team believed the customers were not likely to lose the remote control since it was...</p>
<b>Qasper</b>	<i>Question Answering</i>
<b>Input</b>	<p>Which languages are used in the multi-lingual caption model?</p> <p>Introduction</p> <p>The bilingual lexicon induction task aims to automatically build word translation dictionaries across different languages, which is beneficial for various natural language processing tasks such as cross-lingual information...</p>
<b>Output</b>	<p>German-English, French-English, and Japanese-English</p>

<https://paperswithcode.com/dataset/scrolls>

# Pretrain Pegasus



- Gap Sentences Generation (GSG): Một số lượng câu được chọn(MASK -1) để cho vào tạo nên văn bản tóm tắt (gap sentences)
- Masked Language Model (MLM): Một lượng tokens được chọn (MASK-2) để train cho encoder của mô hình.

# Long input handling



- Đối với long input pretraining, nhóm tác giả thực hiện mở rộng chiều dài input cho phép lên 4096 tokens, điều chỉnh mask ratio từ 45% còn 5.625% để tăng độ dài input và chỉ chọn document dài lớn 10000 chữ.
  - Dataset arXiv, độ dài input được tinh chỉnh lên đến 16384 tokens và 256 output tokens.
  - Dataset GovReport, input có độ dài 10240 input tokens và 1024 output tokens tạo nên nhiệm vụ tóm tắt cho ra văn bản dài hơn.
  - Dataset XSUM và CNN/Daily Mail, input có độ dài 512, và độ dài output lần lượt là 64 và 128.

# Blockwise Transformer



- Thực nghiệm trên các biến thể của Transformer (**Transformer variants**), và đưa ra kiến trúc **blockwise local Transformer đơn giản hơn** với các khối rời rạc (**staggered blocks**) và các khối toàn cục (**global tokens**), tạo ra sự cân bằng tốt giữa hiệu suất và hiệu quả bộ nhớ.
  - A block-local Transformer (Local): Các encoder input tokens được chia thành các block không giao nhau, tokens chỉ có thể tham gia với các token bên trong block..
  - A global tokens with learnable embeddings: có thể tham gia vào bất kỳ token nào (Global-Local)

# Blockwise Transformer



- Thực nghiệm trên các biến thể của Transformer (**Transformer variants**), và đưa ra kiến trúc **blockwise local Transformer đơn giản hơn** với các khối rời rạc (**staggered blocks**) và các khối toàn cục (**global tokens**), tạo ra sự cân bằng tốt giữa hiệu suất và hiệu quả bộ nhớ.
  - Cả local -global attention đều cải thiện hiệu suất và hiệu quả sử dụng bộ nhớ.
  - Staggering local attention blocks cải thiện hiệu suất cao.

Encoder	Stagger Local Blocks	Use Global In Decoder	arXiv		GovReport	
			R1 / R2 / RL	RG	R1 / R2 / RL	RG
Global-Local	✓	✓	<b>48.1 / 20.3 / 28.5</b>	<b>30.3</b>	60.5 / 28.8 / 30.5	37.6
Global-Local		✓	47.0 / 19.5 / 27.9	29.5	60.9 / 28.9 / 30.2	37.6
Global-Local	✓		47.7 / 20.4 / 28.6	30.3	<b>61.3 / 29.4 / 30.8</b>	<b>38.1</b>
Global-Local			46.7 / 19.5 / 27.9	29.4	59.5 / 27.8 / 29.4	36.5
Local	✓	-	46.8 / 19.7 / 28.0	29.6	59.2 / 27.9 / 30.0	36.7
Local		-	46.5 / 19.2 / 27.5	29.1	58.8 / 27.5 / 28.9	36.0

Table 2: Comparison of architectural tweaks to Local and GlobalLocal encoder. Staggering local blocks uses different blocks boundaries for different layers in block-local attention. Global information is incorporated in the decoder via an additional cross-attention before cross-attention over the encoded input.

