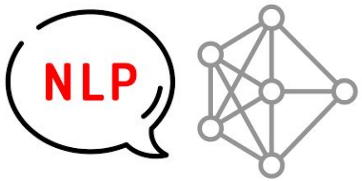
CHULA ENGINEERING
Foundation toward Innovation
COMPUTER



Neural Machine Translation (NMT)

2110572: Natural Language Processing Systems

Assoc. Prof. Peerapon Vateekul, Ph.D.
Department of Computer Engineering,
Faculty of Engineering, Chulalongkorn University
peerapon.v@chula.ac.th

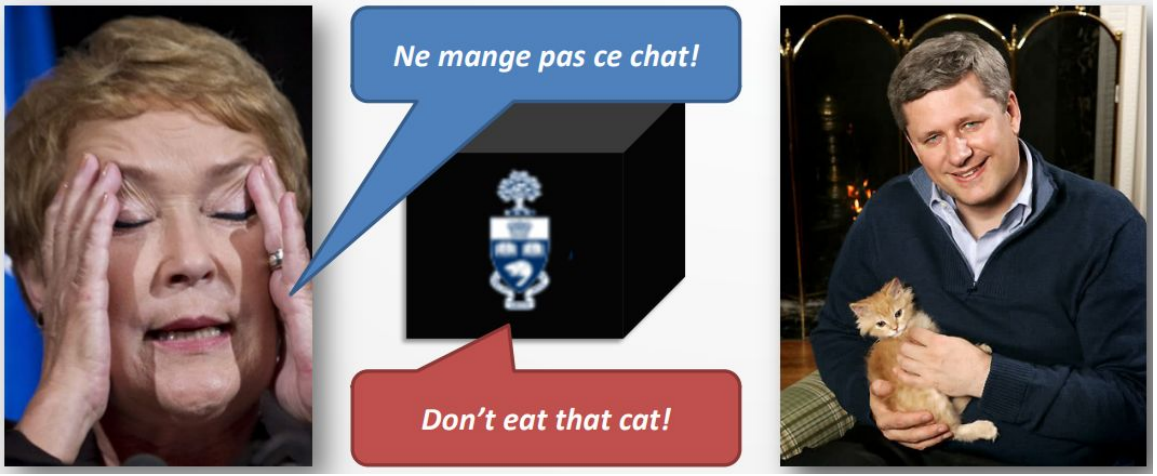
Credit: TA.Knight, TA.Pluem, and all TAs

Outline

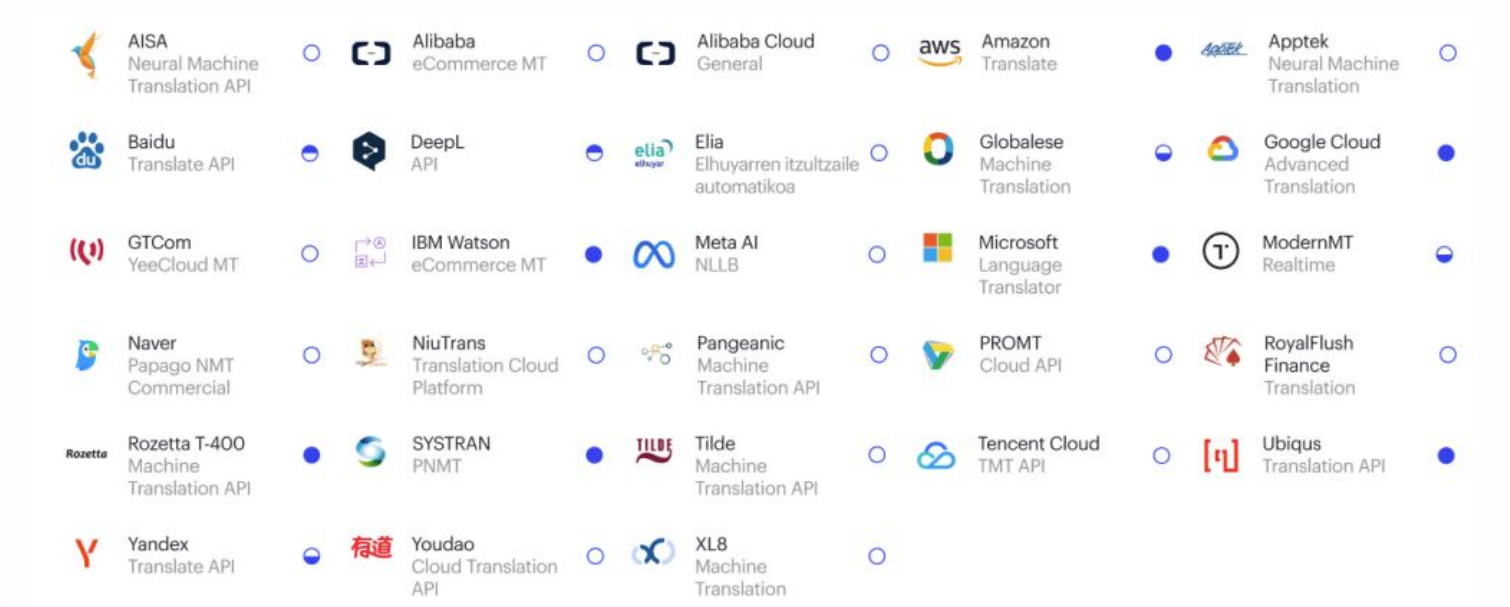
- Part1) MT models
  - mBART (2020)
  - NLLB (started 2018)
    - m2m-100 (2020)
    - NLLB-200 (2022)
- Part2) MT data sets
  - WMT
  - OPUS
  - FLORES-200
- Part3) Evaluation
  - Accuracy-based score (BLEU, charF++)
  - Model-based score
    - COMET
    - Quality Estimation

Introduction

Machine Translation (MT) is a research field in NLP that aims to create a system that can translate text from one language to another.

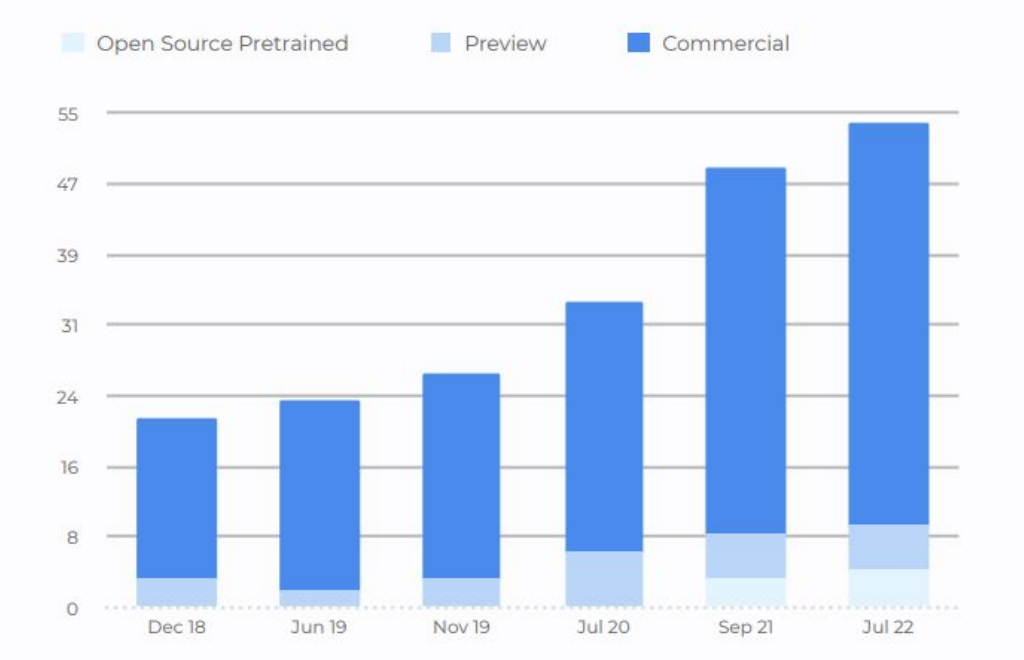


Machine Translation Landscape



2022 Savenkov and Lopez. The State of the Machine Translation 2022. ACL Proceedings of the 15th Biennial Conference of the Association for Machine Translation in the Americas (Volume 2: Users and Providers Track and Government Track) https://aclanthology.org/2022.amta-upg.4/

Cloud MT Vendors with Stock Models



Commercial (45)
AISA, Alibaba, Amazon, Apptek, Baidu, CloudTranslation, DeepL, Elia, Fujitsu, Globalese, Google, GTCom, IBM, iFlyTec, HiThink RoyalFlush, Lesan, Lindat, Lingvanex, Kawamura / NICT, Kingsoft, Masakhane, Microsoft, Mirai, ModernMT, Naver, Niutrans, NTT, Omniscien, Pangeanic, Prompsit, PROMT, Process9, Rozetta, RWS, SAP, Sogou, Systran, Tencent, Tilde, Ubiqus, Viscomtec, XL8, Yandex, YarakuZen, Youdao

