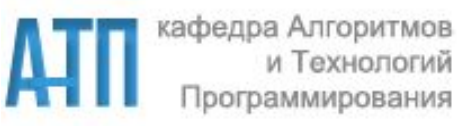


Summarization

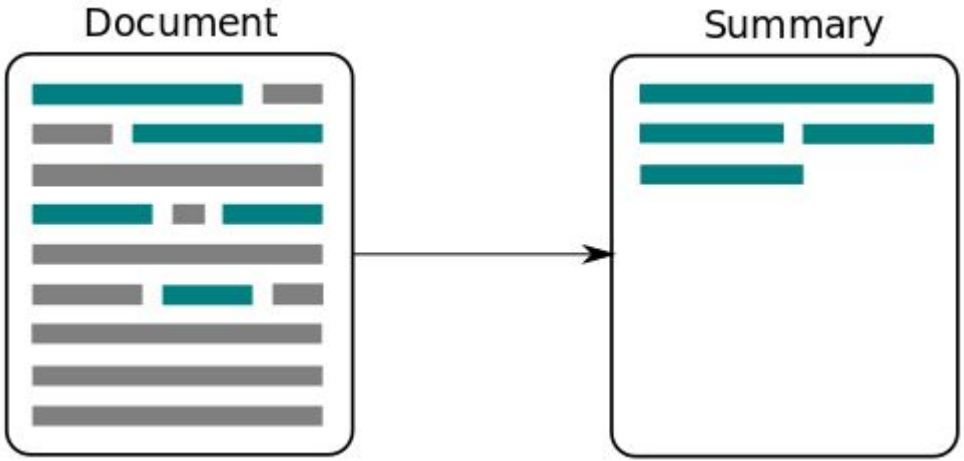
MIPT
14.04.2021
Anton Emelianov

https://github.com/king-menin/mipt-nlp2022

Summarization. Task

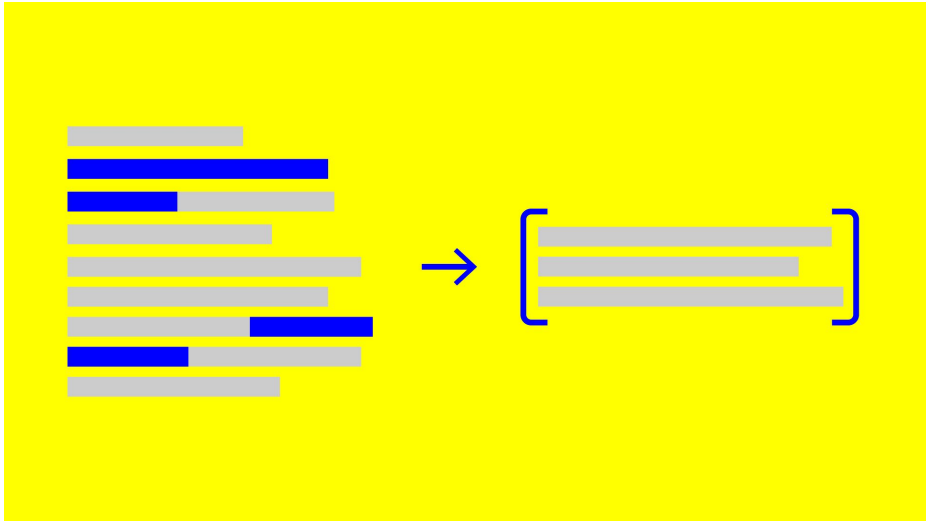
Summarization is the task of condensing a piece of text to a shorter version, reducing the size of the initial text while at the same time preserving key informational elements and the meaning of content.