# Blockwise Transformer



- Thực nghiệm trên các biến thể của Transformer (**Transformer variants**), và đưa ra kiến trúc **blockwise local Transformer đơn giản hơn** với các khối rời rạc (**staggered blocks**) và các khối toàn cục (**global tokens**), tạo ra sự cân bằng tốt giữa hiệu suất và hiệu quả bộ nhớ.
  - Kích thước khối lớn hơn và/hoặc số lượng global token dẫn đến hiệu suất được cải thiện.

Block Size	Global Tokens	arXiv			GovReport			
		R1 / R2 / RL	RG	R1 / R2 / RL	RG	Steps/s	Mem	
4	8	46.2 / 19.1 / 27.5	29.0	60.1 / 28.0 / 29.7	36.8	0.77	1.27	
	32	46.1 / 18.8 / 27.2	28.7	60.1 / 27.6 / 28.9	36.3	0.65	1.70	
	64	- / - / -	-	60.1 / 27.7 / 29.0	36.4	-	-	
	128	- / - / -	-	- / - / -	-	-	-	
16	8	46.9 / 19.6 / 27.9	29.5	60.1 / 28.2 / 29.7	36.9	0.98	1.03	
	32	47.1 / 20.0 / 28.3	29.9	59.7 / 27.8 / 29.2	36.5	0.92	1.15	
	64	46.8 / 19.7 / 28.0	29.6	60.8 / 28.6 / 30.0	37.4	0.75	1.54	
	128	47.7 / 20.0 / 28.2	30.0	60.7 / 28.8 / 30.2	37.5	0.58	1.70	
64	8	46.8 / 19.8 / 28.0	29.6	61.2 / 28.8 / 30.2	37.6	0.98	1.06	
	32	47.7 / 20.3 / 28.5	30.2	61.0 / 29.3 / 30.8	38.0	0.47	1.07	
	64	47.4 / 20.2 / 28.5	30.1	60.9 / 29.1 / 30.7	37.9	0.94	1.10	
	128	<b>47.8 / 20.4 / 28.6</b>	<b>30.3</b>	60.9 / 29.0 / 30.3	37.7	0.85	1.26	
128	8	- / - / -	-	- / - / -	-	-	-	
	32	46.9 / 19.7 / 28.0	29.6	60.9 / 28.7 / 30.1	37.5	<b>1.00</b>	<b>1.00</b>	
	64	47.4 / 20.2 / 28.4	30.1	60.9 / 28.9 / 30.8	37.8	0.96	1.05	
	128	47.1 / 20.0 / 28.3	29.9	61.0 / 28.9 / 30.6	37.8	0.90	1.15	
256	8	46.8 / 20.0 / 28.2	29.8	60.7 / 29.3 / 30.9	38.0	0.96	1.07	
	32	47.3 / 20.2 / 28.3	30.0	61.6 / 29.4 / 30.7	38.2	0.92	1.11	
	64	47.2 / 20.2 / 28.4	30.0	59.2 / 28.6 / 30.5	37.2	0.88	1.16	
	128	48.1 / 20.5 / 28.6	30.4	<b>61.7 / 29.3 / 30.8</b>	<b>38.2</b>	0.83	1.26	
512	8	- / - / -	-	- / - / -	-	-	-	
	32	46.7 / 19.7 / 28.1	29.6	59.8 / 28.2 / 29.8	36.9	0.77	1.35	
	64	47.2 / 20.1 / 28.2	29.9	61.1 / 29.3 / 30.7	38.0	0.75	1.40	
	128	47.2 / 20.0 / 28.2	29.9	61.0 / 29.3 / 30.7	38.0	0.71	1.51	

Table 3: Varying the block size and the number of global tokens of a GlobalLocal encoder. Training steps per second and memory are computed based on arXiv, and normalized to the run with Block Size=128 and Global Tokens=32.