Preview / Limited (5)
eBay, Kakao, QCRI, Tarjama, Birch.AI

Open Source Pretrained (3)
M2M-100, mBART, NLLB by Meta

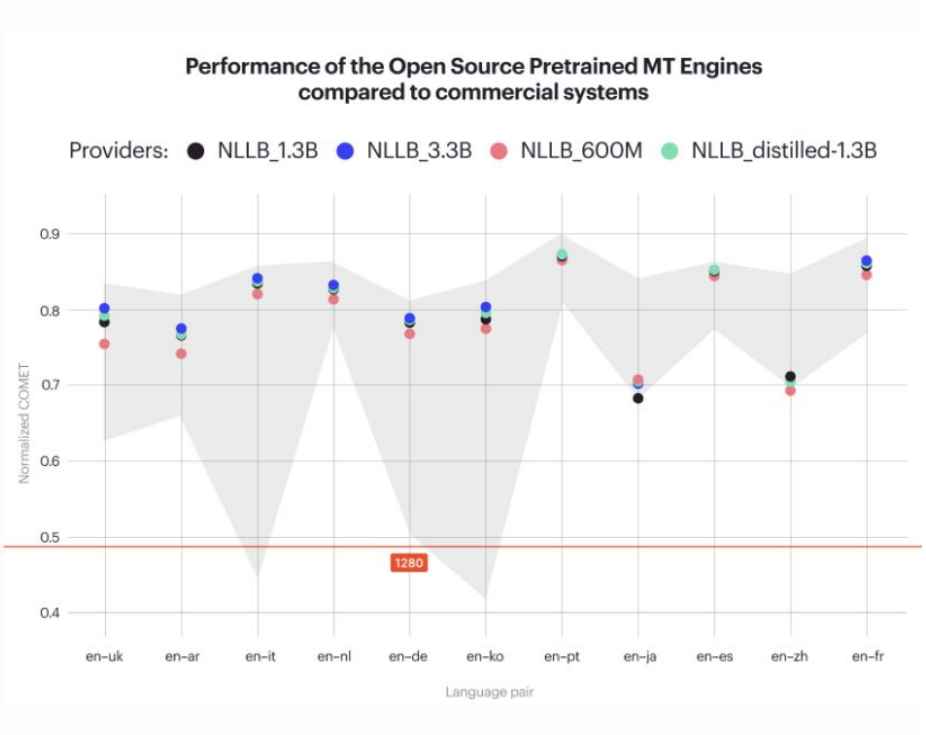
2022 Savenkov and Lopez. The State of the Machine Translation 2022. ACL Proceedings of the 15th Biennial Conference of the Association for Machine Translation in the Americas (Volume 2: Users and Providers Track and Government Track) https://aclanthology.org/2022.amta-upg.4/

Open Source MT Performance (COMET)

NLLB by Meta AI mostly show performance in the 2nd tier of commercial systems.

For **en-es (English-to-Spanish)**, NLLB scores are on par with the best commercial systems

For **en-zh (English-to-Chinese)** and **en-ja (English-to-Japanese)**, the scores are quite low



2022 Savenkov and Lopez. The State of the Machine Translation 2022. ACL Proceedings of the 15th Biennial Conference of the Association for Machine Translation in the Americas (Volume 2: Users and Providers Track and Government Track) https://aclanthology.org/2022.amta-upg.4/

+

Part1) MT models

- mBART (2020)
- NLLB (started 2018)
  - m2m-100 (2020)
  - NLLB-200 (2022)

History of Machine Translation

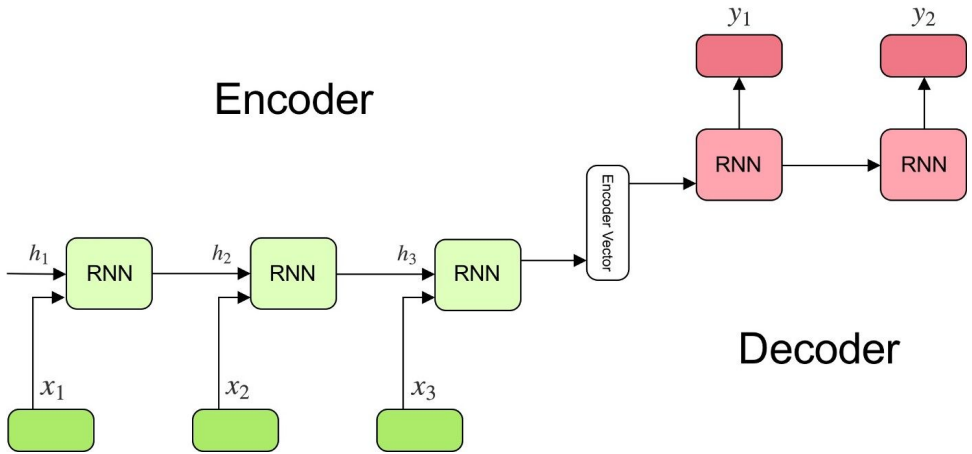
1. Rule-Based Machine Translation
   a. Manual, hand-crafted rules for vocabulary, grammar, etc.
   b. Low-quality translation and time-consuming.
   c. Cannot utilize context information!

2. Statistical Machine Translation
   a. Use statistics from a parallel corpus
   b. Google translate (from 2006 to 2016)

3. Neural Machine Translation
   a. Competitive performance but hard to debug
   b. Google translate (now)

Neural Machine Translation (NMT)

Prior to the Transformer, the dominant model in NMT was the RNN-based encoder-decoder.



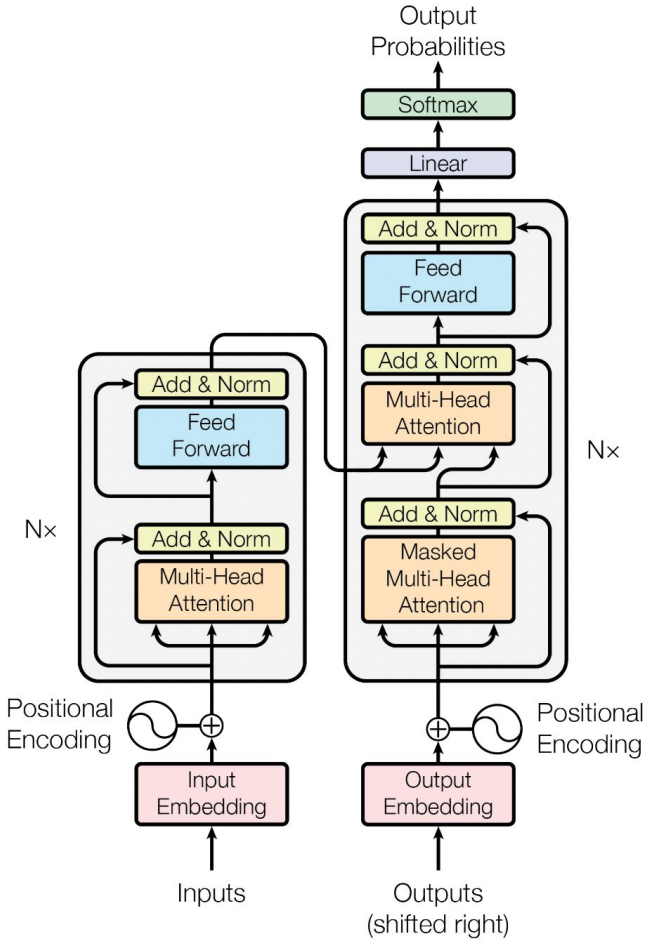
Transformer

The self-attention mechanism in the Transformer allows for better input representations and faster computation due to parallelization.

This brings big performance improvements and thus allows researchers to scale the model more efficiently.

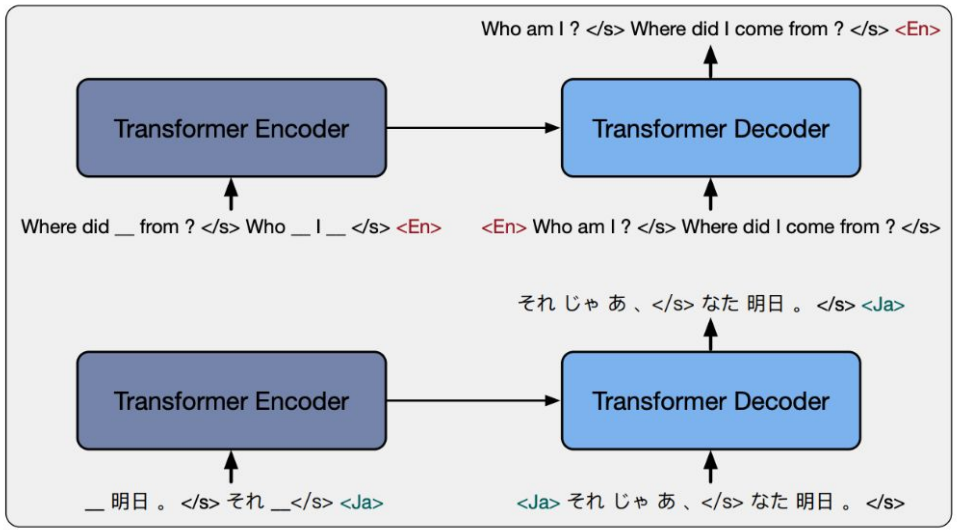
Table 2: The Transformer achieves better BLEU scores than previous state-of-the-art models on the English-to-German and English-to-French newstest2014 tests at a fraction of the training cost.

Model	BLEU		Training Cost (FLOPs)	
	EN-DE	EN-FR	EN-DE	EN-FR
ByteNet [18]	23.75			
Deep-Att + PosUnk [39]		39.2		$1.0 \cdot 10^{20}$
GNMT + RL [38]	24.6	39.92	$2.3 \cdot 10^{19}$	$1.4 \cdot 10^{20}$
ConvS2S [9]	25.16	40.46	$9.6 \cdot 10^{18}$	$1.5 \cdot 10^{20}$
MoE [32]	26.03	40.56	$2.0 \cdot 10^{19}$	$1.2 \cdot 10^{20}$
Deep-Att + PosUnk Ensemble [39]		40.4		$8.0 \cdot 10^{20}$
GNMT + RL Ensemble [38]	26.30	41.16	$1.8 \cdot 10^{20}$	$1.1 \cdot 10^{21}$
ConvS2S Ensemble [9]	26.36	**41.29**	$7.7 \cdot 10^{19}$	$1.2 \cdot 10^{21}$
Transformer (base model)	27.3	38.1	**$3.3 \cdot 10^{18}$**	
Transformer (big)	**28.4**	**41.8**	$2.3 \cdot 10^{19}$	



1) mBART

A standard encoder-decoder transformer model that pretrains on an input denoising objective. There are two steps: (1) pretrain on the denoising task and (2) finetune on the MT task.