- data reduction
- important/key information
- the same meaning



Today

- Summarization
  - types
  - metrics
  - methods
- Paraphrasing
- Simplification



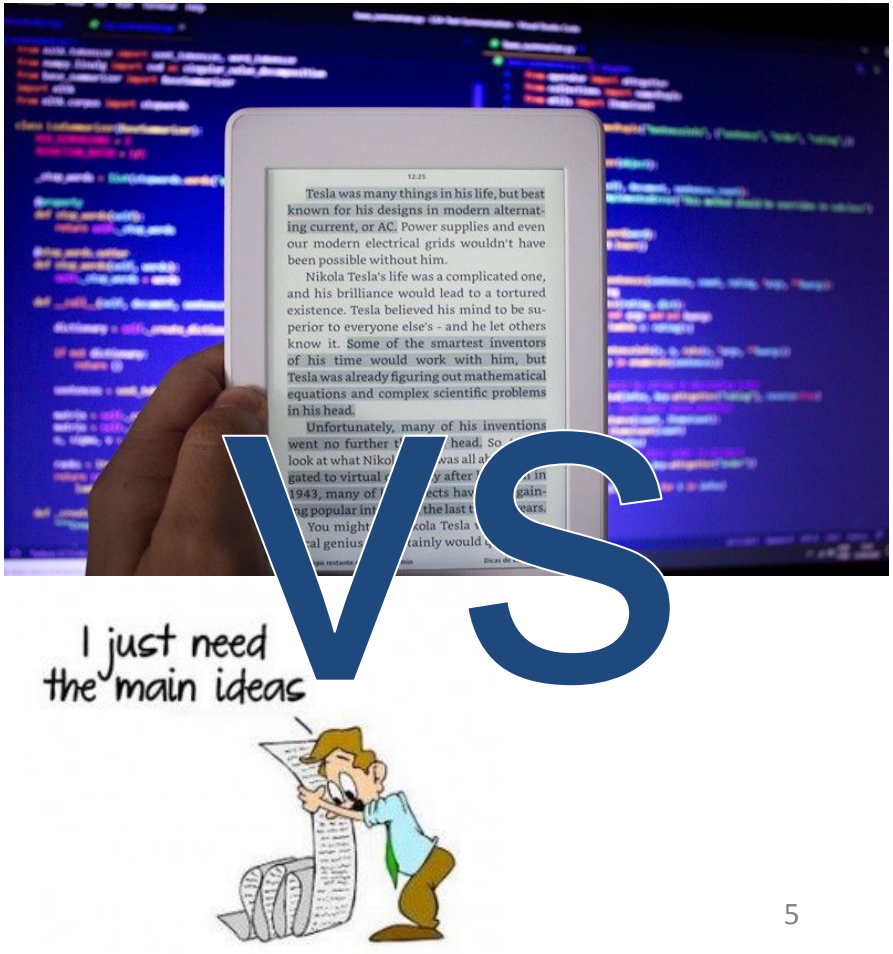
Summarization. Applications

- News
- Books/series/referats summaries
- Documents sum
- Media monitoring
- Video scripting
- Emails overload
- Financial research
- E-learning and class assignments
- in chatbots
- etc.

Summarization. Types

Extractive Summary: the network calculates the most important sentences from the article and gets them together to provide the most meaningful information from the article.

Abstractive Summary: The network creates new sentences to encapsulate maximum gist of the article and generates that as output. The sentences in the summary may or may not be contained in the article.



Summarization. Types. Formats

Document summarization (extreme)

Many documents or huge texts into very short form.

Sentence Compression

Sentence: Floyd Mayweather is open to fighting Amir Khan in the future, despite snubbing the Bolton-born boxer in favour of a May bout with Argentine Marcos Maidana, according to promoters Golden Boy

Compression: Floyd Mayweather is open to fighting Amir Khan in the future.

Summarization. Types

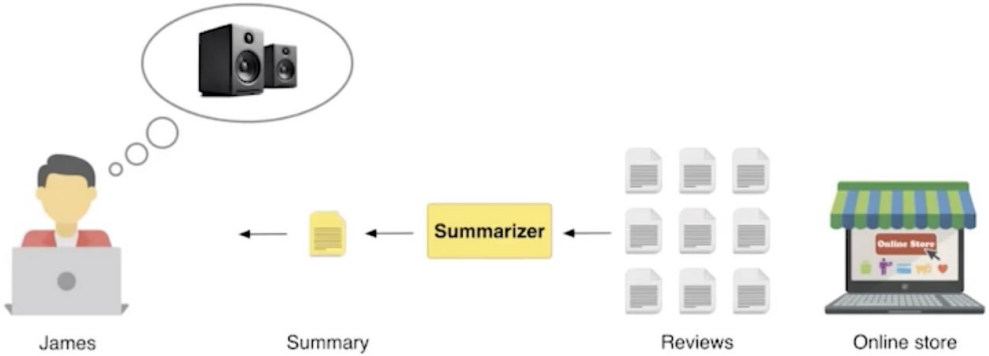
Opinions summarization - lots of opinions need to sum in one join opinion.