# Design choices

- Các tác giả cũng thực hiện xem xét một số lựa chọn thiết kế mô hình khác như:
  - Position encoding schemes (relative position encodings such as RoPE, T5 and ALiBi)



Position Encoding	XSUM			CNN/DM			arXiv			GovReport		
	R1 / R2 / RL	RG	R1 / R2 / RL	RG	Step/s							
None	34.3 / 12.5 / 26.8	22.6	25.6 / 7.8 / 17.7	15.2	36.1 / 9.8 / 22.0	19.8	38.3 / 13.2 / 18.7	21.1	0.96			
Sinusoidal	39.8 / 16.9 / 31.8	27.8	<b>40.0 / 18.6 / 28.4</b>	<b>27.6</b>	44.5 / 17.6 / 26.7	27.6	40.0 / 18.8 / 22.3	25.6	0.96			
T5	<b>40.1 / 17.1 / 32.0</b>	<b>28.0</b>	39.8 / 18.8 / 28.6	27.8	<b>44.9 / 17.9 / 26.8</b>	<b>27.8</b>	<b>40.2 / 19.5 / 22.9</b>	<b>26.2</b>	0.53			
RoPE	39.8 / 16.9 / 31.8	27.8	39.2 / 18.7 / 28.5	27.5	43.5 / 17.2 / 26.5	27.1	40.0 / 19.1 / 22.6	25.8	0.85			
Absolute	39.1 / 16.4 / 31.3	27.2	39.7 / 18.7 / 28.5	27.7	44.3 / 17.5 / 26.5	27.4	38.6 / 17.5 / 21.1	24.2	<b>1.00</b>			

Table 4: Comparison of position encodings schemes for a Transformer encoder-decoder. Training steps per second are computed based on arXiv summarization. Absolute position embeddings are replicated to longer input sequences, following [Beltagy et al. \(2020\)](#). Training steps per second is computed based on arXiv, and normalized to the run with absolute position embeddings.

# Design choices



- Các tác giả cũng thực hiện xem xét một số lựa chọn thiết kế mô hình khác nhau:
  - Encoder-decoder layer distributions: Do tính bất đối xứng của chiều dài input (long document) và output (short summary), có sự đánh đổi tính toán khác nhau để cân bằng khác nhau của các lớp decoder và encoder:
    - Mô hình có nhiều lớp encoder (Encoder-heavy) yêu cầu nhiều bộ nhớ cho các chuỗi input dài
    - Mô hình nhiều lớp Decoder (Decoder-heavy) inference chậm hơn do bản chất autoregressive của decoding.

Architecture	Enc	Dec	XSUM			CNN/DM			arXiv			GovReport		
			R1 / R2 / RL	RG	R1 / R2 / RL	RG	R1 / R2 / RL	RG						
Local	18	6	37.4 / 15.0 / 29.7	25.5	39.0 / 18.2 / 27.9	27.0	46.0 / 19.4 / 27.6	29.1	58.9 / 27.4 / 29.1	36.1				
	12	12	37.5 / 14.9 / 29.7	25.5	38.5 / 18.0 / 27.6	26.7	45.4 / 18.9 / 27.3	28.6	59.2 / 27.6 / 29.3	36.3				
	6	18	37.7 / 15.1 / 29.9	25.7	38.5 / 18.1 / 27.7	26.9	46.3 / 19.3 / 27.6	29.1	59.4 / 27.8 / 29.5	36.5				
Global-Local	18	6	<b>38.6 / 15.9 / 30.9</b>	<b>26.7</b>	39.2 / 18.5 / 28.2	27.3	47.3 / 20.1 / 28.3	30.0	60.2 / 28.7 / 30.6	37.5				
	12	12	38.6 / 15.9 / 30.7	26.6	<b>40.0 / 18.6 / 28.3</b>	<b>27.6</b>	47.5 / 20.1 / 28.3	30.0	<b>61.1 / 29.3 / 30.7</b>	<b>38.1</b>				
	6	18	37.7 / 15.1 / 29.9	25.7	38.5 / 18.1 / 27.7	26.9	46.4 / 19.5 / 27.9	29.3	60.3 / 28.6 / 30.0	37.2				
Global-Local	18	12	38.5 / 15.7 / 30.6	26.4	38.7 / 18.4 / 28.1	27.1	47.3 / 20.0 / 28.3	29.9	60.2 / 29.2 / 31.0	37.9				
	12	18	38.6 / 15.8 / 30.5	26.5	38.6 / 18.3 / 28.0	27.0	<b>47.5 / 20.3 / 28.5</b>	<b>30.2</b>	60.9 / 29.0 / 30.4	37.7				

Table 5: Varying the distribution of encoder/decoder layers)

# Design choices

- Các tác giả cũng thực hiện xem xét một số lựa chọn thiết kế mô hình khác như:
  - Partial Cross Attention: Thực nghiệm chỉ có cross attention ở các lớp decoder:
    - Giảm cross-attention cho 1 phần các lớp decoder có thể giảm tiêu thụ bộ nhớ.