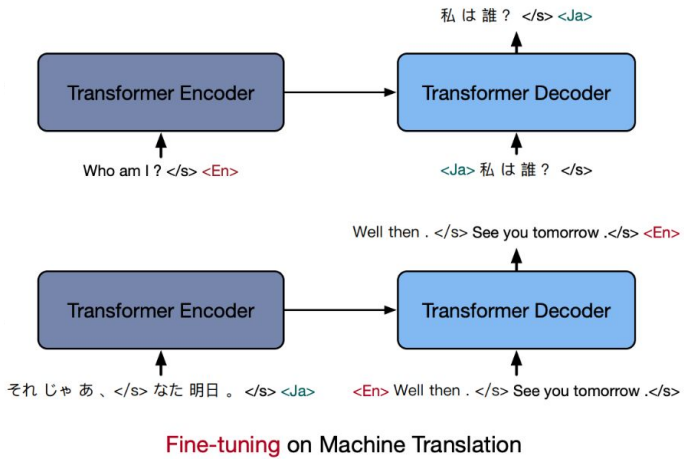
Multilingual Denoising Pre-Training (mBART)

Multilingual Denoising Pre-training for Neural Machine Translation. Liu et al 2020

improves

1) mBART (cont.)

Pretraining on multilingual data (~1TB) improve translation quality, especially for low-resource language pairs.



Languages	En-Gu		En-Kk		En-Vi		En-Tr		En-Ja		En-Ko	
Data Source	WMT19		WMT19		IWSLT15		WMT17		IWSLT17		IWSLT17	
Size	10K		91K		133K		207K		223K		230K	
Direction	←	→	←	→	←	→	←	→	←	→	←	→
Random	0.0	0.0	0.8	0.2	23.6	24.8	12.2	9.5	10.4	12.3	15.3	16.3
mBART25	**0.3**	**0.1**	**7.4**	**2.5**	**36.1**	**35.4**	**22.5**	**17.8**	**19.1**	**19.4**	**24.6**	**22.6**
Languages	En-Nl		En-Ar		En-It		En-My		En-Ne		En-Ro	
Data Source	IWSLT17		IWSLT17		IWSLT17		WAT19		FLoRes		WMT16	
Size	237K		250K		250K		259K		564K		608K	
Direction	←	→	←	→	←	→	←	→	←	→	←	→
Random	34.6	29.3	27.5	16.9	31.7	28.0	23.3	34.9	7.6	4.3	34.0	34.3
mBART25	**43.3**	**34.8**	**37.6**	**21.6**	**39.8**	**34.0**	**28.3**	**36.9**	**14.5**	**7.4**	**37.8**	**37.7**
Languages	En-Si		En-Hi		En-Et		En-Lt		En-Fi		En-Lv	
Data Source	FLoRes		ITTB		WMT18		WMT19		WMT17		WMT17	
Size	647K		1.56M		1.94M		2.11M		2.66M		4.50M	
Direction	←	→	←	→	←	→	←	→	←	→	←	→
Random	7.2	1.2	10.9	14.2	22.6	17.9	18.1	12.1	21.8	20.2	15.6	12.9
mBART25	**13.7**	**3.3**	**23.5**	**20.8**	**27.8**	**21.4**	**22.4**	**15.3**	**28.5**	**22.4**	**19.3**	**15.9**

Table 2: **Low/Medium Resource Machine Translation** Pre-training consistently improves over a randomly initialized baseline, with particularly large gains on low resource language pairs (e.g. Vi-En).

Multilingual Denoising Pre-training for Neural Machine Translation. Liu et al 2020

1) mBART (cont.)

It can also do zero-shot MT (with some tricks) that yields decent results.

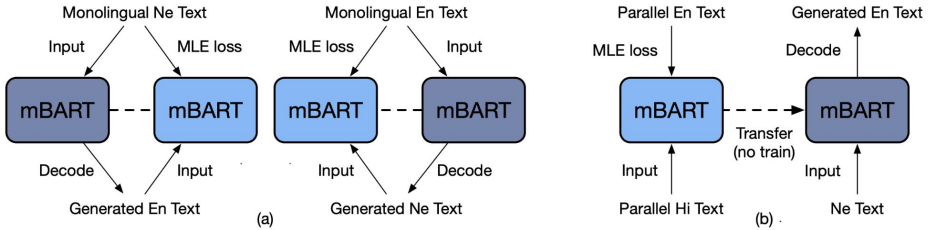


Figure 5: Illustrated frameworks for unsupervised machine translation via (a) back-translation (b) language transfer where Ne-En is used as an example. For both cases, we initialize from multilingual pre-training (e.g. mBART25).

Model	Similar Pairs				Dissimilar Pairs			
	En-De		En-Ro		En-Ne		En-Si	
	←	→	←	→	←	→	←	→
Random	21.0	17.2	19.4	21.2	0.0	0.0	0.0	0.0
XLM (2019)	34.3	26.4	31.8	33.3	0.5	0.1	0.1	0.1
MASS (2019)	**35.2**	28.3	**33.1**	**35.2**	-	-	-	-
mBART	34.0	**29.8**	30.5	35.0	**10.0**	**4.4**	**8.2**	**3.9**

Table 10: **Unsupervised MT via Back-Translation**. En-De, En-Ro are initialized by mBART02, while En-Ne, En-Si are initialized by mBART25. Our models are trained on monolingual data used in pre-training.

Multilingual Denoising Pre-training for Neural Machine Translation. Liu et al 2020

1) mBART (cont.)

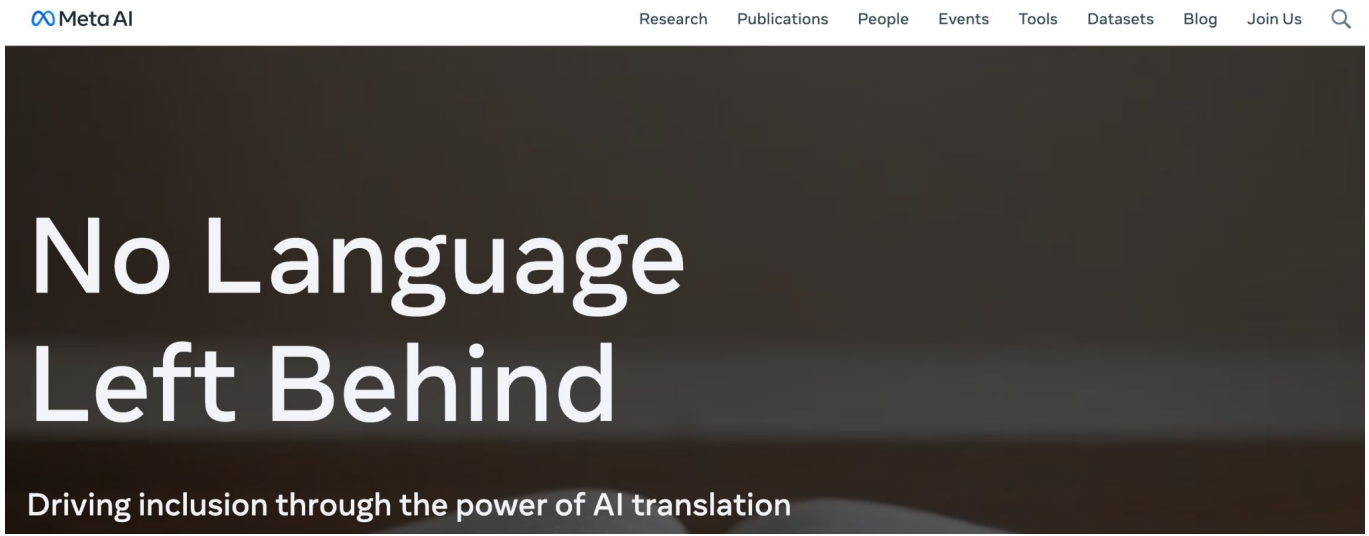
- Originally trained on 25 languages, 25 more languages (50 total languages) were added via continual pretraining.
- Thai language included! The model is very large, though.

Data size	Languages
10M+	German, Czech, French, Japanese, Spanish, Russian, Polish, Chinese
1M - 10M	Finnish, Latvian, Lithuanian, Hindi, Estonian
100k to 1M	Tamil, Romanian, Pashto, Sinhala, Malayalam, Dutch, Nepali, Italian, Arabic, Korean, Hebrew, Turkish, Khmer, Farsi, Vietnamese, Croatian, Ukrainian
10K to 100K	Thai, Indonesian, Swedish, Portuguese, Xhosa, Afrikaans, Kazakh, Urdu, Macedonian, Telugu, Slovenian, Burmese, Georgia
10K-	Marathi, Gujarati, Mongolian, Azerbaijani, Bengali

Multilingual Translation with Extensible Multilingual Pretraining and Finetuning. Tang et al 2020

2) No Language Left Behind (NLLB) [FB started in 2018]

From the webpage - “No Language Left Behind (NLLB) is a first-of-its-kind, AI breakthrough project that open-sources models capable of delivering evaluated, high-quality translations **directly between 200 languages**”...