Opinion summary should be:

(1) centered on entities and aspects

and sentiments about them

(2) quantitative



Contrastive summarization (for some style) jointly generating summaries for two entities in order to highlight their differences.
(read here)

Summarization. Metrics. ROUGE

Recall: $$\frac{|\text{ngrams}(ref) \ \& \ \text{ngrams}(hyp)|}{|\text{ngrams}(ref)|}$$

Precision: $$\frac{|\text{ngrams}(ref) \ \& \ \text{ngrams}(hyp)|}{|\text{ngrams}(hyp)|}$$

F1: $$2\frac{P * R}{R + P}$$

ROUGE-N: Overlap of N-grams

ROUGE-1 refers to the overlap of ***unigram*** *(each word)* between the system and reference summaries; ROUGE-2 - of ***bigrams***.

- compute unique ngrams
- check overlap and length
- => F1 mesuare

ROUGE-L: Longest Common Subsequence (LCS) based statistics. Longest common subsequence problem takes into account sentence level structure similarity naturally and identifies longest co-occurring in sequence n-grams automatically.

Better for abstractive! FLUENCY

ROUGE-W: Weighted LCS-based statistics that favors consecutive LCSes .

ROUGE-S: Skip-bigram based co-occurrence statistics. Skip-bigram is any pair of words in their sentence order.

ROUGE-SU: Skip-bigram plus unigram-based co-occurrence statistics.

Summarization. Metrics. Meteor

The Meteor automatic evaluation metric scores machine translation and other generation tasks hypotheses by aligning them to one or more references.

Alignments are based on exact, stem, synonym, and paraphrase matches between words and phrases.

Weighted F-score $F = \frac{PR}{\alpha P + (1-\alpha)R)}$

Penalty function for incorrect word order $Penalty = \gamma(\frac{c}{m})^{\beta}, \text{ where } 0 \leq \gamma \leq 1$

$$Score = Fmean * (1 - Penalty)$$

Summarization. Datasets

English:

- CNN / Daily Mail (single document, many extractive)
- X-Sum (single doc, short summaries)
- Newsroom
- MultiNews (multi documents)
- DUC 2004 Task 1
- Webis-TLDR-17 Corpus
- Gigaword
- BIGPATENT https://www.aclweb.org/anthology/P19-1212/

Russian

- MLSUM (CNN/Daily)
- Gazeta. Russian News https://github.com/IlyaGusev/gazeta

Extractive datasets. Lifehack:

- utilize abstractive sum datasets
- select sentences that have max ROUGE scores

Summarization. Extractive methods

Ussually framed as tagging problem.

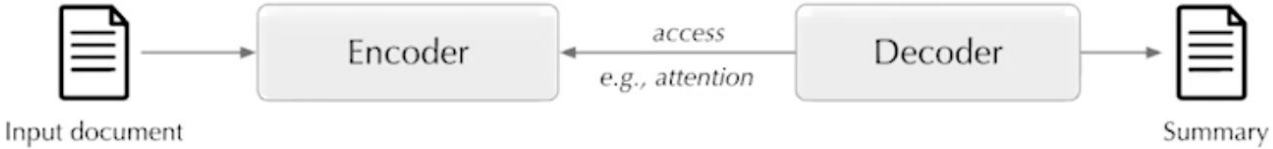
- Given document *D*.
- Select *K* summaring (most important) fragments
- Concatenate K fragments in summary

Methods:

- LSA (Latent semantic analysis)
- Luhn Summarization algorithm (tf-idf)
- LexRank
- ...
- As binary classification assign tags 0 or 1 to important sentences.
  Neural encoder => sentence semantic representaion => sigmoid

Summarization. Abstractive methods

Abstract answer => Generation



Encoder-decoder architectures

BertSum, BART, T5

Or just decoders GPT-2, GPT-3

Pros:

- richer vocabulary
- abstract/rephrase
- conflict info/opinions

NEED DATA

Summarization. Methods

Pretrained models / Fine-tuning

BertSum (extractive)

BertSum assigns scores to each sentence that represents how much value that sentence adds to the overall document. So, [s1,s2,s3] is assigned [score1, score2, score3]. The sentences with the highest scores are then collected and rearranged to give the overall summary of the article.