Cross-Attention	Model	arXiv			GovReport	
		R1 / R2 / RL	RG	R1 / R2 / RL	RG	
Pretrained	Full	47.7 / 20.4 / 28.6	30.3	<b>61.3 / 29.4 / 30.8</b>	<b>38.1</b>	
	Cross[0,2,4,6,8,10]	<b>48.1 / 20.4 / 28.6</b>	<b>30.4</b>	61.0 / 29.0 / 30.7	37.9	
	Cross[0,6]	47.1 / 19.8 / 28.1	29.7	60.4 / 28.1 / 29.7	36.9	
Converted	Cross[0,2,4,6,8,10]	46.4 / 19.7 / 28.1	29.5	60.2 / 28.8 / 30.3	37.4	
	Cross[0,6]	46.2 / 19.7 / 28.1	29.5	60.2 / 28.1 / 29.8	36.9	

Table 9: Comparison of models pretrained with cross-attention for a subset of layers, and adapting a pretrained model by dropping cross-attention layers only during fine-tuning

# Pretrained and Fine-tune



- Các tác giả cũng thực hiện xem xét một số lựa chọn thiết kế mô hình khác nhau:
  - So sánh thực nghiệm các mô hình fine-tune và pretrained
    - Với ngân sách điện toán cố định, việc phân bổ một phần để huấn luyện cho đầu vào long document có thể cải thiện hiệu suất.
    - Chỉ huấn luyện trên long document làm giảm hiệu suất của mô hình

Pretraining Scheme	Encoder	XSUM		CNN/DM		arXiv		GovReport	
		R1 / R2 / RL	RG						
Short (50%)	Local	38.4 / 15.8 / 30.6	26.5	39.2 / 18.1 / 27.9	27.1	46.8 / 19.7 / 28.0	29.6	60.1 / 28.3 / 29.8	37.0
	Global-Local	39.4 / 16.5 / 31.5	27.4	39.1 / 18.6 / 28.3	27.4	47.7 / 20.4 / 28.6	30.3	61.9 / 29.6 / 30.8	38.4
Short (100%)	Local	39.2 / 16.3 / 31.3	27.1	39.2 / 18.6 / 28.3	27.4	46.9 / 19.7 / 28.0	29.6	60.1 / 28.3 / 29.8	37.0
	Global-Local	<b>39.9 / 17.0 / 31.9</b>	<b>27.9</b>	39.8 / 18.6 / 28.3	27.6	48.1 / 20.5 / 28.7	30.5	61.9 / 29.6 / 30.8	38.4
Short (75%) → Long (25%)	Local	38.8 / 15.9 / 30.7	26.7	39.1 / 18.2 / 28.0	27.1	47.5 / 20.1 / 28.2	30.0	60.6 / 28.9 / 30.6	37.7
	Global-Local	39.6 / 16.8 / 31.7	27.6	<b>39.8 / 18.8 / 28.5</b>	<b>27.7</b>	48.4 / 20.7 / 28.8	30.7	61.8 / 29.8 / 31.1	38.5
Short (50%) → Long (50%)	Local	38.4 / 15.7 / 30.5	26.4	39.4 / 18.1 / 27.9	27.1	47.7 / 20.2 / 28.3	30.1	60.9 / 29.1 / 30.7	37.9
	Global-Local	39.3 / 16.4 / 31.4	27.3	39.4 / 18.3 / 28.1	27.3	<b>48.4 / 20.9 / 29.1</b>	<b>30.9</b>	<b>61.7 / 30.0 / 31.2</b>	<b>38.7</b>
Long (100%)	Local	36.0 / 14.0 / 28.6	24.3	38.4 / 17.7 / 27.4	26.5	46.7 / 19.5 / 27.7	29.3	59.8 / 28.0 / 29.5	36.7
	Global-Local	36.4 / 14.3 / 28.9	24.7	38.5 / 17.8 / 27.5	26.6	47.3 / 19.9 / 28.1	29.8	61.1 / 29.1 / 30.7	37.9

Table 7: Comparison of different pretraining formats, given a input token budget of 131B tokens, which corresponds to 1M steps with 512 input tokens. Short pretraining uses 512 input tokens, whereas long pretraining uses 4096 input tokens.