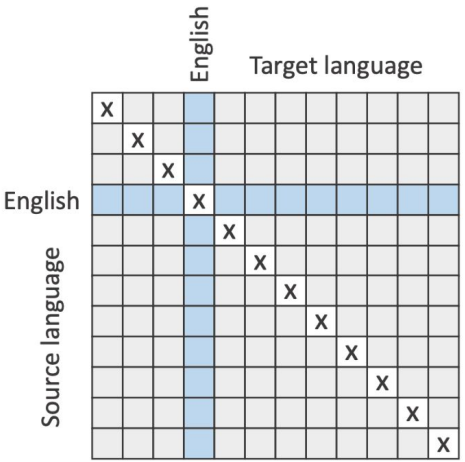
https://ai.facebook.com/research/no-language-left-behind

No Language Left Behind

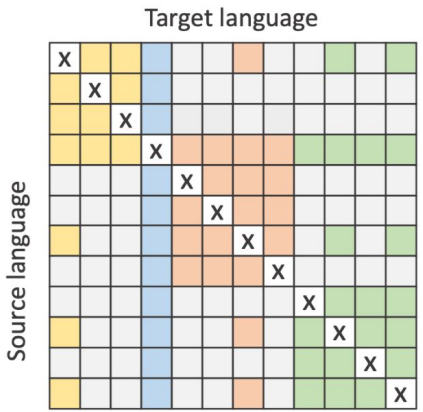
2.1) m2m-100

Previous works mostly focus on translation from/to English (English-centric).

This paper introduces a many-to-many translation model and dataset of 100 languages (that's 9900 directions!)



(a) English-Centric Multilingual



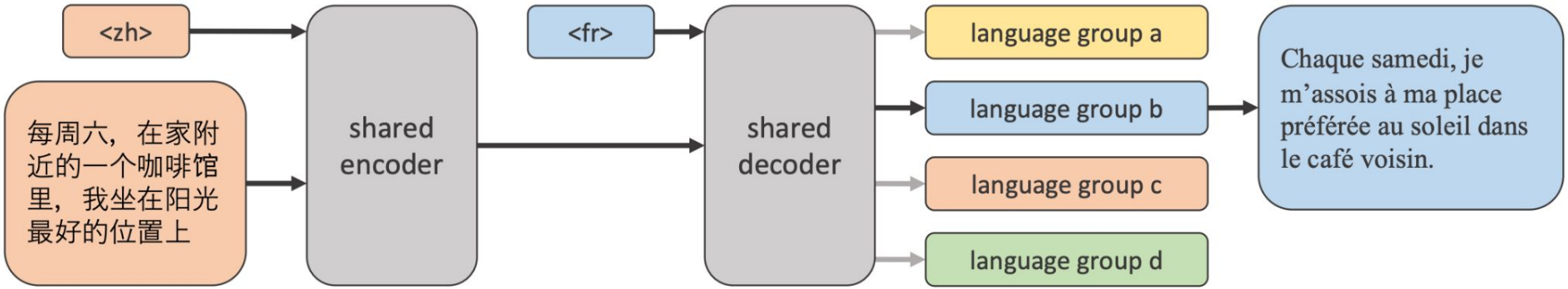
(b) M2M-100: Many-to-Many Multilingual Model

Beyond English-Centric Multilingual Machine Translation. Fan et al 2020

No Language Left Behind

2.1) m2m-100 (cont.)

The model is also an encoder-decoder model with additional language-specific (language token) sparse models.



(c) Translating from Chinese to French with Dense + Language-Specific Sparse Model

Beyond English-Centric Multilingual Machine Translation. Fan et al 2020

No Language Left Behind

2.1) m2m-100 (cont.)

The model outperforms mBART even though it trains on 100 languages. Thai language is also covered by this model.

Benchmark	Model	BLEU
mBART	Previous Work (Liu et al., 2020)	23.9
	M2M-100	**24.6**
CCMatrix	Previous Work (Schwenk et al., 2019)	16.3
	M2M-100	**18.7**
OPUS100	Previous Work (Zhang et al., 2020)	14.1
	M2M-100	**18.4**

Table 11: **Comparison on various evaluation settings from previous work**. We display the best performing model from the published work and report average BLEU on the test set. For these comparisons, we use the tokenization and BLEU evaluation script used by each work for comparability. Liu et al. (2020) report Low/Mid resource directions into and out of English and High resource directions into English, we average across all. Schwenk et al. (2019) report the full matrix on 28 languages, we average across all. Zhang et al. (2020) report results on non-English directions, we average across all.

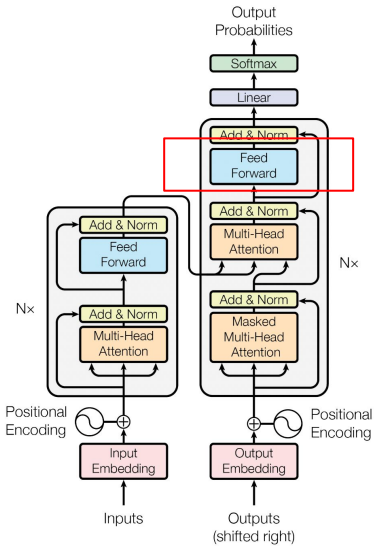
Beyond English-Centric Multilingual Machine Translation. Fan et al 2020

No Language Left Behind

2.2) NLLB-200

This model is capable of a total of 40,602 translation directions!

It is also an encoder-decoder transformer model. However, it is a sparse model (Mixture of Experts transformer).



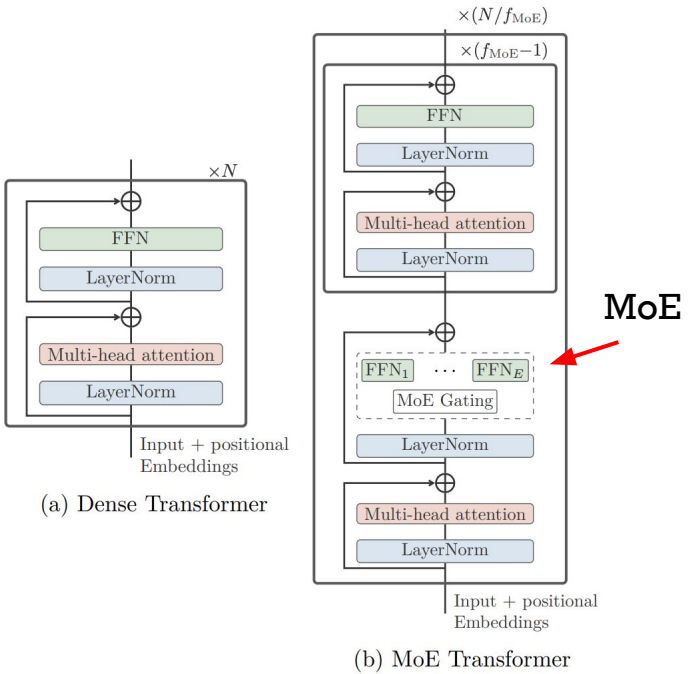


Figure 16: **Illustration of a Transformer encoder with MoE layers** inserted at a $1{:}f_{\text{MoE}}$ frequency. Each MoE layer has E experts and a gating network responsible for dispatching tokens.

No Language Left Behind: Scaling Human-Centered Machine Translation. NLLB Team 2022

No Language Left Behind

2.2) NLLB-200 (cont.)

With just 1.3B parameters, it outperforms Google Translate in some language pairs.

(Distilled) weight available!
https://huggingface.co/facebook/nllb-200-distilled-600M

	eng_Latn-xx	xx-eng_Latn	xx-yy	Avg.
87 languages				
M2M-100	-/-	-/-	-/-	13.6/-
Deepnet	-/-	-/-	-/-	18.6/-
NLLB-200	**35.4**/52.1	**42.4**/62.1	**25.2**/43.2	**25.5**/43.5
101 languages				
DeltaLM	26.6/-	33.2/-	16.4/-	16.7/-
NLLB-200	**34.0**/50.6	**41.2**/60.9	**23.7**/41.4	**24.0**/41.7

Table 30: **Comparison on FLORES-101 devtest**. We evaluate over full FLORES-101 10k directions. We report both spBLEU/chrF++ where available. All spBLEU numbers are computed with FLORES-101 SPM tokenizer. Scores for DeltaLM are taken from FLORES-101 leaderboard. M2M-100 and Deepnet average is only over 87 languages that overlap with FLORES-101, we also show NLLB-200 performance on that subset of languages. NLLB-200 outperforms previous state of the art models by a significant margin, even after supporting twice as many languages.

	eng_Latn-xx		xx-eng_Latn		Average	
	low	v.low	low	v.low	low	v.low
Google Translate	**32.3/50.3**	**27.0/46.5**	35.9/57.1	35.8/57.0	34.1/53.7	31.3/51.7
NLLB-200	30.3/48.2	25.7/45.0	**41.3/60.4**	**41.1/60.3**	**35.8/54.3**	**33.4/52.6**

Table 34: **Comparison on 102 Low-Resource Directions on FLORES-200 devtest against commercial translation systems.** We evaluate on all English-centric low-resource directions that overlap between FLORES-200 and Google's Translation API as of this writing. We report both spBLEU/chrF++ and bold the best score. We observe that NLLB-200 outperforms significantly on xx-eng_Latn and overall average.

No Language Left Behind

2.2) NLLB-200 (cont.)

This paper also explains how they created a parallel corpus (NLLB-SEED), including a multilingual sentence encoder and a language identification model.

The whole paper is 192 pages!