- Based on a pre-trained encoder (Liu and Lapata, 2019)
- Use a pre-trained BERT encoder (Devlin et al., 2019)
- BertSum has a transformer encoder-decoder architecture
- The decoder is trained from scratch

Summarization. Methods

Pretrained models / Fine-tuning

<u>BertSum (extractive)</u>

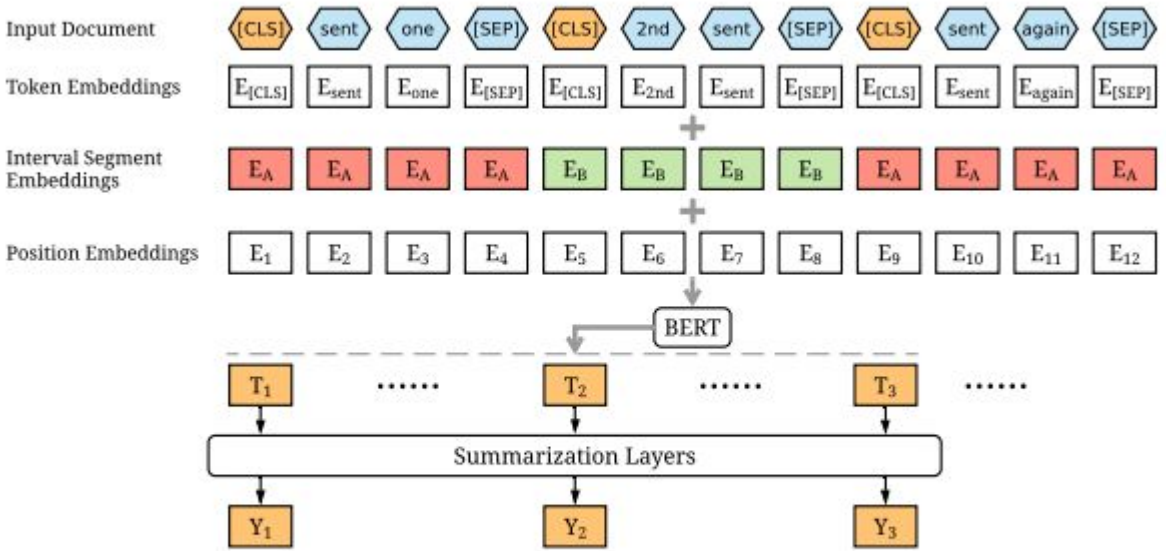


Figure 1: The overview architecture of the BERTSUM model.