# Long Input Summarization



- Kết quả thực nghiệm cho thấy rằng với ngân sách định, **pretrained** trên các chuỗi ngắn và sau đó điều chỉnh lại (**pre-adapting**) mô hình thành kiến trúc Transformer hiệu quả trên chuỗi dài và thực hiện các bước các bước huấn luyện bổ sung dẫn đến hiệu suất vượt trội so với chỉ huấn luyện cho chuỗi dài hoặc không có bất kỳ điều chỉnh nào.



05



## Experiment and conclusion

Sơ lược thực nghiệm và kết luận

# Metrics for evaluation



- **ROUGE** là viết tắt của Recall-Oriented Understudy for Gisting Evaluation. Nó hoạt động bằng cách so sánh một bản tóm tắt hoặc bản dịch được tạo tự động với một tập hợp các bản tóm tắt tham khảo.
  - **ROUGE-N (N-gram Co-Occurrence Statistics)**: Dùng để đo sự trùng lặp n-gram giữa bản tóm tắt được tạo tự động và các bản tóm tắt tham chiếu.
  - **ROUGE-L**: Biện pháp này sử dụng khái niệm Chuỗi con chung dài nhất (LCS) giữa hai tài liệu. Theo trực giác, LCS giữa hai tài liệu tóm tắt càng dài thì chúng càng giống nhau.
- Các tác giả báo cáo điểm số **validator score** tốt nhất dựa trên giá trị trung bình hình nhân (RG) của các điểm số ROUGE-1, ROUGE-2 và ROUGE-L.



# Summarization tasks set up

- Sử dụng kiến trúc Global-Local với khối nằm so le, số lượng lớn global token và kích thước khối lớn trong quá trình pre-training
- Tiến hành một giai đoạn bổ sung huấn luyện long input trên 4096 token đầu vào trong 300,000 bước.
- Mở rộng chuỗi đầu vào lên tới 16384 token đầu vào để tinh chỉnh, tùy thuộc vào tác vụ cần phải xử lý.

	PEGASUS-X <sub>Base</sub>	PEGASUS-X
# Parameters	272M	568M
# Global Tokens	128	128
Block Size	512	512
Batch Size	512	1024
Additional Pretraining	300K steps	300K steps