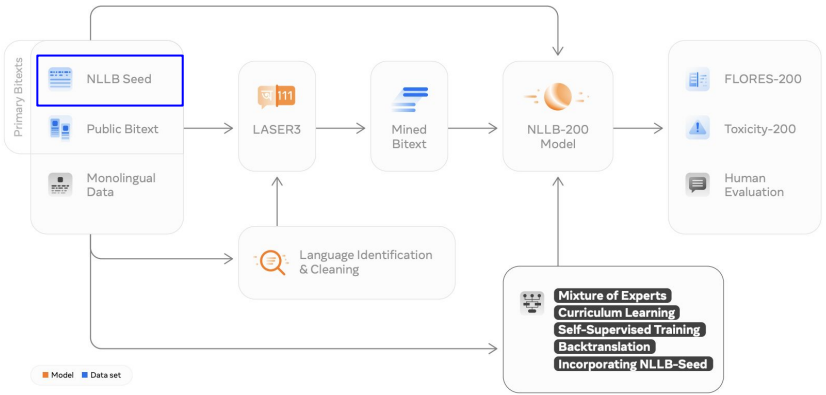


Figure 14: **Modeling Contributions of No Language Left Behind**: As highlighted, we describe several modeling techniques to enable coverage of hundreds of languages in one model. We focus on effectively scaling model capacity while mitigating overfitting, as well as how to improve backtranslation for low-resource languages and incorporate NLLB-SEED.

No Language Left Behind: Scaling Human-Centered Machine Translation. NLLB Team 2022

MT on LLM Performance

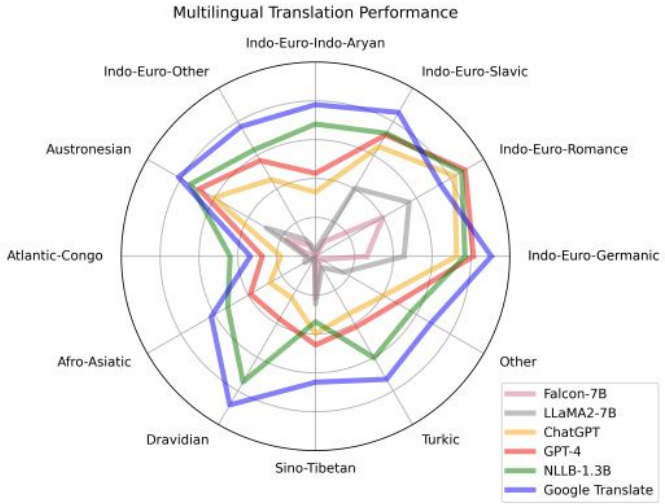


Figure 1: Multilingual translation performance (BLEU) of some popular LLMs and traditional supervised systems in translating from English to non-English. LLMs have demonstrated great potential in multilingual machine translation.

LLM capabilities are evolving

GPT-4 beat supervised baseline **NLLB** in 41% of translation pairs but still has large gap to **Google Translate**

But the advantage is LLM can acquire moderate translation ability in **zero-resource languages.**

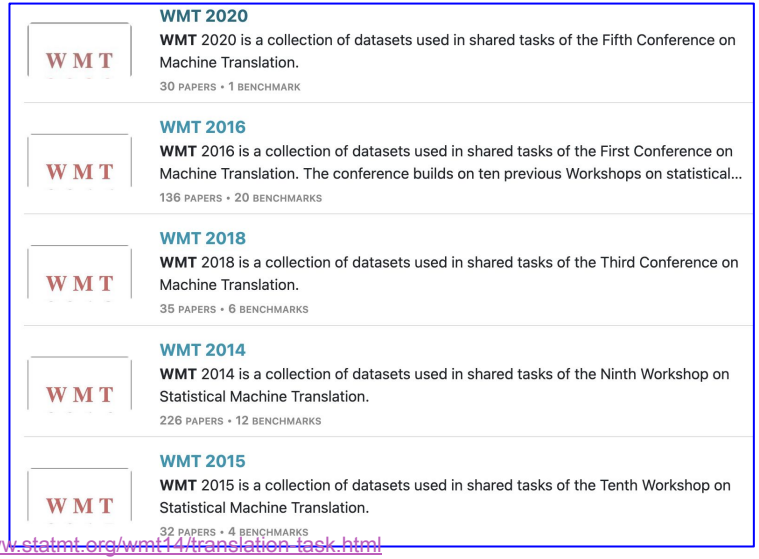
2024 Zhu et. al. Multilingual Machine Translation with Large Language Models: Empirical Results and Analysis. ACL Findings of the Association for Computational Linguistics: NAACL 2024 https://aclanthology.org/2024.findings-naacl.176/

Part2) Notable datasets

WMT
OPUS
FLORES-200

1) WMT

- WMT (Workshop on Statistical Machine Translation)—This is a machine translation dataset composed of a collection of various sources, including (1) news commentaries and (2) parliament proceedings.
- It focuses mainly on European language pairs.
- There are many versions from 2008 to 2014.



WMT 2020
WMT 2020 is a collection of datasets used in shared tasks of the Fifth Conference on Machine Translation.
30 PAPERS • 1 BENCHMARK

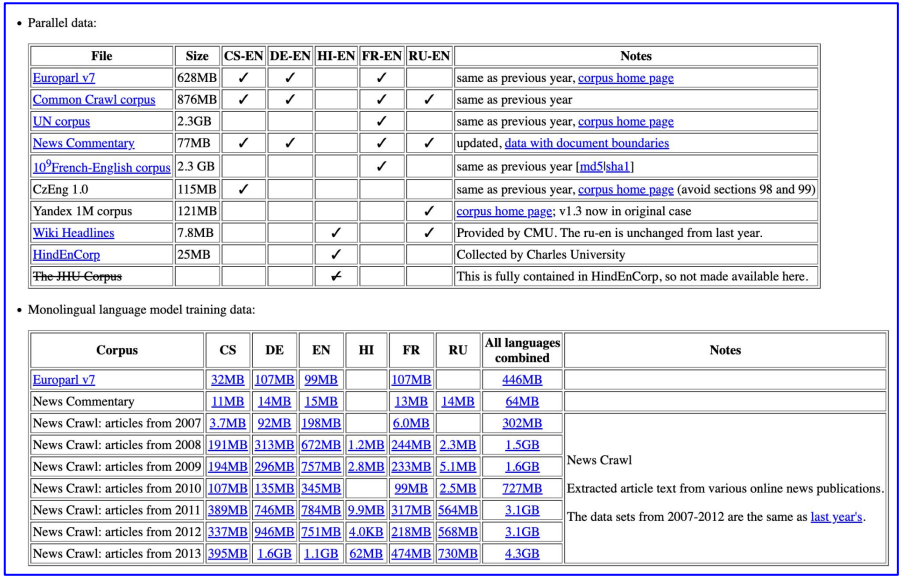
WMT 2016
WMT 2016 is a collection of datasets used in shared tasks of the First Conference on Machine Translation. The conference builds on ten previous Workshops on statistical...
136 PAPERS • 20 BENCHMARKS

WMT 2018
WMT 2018 is a collection of datasets used in shared tasks of the Third Conference on Machine Translation.
35 PAPERS • 6 BENCHMARKS

WMT 2014
WMT 2014 is a collection of datasets used in shared tasks of the Ninth Workshop on Statistical Machine Translation.
226 PAPERS • 12 BENCHMARKS

WMT 2015
WMT 2015 is a collection of datasets used in shared tasks of the Tenth Workshop on Statistical Machine Translation.
32 PAPERS • 4 BENCHMARKS

https://www.statmt.org/wmt14/translation-task.html



- Parallel data:

File	Size	CS-EN	DE-EN	HI-EN	FR-EN	RU-EN	Notes
Europarl v7	628MB	✓	✓		✓		same as previous year, corpus home page
Common Crawl corpus	876MB	✓	✓		✓	✓	same as previous year
UN corpus	2.3GB				✓		same as previous year, corpus home page
News Commentary	77MB	✓	✓		✓	✓	updated, data with document boundaries
10^9French-English corpus	2.3 GB				✓		same as previous year [md5|sha1]
CzEng 1.0	115MB	✓					same as previous year, corpus home page (avoid sections 98 and 99)
Yandex 1M corpus	121MB					✓	corpus home page; v1.3 now in original case
Wiki Headlines	7.8MB			✓		✓	Provided by CMU. The ru-en is unchanged from last year.
HindEnCorp	25MB			✓			Collected by Charles University
~~The JHU Corpus~~				✓			This is fully contained in HindEnCorp, so not made available here.

- Monolingual language model training data:

Corpus	CS	DE	EN	HI	FR	RU	All languages combined	Notes
Europarl v7	32MB	107MB	99MB		107MB		446MB	
News Commentary	11MB	14MB	15MB		13MB	14MB	64MB	
News Crawl: articles from 2007	3.7MB	92MB	198MB		6.0MB		302MB	News Crawl Extracted article text from various online news publications. The data sets from 2007-2012 are the same as last year's.
News Crawl: articles from 2008	191MB	313MB	672MB	1.2MB	244MB	2.3MB	1.5GB	
News Crawl: articles from 2009	194MB	296MB	757MB	2.8MB	233MB	5.1MB	1.6GB	
News Crawl: articles from 2010	107MB	135MB	345MB		99MB	2.5MB	727MB	
News Crawl: articles from 2011	389MB	746MB	784MB	9.9MB	317MB	564MB	3.1GB	
News Crawl: articles from 2012	337MB	946MB	751MB	4.0KB	218MB	568MB	3.1GB	
News Crawl: articles from 2013	395MB	1.6GB	1.1GB	62MB	474MB	730MB	4.3GB	

2) OPUS

Opus - https://opus.nlpl.eu/

... the open parallel corpus

OPUS is a growing collection of translated texts from the web. In the OPUS project we try to convert and align free online data, to add linguistic annotation, and to provide the community with a publicly available parallel corpus. OPUS is based on open source products and the corpus is also delivered as an open content package. We used several tools to compile the current collection. All pre-processing is done automatically. No manual corrections have been carried out.