Summarization. Abstractive methods

Pointer generation

Augment the standart attention module

Attention mechanism

COPY mechanism

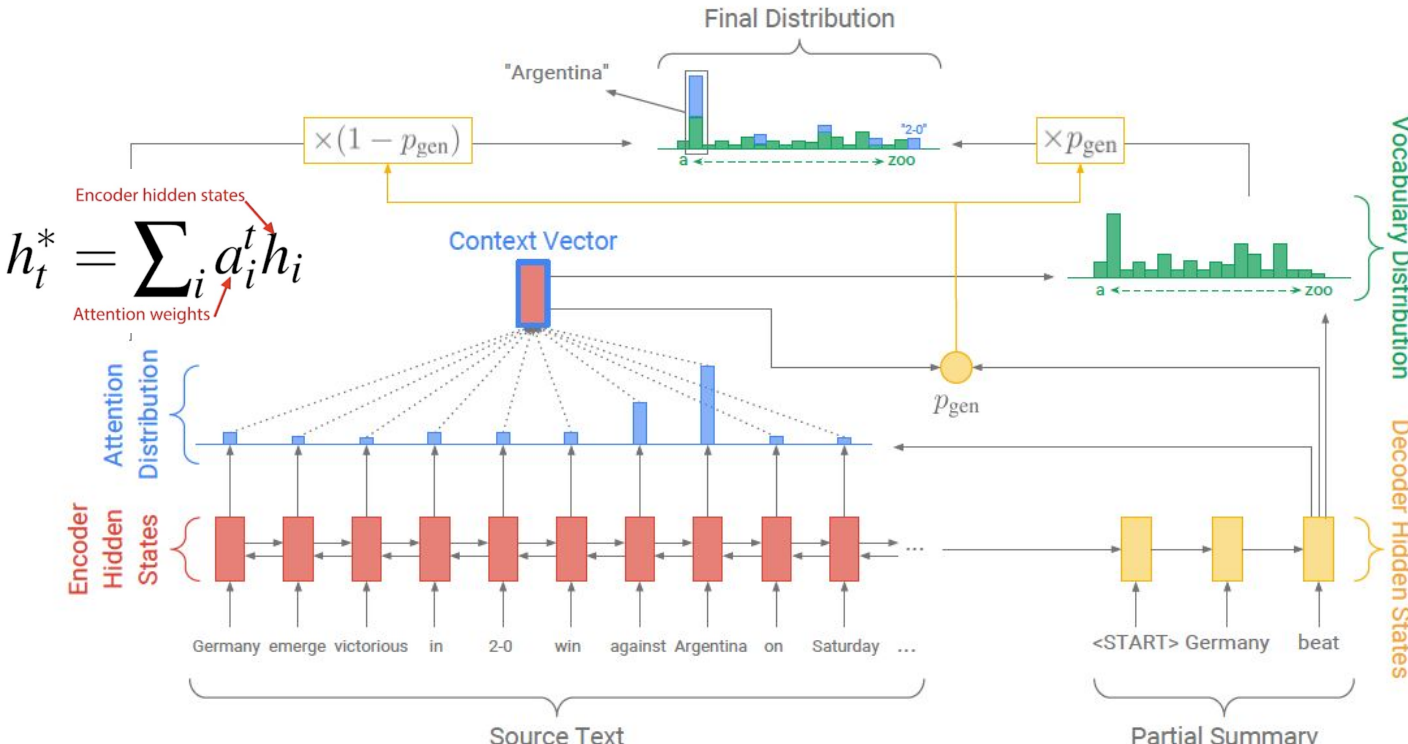
(pointer network)

solve OOV problems

Decoder hidden states

$$P_{\text{vocab}} = \text{softmax}(V'(V[s_t, h_t^*] + b) + b')$$

Context vector



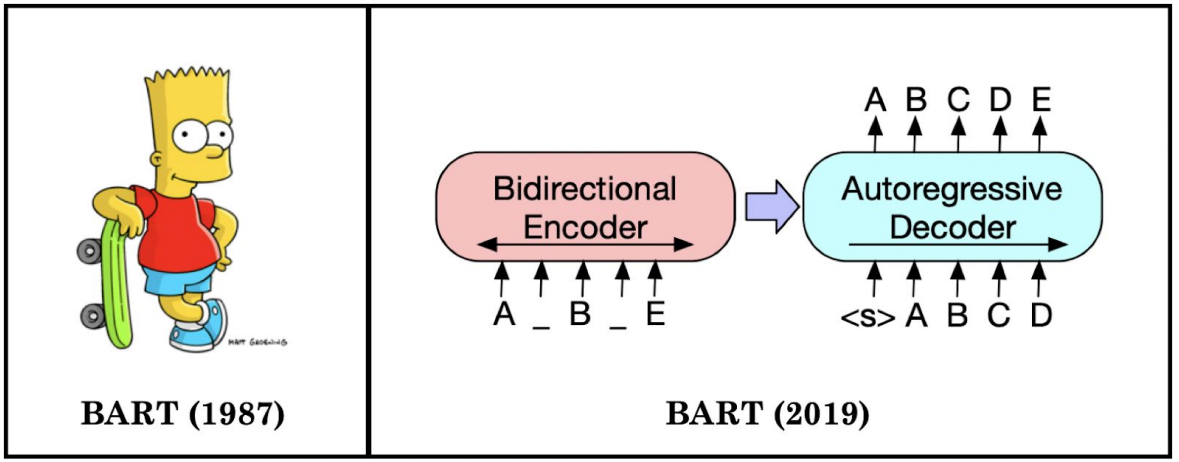
Summarization. Abstractive methods

BART

Encoder + decoder

MBART (multilingual variant, Russian included)