Table 10: Hyperparameters of Pegasus-X Models

# Results on 3 Summarization tasks

Model	#Params	arXiv		Big Patent		PubMed	
		R1 / R2 / RLs	RG	R1 / R2 / RLs	RG	R1 / R2 / RLs	RG
PEGASUS <sub>Base</sub>	271M	34.8 / 10.2 / 22.5*	20.0*	43.5 / 20.4 / 31.8*	30.5*	40.0 / 15.2 / 25.2*	24.8*
PEGASUS <sub>Base+</sub>	271M	42.2 / 15.8 / 37.3	29.2	51.2 / 32.6 / 41.0	40.9	44.1 / 18.3 / 40.1	31.9
PEGASUS <sub>Base+</sub> + Global-Local	272M	47.6 / 20.2 / 42.4	34.4	58.1 / 39.5 / 47.2	47.7	47.3 / 21.4 / 43.0	35.2
PEGASUS-X <sub>Base</sub>	272M	49.4 / 21.6 / 44.0	36.1	61.3 / 42.6 / 50.1	50.8	49.6 / 23.6 / 45.2	37.5
PEGASUS <sub>Large</sub>	567M	44.7 / 17.2 / 25.7*	27.0*	53.4 / 32.9 / 42.1*	42.0*	45.1 / 19.6 / 27.4*	28.9*
PEGASUS-X	568M	50.0 / 21.8 / 44.6	36.5	64.8 / 47.5 / 54.3	55.1	<b>51.0 / 24.7 / 46.6</b>	<b>38.9</b>
Longformer Encoder-Decoder	464M	46.6 / 19.6 / 41.8	33.7	-. / -. / .-	-.	-. / -. / .-	-.
Top-Down (AvgP)	464M	48.7 / 20.7 / 43.9	35.4	-. / -. / .-	-.	48.3 / 21.4 / 44.2	35.7
Top-Down (AdaP)	464M	<b>51.0 / 21.9 / 45.6</b>	<b>37.1</b>	-. / -. / .-	-.	51.1 / 23.3 / 46.5	38.1
Big Bird-Pegasus	567M	46.6 / 19.0 / 41.8	33.3	60.6 / 42.5 / 50.1	50.5	46.3 / 20.7 / 42.3	34.4
LongT5 <sub>Large</sub>	770M	48.3 / 21.6 / 44.1	35.8	70.4 / 56.8 / 62.7	63.1	50.0 / 24.7 / 46.5	38.6
LongT5 <sub>XL</sub>	3B	48.4 / 21.9 / 44.3	36.1	<b>76.9 / 66.1 / 70.8</b>	<b>71.1</b>	50.2 / 24.8 / 46.7	38.7

Table 11: Comparison on long summarization tasks (Test sets). Results for other models are taken from their respective papers. \*: PEGASUS (Zhang et al., 2020) only reports ROUGE-L and not ROUGE-LSum.

# Results on SCROLLS tasks

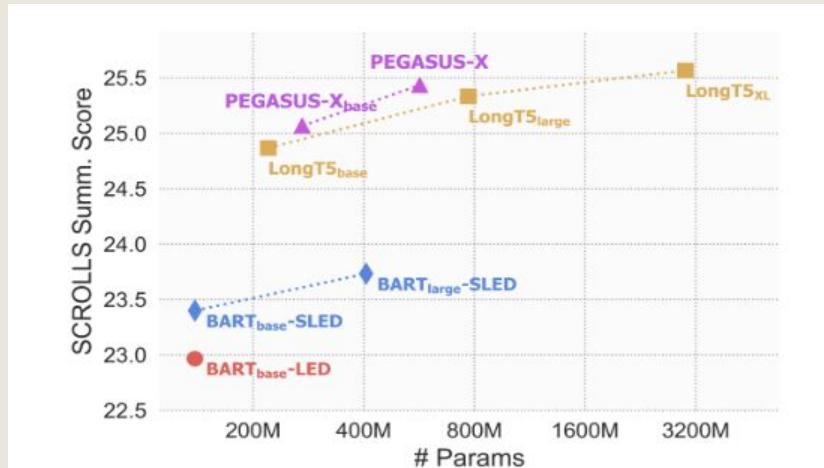


Figure 1: Model scores on SCROLLS (Shaham et al., 2022) summarization tasks. All models evaluated on up to 16K input tokens. PEGASUS-X outperforms other models at comparable model sizes. Scores are computed by taking the average of the geometric mean of ROUGE-1/2/L.





A-Z

06

# Reference

Các tài liệu tham khảo



# Reference

- [1] Phang, Jason, Yao Zhao, and Peter J. Liu. "Investigating Efficiently Extending Transformers for Long Input Summarization." arXiv preprint arXiv:2208.04347 (2022).
- [2] Zhang, Jingqing, Yao Zhao, Mohammad Saleh, and Peter Liu. "Pegasus: Pre-training with extracted gap-sentences for abstractive summarization." In International Conference on Machine Learning, pp. 11328-11339. PMLR, 2020.
- [3] Koh, Huan Yee, Jiaxin Ju, Ming Liu, and Shirui Pan. "An empirical survey on long document summarization: Datasets, models and metrics." ACM Journal of the ACM (JACM) (2022).

Thank  
you!

Any  
**Question**

A large blue question mark icon with a light effect, consisting of a central circle with a vertical bar, surrounded by a semi-circle and small blue squares radiating outwards.