The OPUS collection is growing! Check this page from time to time to see new data arriving ...
Contributions are very welcome! Please contact <jorg.tiedemann@helsinki.fi >

Search & download resources: -- select -- -- select -- all show all versions

No Language Left Behind

3) FLORES-200

- FLORES-200 (Few-Shot Learning Oriented Evaluation for Machine Translation) is a large-scale multilingual dataset designed to evaluate machine translation (MT) models across 200 languages. It was developed by Meta AI (formerly Facebook AI) to benchmark multilingual models, such as mBART-50, NLLB (No Language Left Behind), and M2M-100.
- FLORES-200 consists of translations from 842 distinct web articles, totaling 3,001 sentences. These sentences are divided into three splits: dev, devtest, and test (hidden). On average, sentences are approximately 21 words long.

https://github.com/facebookresearch/flores/blob/main/flores200/README.md

Part3) Evaluating a Translation

Syntactic Similarity (BLEU, ROUGE, METEOR, TER, chrF)
Semantic Similarity (BERTScore, COMET, BLEURT)
Human Judgement (HTER, DA, MQM)

Evaluation

Syntactic Similarity

- BLEU
- ROUGE
- METEOR
- TER
- ChrF

Semantic Similarity

- BERTScore
- COMET
- BLEURT

Human Judgement

- HTER
- DA
- MQM

Syntactic Similarity

$$RECALL = \frac{Overlapping\ number\ of\ n-grams}{Number\ of\ n-grams\ in\ the\ reference}$$

$$PRECISION = \frac{Overlapping\ number\ of\ n-grams}{Number\ of\ n-grams\ in\ the\ candidate}$$

BLEU

BLEU is a precision focused metric that calculates n-gram overlap of the reference and generated texts (including brevity penalty–penalizing short sentences). Works well for structured content but struggles with paraphrasing.

$$BLEU = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right)$$

ROUGE

Very similar to the BLEU definition, the difference being that Rouge is recall focused overlap. Often used for summarization but can apply to MT. *ROUGE-N* considers n-gram overlap. *ROUGE-L* considers longest common subsequence (LCS).

$$\text{ROUGE-N} = \frac{\sum_{S \in \{ReferenceSummaries\}} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in \{ReferenceSummaries\}} \sum_{gram_n \in S} Count(gram_n)} \quad (1)$$

METEOR

Improves over BLEU by considering synonyms, stemming, and paraphrasing (WordNet). The metric is a harmonic mean of unigram precision and recall (recall weighted 9x higher than precision). Penalty is changed to be correlated to number of adjacent chunks.

$$F_{mean} = \frac{10PR}{R + 9P} \qquad p = 0.5\left(\frac{c}{u_m}\right)^3$$

$$M = F_{mean}(1 - p)$$

TER

Translation Edit Rate: Measures the number of edits (insertions, deletions, shifts, and substitutions) required to transform a machine translation into the reference translation. Penalizes paraphrases/synonyms. Penalizes translations of different length.

$$TER = \frac{number-of-edits}{word-length-of-Reference-text}$$

ChrF

An F-score based on character n-gram precision and recall (instead of word or n-gram level). No need for tokenization. Better for highly inflected languages (e.g., Finnish, Turkish, Thai, Arabic) where words change forms frequently.

$$\text{CHRF}\beta = (1 + \beta^2)\frac{\text{CHRP} \cdot \text{CHRR}}{\beta^2 \cdot \text{CHRP} + \text{CHRR}}$$

Very popular metrics

BLEU (Bilingual Evaluation Understudy) [2002]

The most popular metric to (try to) measure the quality of predicted translations.

The idea is to measure the similarity of the predictions with human references.

> *The BLEU metric ranges from 0 to 1. Few translations will attain a score of 1 unless they are identical to a reference translation. For this reason, even a human translator will not necessarily score 1. […] on a test corpus of about 500 sentences (40 general news stories), a human translator scored 0.3468 against four references and scored 0.2571 against two references.*

— BLEU: a Method for Automatic Evaluation of Machine Translation, 2002.

BLEU (cont.)

The score is calculated based on **clipped n-gram precision.**

Clipping prevents repetitive predicted sentence from getting a good score

For example, we could have

- **Target Sentence:** He eats an apple
- **Predicted Sentence:** He He He

This means that the 1-gram precision is 3/3 or 100% (it shouldn't be like this!)

https://towardsdatascience.com/foundations-of-nlp-explained-bleu-score-and-wer-metrics-1a5ba06d812b

BLEU (cont.)

The score is calculated based on **clipped n-gram precision.**

So we limit the count of each word to the maximum number of times that the word occurs in the target Sentence

- **Target Sentence 1:** He eats a sweet apple
- **Target Sentence 2:** He is eating a tasty apple
- **Predicted Sentence:** He He He eats tasty fruit

"He" occurs max. one time in 2 reference sentences. So we clip it to 1.

Word	*Matching Sentence*	*Matched Predicted Count*	*Clipped Count*
He	*Both*	*3*	*1*
eats	*Target 1*	*1*	*1*
tasty	*Target 2*	*1*	*1*
fruit	*None*	*0*	*0*
Total		*5*	*3*

Now the 1-gram precision becomes 3/6
There are 6 words in the predicted sentence

Note that precision now refers to the clipped precision

https://towardsdatascience.com/foundations-of-nlp-explained-bleu-score-and-wer-metrics-1a5ba06d812b

BLEU (cont.)

The score is calculated based on **clipped n-gram precision.**

Now we calculate the n-gram (clipped) precision for all N. The widely used number for N is 4 and a uniform weight w_n of N/4.

Let's look at all the 2-grams in our predicted sentence:

Target Sentence: The guard arrived late because it was raining

Predicted Sentence: The guard arrived late because of the rain

$$\begin{aligned} Geometric\ Average\ Precision\ (N) &= exp(\sum_{n=1}^{N} w_n\ log\ p_n) \\ &= \prod_{n=1}^{N} p_n^{w_n} \\ &= (p_1)^{\frac{1}{4}} \cdot (p_2)^{\frac{1}{4}} \cdot (p_3)^{\frac{1}{4}} \cdot (p_4)^{\frac{1}{4}} \end{aligned}$$

The precision of 2-grams is 4/7. (#correct/#total)

[pre(1-gram)*pre(2-gram)*pre(3-gram)*pre(4-gram)]$^{1/4}$

https://towardsdatascience.com/foundations-of-nlp-explained-bleu-score-and-wer-metrics-1a5ba06d812b

BLEU (cont.)

The score is calculated based on **clipped n-gram precision.**

Next, we compute the brevity penalty. This **penalizes predicted sentences that are too short.**

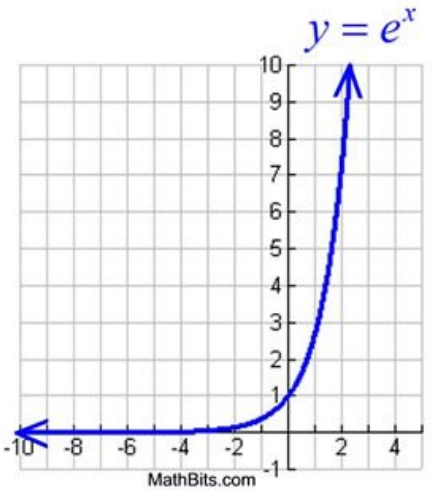
For example, a predicted sentence can contain only one word and get a perfect n-gram precision score

$$Brevity\ Penalty = \begin{cases} 1, & \text{if } c > r \\ e^{(1-r/c)}, & \text{if } c <= r \end{cases}$$

[0,1]

1

[0,1]



- c is *predicted length = number of words in the predicted sentence* and
- r is *target length = number of words in the target sentence*

https://towardsdatascience.com/foundations-of-nlp-explained-bleu-score-and-wer-metrics-1a5ba06d812b

BLEU (cont.)

The score is calculated based on **clipped n-gram precision.**

Finally, we can compute the BLEU score by

$$Bleu\ (N) = Brevity\ Penalty \cdot Geometric\ Average\ Precision\ Scores\ (N)$$

2
(larger is better)

1
(larger is better)

https://towardsdatascience.com/foundations-of-nlp-explained-bleu-score-and-wer-metrics-1a5ba06d812b

BLEU (cont.)

The score is calculated based on **clipped n-gram precision.**

Finally, we can compute the BLEU score by

$$Bleu\ (N) = Brevity\ Penalty \cdot Geometric\ Average\ Precision\ Scores\ (N)$$

Even though BLEU score is widely used, it has some important weaknesses:

- It does not consider words that have the **same meaning** to be correct. For example, for the word "dog," we can either use "หมา" or "สุนัข".
- It ignores the **importance of words**. With Bleu Score an incorrect word like "to" or "an" that is less relevant to the sentence is penalized just as heavily as a word that contributes significantly to the meaning of the sentence.
- Most importantly, **higher BLEU does not always mean a good score based on human judgment [1].**

https://towardsdatascience.com/foundations-of-nlp-explained-bleu-score-and-wer-metrics-1a5ba06d812b

[1] https://aclanthology.org/J18-3002.pdf

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) [2004]

Introduced in 2004 Paper: Lin. ROUGE: a Package for Automatic Evaluation of Summaries.

https://aclanthology.org/W04-1013/

Instead of precision measures like BLEU, ROUGE is more focused on recall.

- Mostly used for summarization
- 4 Versions of ROUGE:
  - **ROUGE-N**: Measures the *n-gram overlap* between the generated text and reference text.
  - **ROUGE-L**: Takes the *longest common subsequences (LCS)*, useful for capturing structural similarity.
  - **ROUGE-W**: Weighs *contiguous matches* that are higher than other n-grams.
  - **ROUGE-S**: Measures *skip-bigram overlap*, where two words are considered but may not be adjacent.

*Note: **ROUGE-1, ROUGE-2,** and **ROUGE-L** is mostly used*

https://www.geeksforgeeks.org/understanding-bleu-and-rouge-score-for-nlp-evaluation/
https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4753220 This paper says which ROUGE is mostly used in literature.