Unsupervised denoising objective



	CNN/DailyMail			XSum		
	R1	R2	RL	R1	R2	RL
Lead-3	40.42	17.62	36.67	16.30	1.60	11.95
PTGEN see:2017	36.44	15.66	33.42	29.70	9.21	23.24
PTGEN+COV see:2017	39.53	17.28	36.38	28.10	8.02	21.72
UniLM	43.33	20.21	40.51	-	-	-
BERTSUMABS (bertsum)	41.72	19.39	38.76	38.76	16.33	31.15
BERTSUMEXTABS (bertsum)	42.13	19.60	39.18	38.81	16.50	31.27
BART	**44.16**	**21.28**	**40.90**	**45.14**	**22.27**	**37.25**

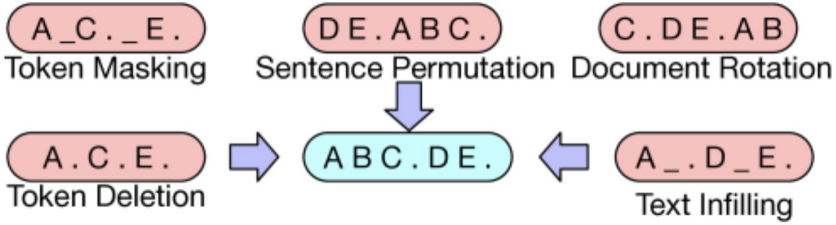


Figure 2: Transformations for noising the input that we experiment with. These transformation

Summarization. Abstractive methods

Pegasus by Google

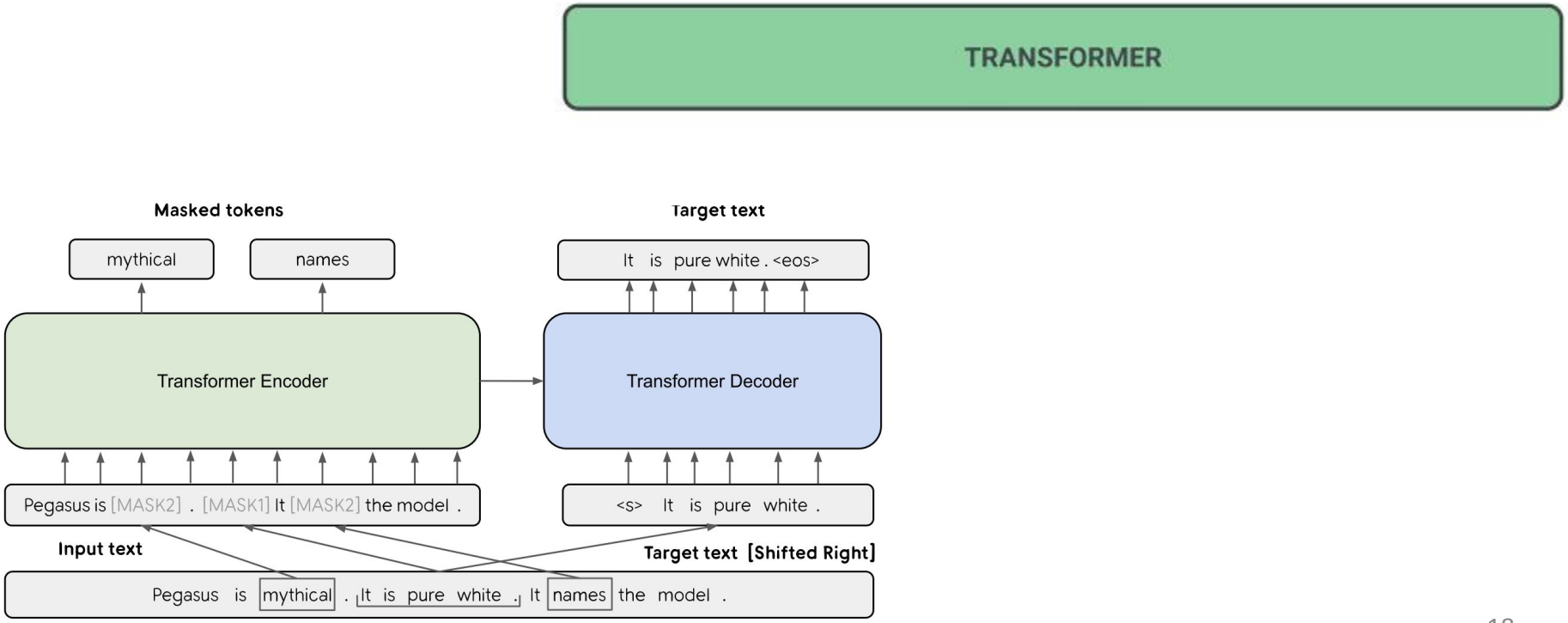
Gap Sentence Generation (GSG) + MLM
Complete sentences are removed from a document (i.e. they are 'masked'), and the **model is trained to predict these masked sentences**.
Choosing the most important sentences from the document for masking works best. This is done by finding sentences that are the most similar to the complete document.
Three strategies to select gap sentences (without replacement):

1) Random
2) Lead
3) Principal (selecting top-m scored sentences based on their importance, - measured by the ROUGE-1 score between the sentence and the rest of the document).

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 words are randomly masked by [MASK2] (MLM).

https://arxiv.org/abs/1912.08777 https://ai.googleblog.com/2020/06/pegasus-state-of-art-model-for.html

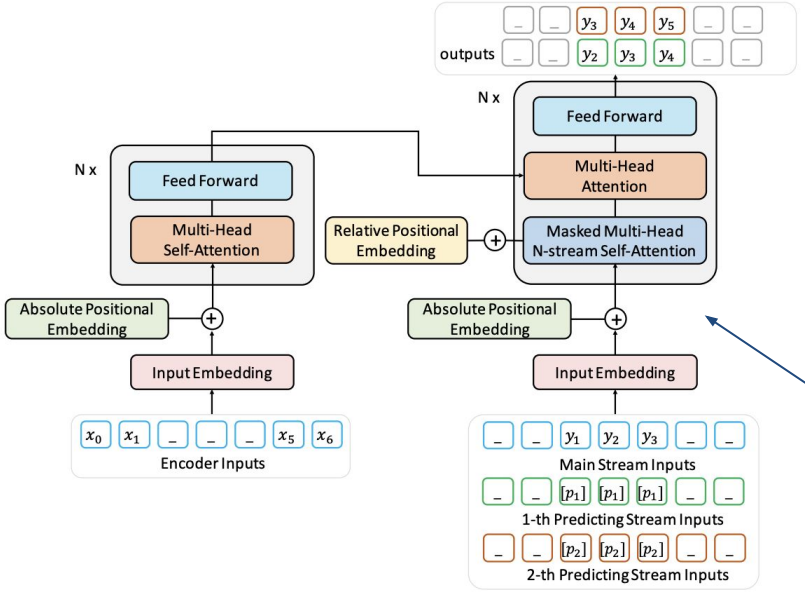
Summarization. Abstractive methods

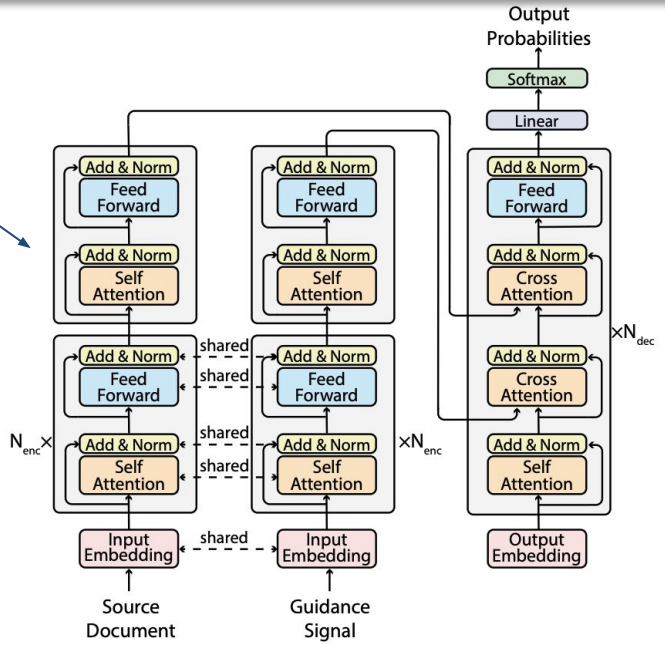


https://arxiv.org/abs/1912.08777 https://ai.googleblog.com/2020/06/pegasus-state-of-art-model-for.html