ROUGE-N

ROUGE-N is an n-gram **recall** between a candidate summary and a set of references.

N indicates the *number of N grams* which can be 1 (unigram) and 2 (bigram).

In the paper, ROUGE-N is only defined for **recall**.
In practice, the Python implementation shows **ROUGE-N F1**.

As defined in paper

$$\text{ROUGE-N} = \frac{\sum_{S \in \{ReferemceSummaries\}} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in \{ReferenceSummaries\}} \sum_{gram_n \in S} Count(gram_n)}$$

ROUGE-1

```
Candidate 1 : Summarization is cool
Reference 1 : Summarization is beneficial and cool
Reference 2 : Summarization saves time
```

Use only reference 1 since there are more overlapping words

Recall = 3/5 = 0.6
Precision = 3/3 = 1

ROUGE-1 F1= 2RP/(R+P)= 2*0.6/1.6 = **0.75**

$$RECALL = \frac{Overlapping\ number\ of\ n-grams}{Number\ of\ n-grams\ in\ the\ reference}$$

$$PRECISION = \frac{Overlapping\ number\ of\ n-grams}{Number\ of\ n-grams\ in\ the\ candidate}$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

https://medium.com/@eren9677/text-summarization-387836c9e178

ROUGE-N

ROUGE-2

It's the same process as above, but right now bigrams are determined.

```
Candidate 1: (Summarization is),(is cool)
Reference 1: (Summarization is),(is beneficial),(beneficial and),(and cool)
Reference 2: (Summarization saves),(saves time)
```

Recall = 1/4 = 0.25
Precision = 1/2 = 0.5

ROUGE-2 F1= 2RP/(R+P)= 2*0.25*0.5/0.75 = **0.33**

Huggingface Evaluate uses `rouge-score` package (replication of original perl package)
https://pypi.org/project/rouge-score/

```
candidates = ["Summarization is cool"]
references = [["Summarization is beneficial and cool","Summarization saves time"]]

results = rouge.compute(predictions=candidates, references=references)
print(results)
```

```
{'rouge1': 0.7499999999999999, 'rouge2': 0.3333333333333333, 'rougeL': 0.7499999999999999, 'rougeLsum': 0.7499999999999999}
```

https://medium.com/@eren9677/text-summarization-387836c9e178

ROUGE-L

ROUGE-L is **Longest Common Subsequence (LCS)** oriented. LCS is the longest sequence of words that appear in both the candidates and reference summaries, while keeping the order of the words intact.

Note that LCSes are not necessarily consecutive but still in order

```
Candidate: A fast brown fox leaps over a sleeping dog.
Reference: The quick brown fox jumps over the lazy dog.
```

Recall = 4/9 = 0.44
Precision = 4/9 = 0.44

ROUGE-L = 2RP/(R+P)= 2*0.44*0.44/0.89 = **0.44**

```
[6] candidates = ["A fast brown fox leaps over a sleeping dog"]
    references = [["The quick brown fox jumps over the lazy dog"]]

    results = rouge.compute(predictions=candidates, references=references)
    print(results)

{'rouge1': 0.4444444444444444, 'rouge2': 0.125, 'rougeL': 0.4444444444444444, 'rougeLsum': 0.4444444444444444}
```

As defined in paper

$$R_{lcs} = \frac{LCS(X,Y)}{m}$$

$$P_{lcs} = \frac{LCS(X,Y)}{n}$$

$$F_{lcs} = \frac{(1+\beta^2)R_{lcs}P_{lcs}}{R_{lcs} + \beta^2 P_{lcs}}$$

The paper introduced β in F1-score.
β = 1 means F1.
β > 1, recall is given more weight than precision.

https://medium.com/@eren9677/text-summarization-387836c9e178

Other ROUGE Flavours

ROUGE-L Sum

- Previous ROUGE-L is at the sentence level. In summary level, newlines are interpreted as sentence boundaries, and the LCS is computed between each pair of reference and candidate sentences, and something called union-LCS is computed.
- ROUGE-LSum evaluates LCS across multiple sentences, unlike ROUGE-L, which is sentence-level.

Reference Summary (Ground Truth) [20 tokens]

"The President held a press conference today. He discussed the economy and upcoming policies. Several journalists asked about inflation concerns."

Generated Summary (Model Output) [19 tokens]

"The President spoke at a press conference. He mentioned new policies and economic conditions. Reporters questioned him about inflation."

LCS length = 7

Precision

$$\text{ROUGE} - \text{L Precision} = \frac{\text{LCS Length}}{\text{Generated Summary Length}}$$

$$= \frac{7}{19} = 0.368$$

Recall

$$\text{ROUGE} - \text{L Recall} = \frac{\text{LCS Length}}{\text{Reference Summary Length}}$$

$$= \frac{7}{20} = 0.35$$

F1 Score

$$\text{ROUGE} - \text{L F1} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

$$= \frac{2 \times 0.368 \times 0.35}{0.368 + 0.35} = \frac{0.257}{0.718} = 0.359$$

Other ROUGE Flavours (cont.)

ROUGE-W

ROUGE-L LCS does not differentiate consecutiveness. In the following example, Y1 and Y2 will have the same ROUGE-L score. However, Y1 should be a better match than Y1 because it has consecutive matches. ROUGE-W improves the ROUGE-L by assigning more weight to consecutive LCSes. More details in this link.

$$
\begin{aligned}
X: &\quad [\underline{A}\ \underline{B}\ \underline{C}\ \underline{D}\ E\ F\ G] \\
Y_1: &\quad [\underline{A}\ \underline{B}\ \underline{C}\ \underline{D}\ H\ I\ K] \\
Y_2: &\quad [\underline{A}\ H\ \underline{B}\ K\ \underline{C}\ I\ \underline{D}]
\end{aligned}
$$

ROUGE-S

ROUGE-S stands for skip-grams. The order of the words in each sequence is preserved, but arbitrary gaps are allowed between words.

```
Sentence   : police killed the gunman
Skip-Grams : (“police killed”, “police the”, “police gunman”,
              “killed the”,  “killed gunman”, “the gunman”)
```

Formula of ROUGE-S is almost same as ROUGE-L with an addition of combination function(C).

$$R_{skip2} = \frac{SKIP2(X,Y)}{C(m,2)}$$

$$P_{skip2} = \frac{SKIP2(X,Y)}{C(n,2)}$$

$$F_{skip2} = \frac{(1+\beta^2)R_{skip2}P_{skip2}}{R_{skip2}+\beta^2 P_{skip2}}$$

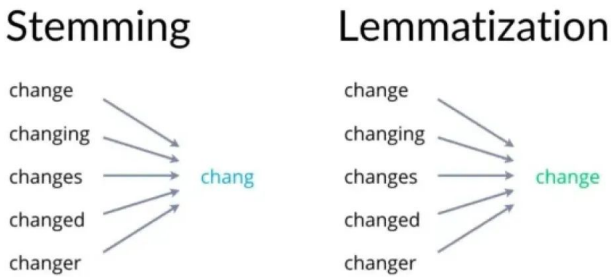
https://medium.com/@eren9677/text-summarization-387836c9e178

METEOR [2005]

Introduced in 2005 Paper: Banerjee and Lavie. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments https://aclanthology.org/W05-0909/

It's a metric *aimed to improve BLEU* for machine translation based on the harmonic mean of unigram precision and recall (with recall weighted higher than precision).

Several features that are not found in other metrics, such as **stemming** and **synonymy** matching, along with the standard **exact word** matching. (Uses WordNet)

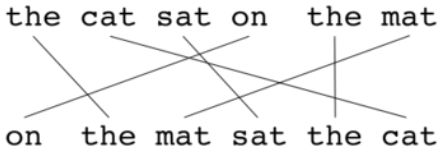


METEOR has a correlation of up to 0.964 with human judgment at the corpus level, compared to BLEU's achievement of 0.817 on the same data set.

https://spotintelligence.com/2024/08/26/meteor-metric-in-nlp-how-it-works-how-to-tutorial-in-python/
https://en.wikipedia.org/wiki/METEOR

METEOR (cont.)

The algorithm creates an alignment between unigrams of the candidate string and the reference string.



We then calculate the precision P, recall R, and harmonic mean (with recall weighted 9x more than precision)

$$P = \frac{m}{w_t} \qquad R = \frac{m}{w_r} \qquad F_{mean} = \frac{10PR}{R + 9P}$$

m is the intersecting unigrams, w_t is candidate unigrams and w_r is reference unigrams

These measures account only for single words; they need to account for larger segments too! A penalty *p* is introduced for the grouping of the *fewest possible adjacent chunks c* in the candidate.

$$M = F_{mean}(1 - p)$$

The final score is F_{mean} with a penalty applied.

$$p = 0.5\left(\frac{c}{u_m}\right)^3$$

- u_m is is the number of unigrams that have been mapped.
- c is the number of chunks.
- This penalty has the effect of reducing F_{mean} by up to 50% without bigram or longer matches.

https://en.wikipedia.org/wiki/METEOR

$$P = \frac{m}{w_t} \qquad R = \frac{m}{w_r} \qquad F_{mean} = \frac{10PR}{R + 9P}$$

METEOR (cont.)

$$p = 0.5\left(\frac{c}{u_m}\right)^3$$

$$M = F_{mean}(1 - p)$$

Example 1

```
Candidate : the   cat   sat   on    the   mat
Reference : on    the   mat   sat   the   cat
```

P = 6/6 = 1.0
R = 6/6 = 1.0
Fmean = 10 * 1.0 * 1.0 / (9*1.0 + 1.0) = 1.0

p = 0.5 * (3 / 6)^3 = 0.0625

```
Chunks    : (the  cat)  (sat) (on   the   mat)
```

METEOR = 1.0 * (1-0.0625) = **0.9375**

Example 2

```
Candidate : the   cat   sat   on    the   mat
Reference : the   cat   sat   on    the   mat
```

P = 6/6 = 1.0
R = 6/6 = 1.0
Fmean = 10 * 1.0 * 1.0 / (9*1.0 + 1.0) = 1.0

p = 0.5 * (1 / 6)^3 = 0.0023

```
Chunks    : (the  cat   sat   on    the   mat)
```

METEOR = 1.0 * (1-0.0023) = **0.9977**

https://en.wikipedia.org/wiki/METEOR

Translation Edit Rate (TER) [2006]

Introduced in 2006 Paper: Snover et. al. A Study of Translation Edit Rate with Targeted Human Annotation. https://aclanthology.org/2006.amta-papers.25/ . This metric was introduced with the human-edited version (HTER), which will be introduced later.