Summarization. Abstractive methods

GSum - general and extensible guided summarization framework that can effectively take external various types of guidance signals.
https://arxiv.org/pdf/2010.08014v1.pdf





ProphetNet is an encoder-decoder model and can predict n-future tokens for “ngram” language modeling instead of just the next token.

https://arxiv.org/pdf/2001.04063v3.pdf

http://nlpprogress.com/english/summarization.html

Abstractive methods

Cnn/Daily

Model	ROUGE-1	ROUGE-2	ROUGE-L
PEGASUS (Zhang et al., 2019)	47.21	24.56	39.25
BART (Lewis et al., 2019)	45.14	22.27	37.25

Xsum

Model	ROUGE-1	ROUGE-2	ROUGE-L
GSum (Dou et al., 2020)	45.94	22.32	42.48
ProphetNet (Yan, Qi, Gong, Liu et al., 2020)	44.20	21.17	41.30
PEGASUS (Zhang et al., 2019)	44.17	21.47	41.11
BART (Lewis et al., 2019)	44.16	21.28	40.90

http://nlpprogress.com/english/summarization.html

Paraphrasing

Paraphrasing is expressing the meaning of an input sequence in alternative ways while maintaining grammatical, syntactical correctness.

1) Paraphrase identification - detecting if a pair of text inputs has the same meaning; classification task.

2) Paraphrase generation - producing paraphrases allows for the creation of more varied and fluent text; generation task

Build a model that reads a sequence of words and generates a different sequence with the same meaning

Paraphrasing

Why paraphrasing?

- style transfer:
  - translation from rude to polite
  - translation from professional to simple language
- data augmentation: increasing the number of examples for training ML-models
- increasing the stability of ML-models: training models on a wide variety of examples, in different styles, with different sentiment, but the same meaning / intent of the user

Paraphraser dataset
http://paraphraser.ru/

Tools for paraphrasing
https://github.com/RussianNLP/russian_paraphrasers

Simplification

Text Simplification (sentence simplification) is the task of reducing the complexity of the vocabulary and sentence structure of text while retaining its original meaning, with the goal of improving readability and understanding.

Sentence complexity criteria include:

- the presence of complex grammatical structures
- participial and adverbial expressions, subordinate sentences,
- the presence of rare, and ambiguous words,
- etc

Original Sentence	Simplified Sentence
Owls are the order Strigiformes, comprising 200 bird of prey species.	An owl is a bird. There are about 200 kinds of owls.
Owls hunt mostly small mammals, insects, and other birds though some species specialize in hunting fish.	Owls' prey may be birds, large insects (such as crickets), small reptiles (such as lizards) or small mammals (such as mice, rats, and rabbits).

Simplification

Metrics:

SARI (Xu et al., 2016) is a *lexical simplicity* metric that measures "how good" are the words added, deleted and kept by a simplification model. The metric compares the model's output to *multiple simplification references* and the original sentence.

Datasets:

- based on Wikipedia - **WikiLarge, WikiSmall**
  Simple English Wikipedia is an online encyclopedia aimed at English learners. Its articles are expected to contain fewer words and simpler grammar structures than those in their *Main English Wikipedia* counterpart. Much of the popularity of using Wikipedia for research in Simplification comes from publicly available sentence alignments between "equivalent" articles in Main and Simple English Wikipedia.
- **Turk Corpus**
- **ASSET**
- **Newsela**

Simplification

Generation aproaches - bart, gpts

RuGPT XL:

Origin: Аспирин — это лечение первой линии для лихорадки и симптомов суставной боли при остром ревматизме.
Generated: Аспирин обычно используется для лечения лихорадки, боли в суставах и мышечных судорог.

Origin: Боевые действия проходили на фоне разыгравшейся в тылу масштабной эпидемии чумы, унёсшей большое количество жизней.
Generated: В ходе боевых действий эпидемия чумы унесла много жизней.

State-of-the-art:

Model	BLEU	SARI
MUSS (Martin et al., 2020)	78.17	42.53
ACCESS (Martin et al., 2019)	72.53	41.87
DMASS + DCSS (Zhao et al., 2018)		40.45

http://nlpprogress.com/english/simplification.html

Questions?

https://github.com/king-menin/mipt-nlp2022