It is an intuitive measure that measures the amount of editing that a human would have to perform to change a system output so it exactlv matches the reference translation.

$$\text{TER} = \frac{\text{\# of edits}}{\text{average \# of reference words}}$$

Possible edits include the insertion, deletion, and substitution of single words as well as shifts of word sequences. A shift moves a contiguous sequence of words within the hypothesis to another location within the hypothesis.

Punctuation tokens are treated as normal words, and miscapitalization is counted as an edit.

https://aclanthology.org/2006.amta-papers.25.pdf

TER (cont.)

Reference Translation (Ground Truth) [14 tokens]
“The President held a press conference today to discuss the economy and upcoming policies.”

Generated (Hypothesis) Translation [13 tokens]
“The President spoke at a press conference about the economy and new policies.”

Step 1: Count Edits

1. “held” → “spoke” → Substitution (1 edit)
2. “today to discuss” → Removed → Deletion (2 edits)
3. “upcoming” → “new” → Substitution (1 edit)

Step 2: Compute TER Score

$$TER = \frac{\text{Total Edits}}{\text{Total Words in Reference Translation}} \qquad TER = \frac{4}{14} = 0.429$$

Metric	Value
Total Edits	4
Total Words in Reference	14
TER Score	0.429

chrF & chrF++

- A score based on character n-gram precision and recall.
  - No tokenization required!
- The score averages over all n-grams.
  - The widely used number is 6 characters
- Later, chrF++ adds *word* n-gram (2-gram) to the metric since it correlates more strongly with human judgement.
  - Now tokenization is required.

$$\text{CHRF}\beta = (1+\beta^2)\frac{\text{CHRP}\cdot\text{CHRR}}{\beta^2\cdot\text{CHRP}+\text{CHRR}}$$

where CHRP and CHRR stand for character n-gram precision and recall arithmetically averaged over all n-grams:

- CHRP
  percentage of n-grams in the hypothesis which have a counterpart in the reference;
- CHRR
  percentage of character n-grams in the reference which are also present in the hypothesis.

and β is a parameter which assigns β times more importance to recall than to precision – if $\beta = 1$, they have the same importance.

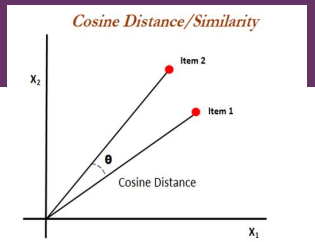
Recent experiments have shown that adding word 1-grams and 2-grams to the standard character 6-grams improves the Pearson correlation with direct human assessments. If you want to use only character n-grams, just set the word n-gram order to 0.

https://github.com/m-popovic/chrF
https://aclanthology.org/W15-3049.pdf

Semantic Similarity

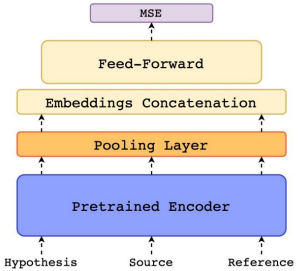
BERTScore

Analyzes cosine distances between BERT representations of machine translation and human reference. More robust against word reordering and paraphrasing. May be unreliable for terminologies underrepresented in BERT model.



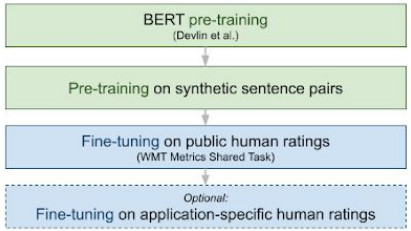
COMET

Uses a neural model (i.e. XLM-RoBERTa + Pooling/Feed-Forward) to predict machine translation quality (use source input and reference translation). More correlated with human judgements than BLEU. May penalize paraphrases/synonyms.



BLEURT

Improves over BERTScore by fine-tuning on human evaluation scores (like MQM, DA) to predict translation quality with strong human correlation.



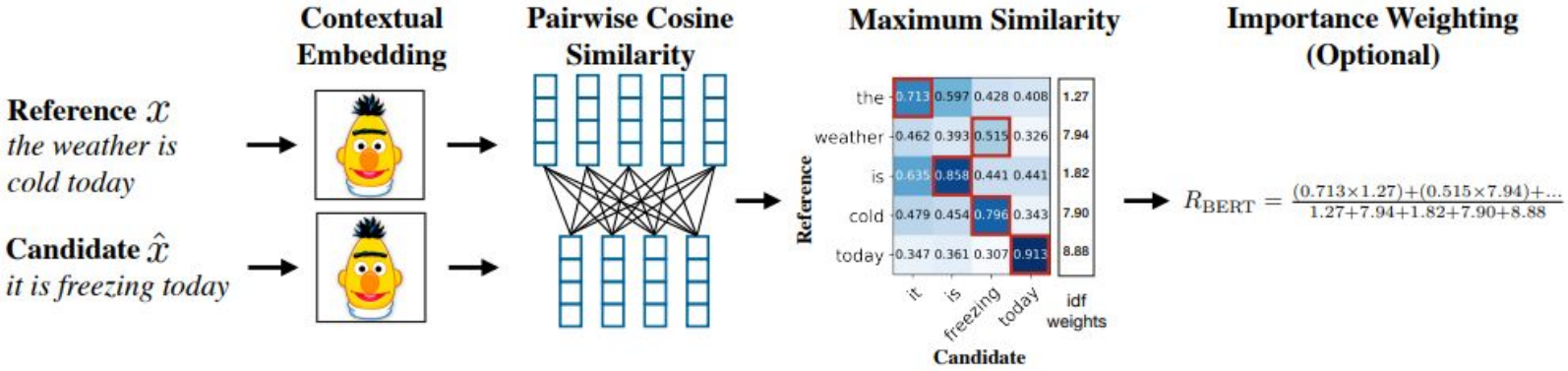
COMET https://aclanthology.org/2020.emnlp-main.213.pdf
BLEURT https://research.google/blog/evaluating-natural-language-generation-with-bleurt/

BERTScore

Introduced in 2020 ICLR Paper: Zhang et. al. [BERTScore: Evaluating Text Generation with BERT]

Uses **pre-trained BERT embeddings** to match words in candidate and reference with **cosine similarity**.

Shown to correlate with human judgment on sentence-level and system-level evaluation.
It's *semantic nature* can be useful for evaluating language generation tasks (instead of syntactic similarity).



https://arxiv.org/abs/1904.09675

BERTScore (cont.)

Matches each token x to xˆ to compute recall and xˆ to x to compute precision.
Uses greedy matching to maximize the matching similarity score (each token is matched to the most similar token in the other sentence).

F1 is the harmonic mean of the two scores.

$$R_{\text{BERT}} = \frac{1}{|x|} \sum_{x_i \in x} \max_{\hat{x}_j \in \hat{x}} \mathbf{x}_i^\top \hat{\mathbf{x}}_j \quad , \quad P_{\text{BERT}} = \frac{1}{|\hat{x}|} \sum_{\hat{x}_j \in \hat{x}} \max_{x_i \in x} \mathbf{x}_i^\top \hat{\mathbf{x}}_j \quad , \quad F_{\text{BERT}} = 2 \frac{P_{\text{BERT}} \cdot R_{\text{BERT}}}{P_{\text{BERT}} + R_{\text{BERT}}} \quad .$$

Importance Weighting

$$\text{idf}(w) = -\log \frac{1}{M} \sum_{i=1}^{M} \mathbb{I}[w \in x^{(i)}] \quad ,$$

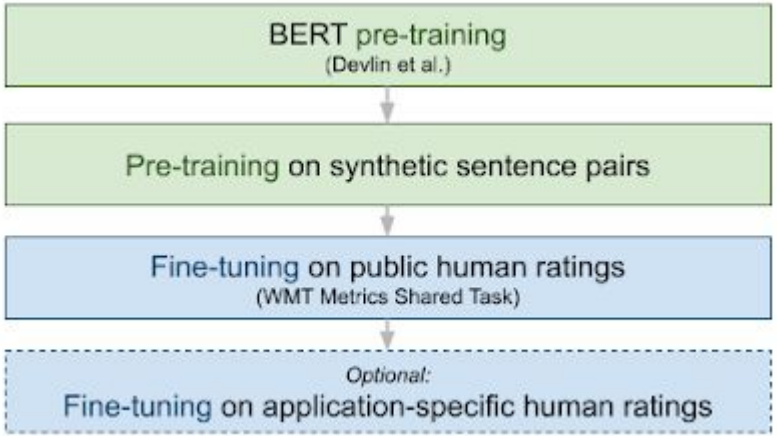
Rare words can be more indicative of sentence similarity than common words.

Inverse Document Frequency (IDF) score can be optionally used to put more weight on more rare words.

https://arxiv.org/abs/1904.09675

BLEURT (Bilingual Evaluation Understudy with Representations from Transformers)

Improves over BERTScore by fine-tuning on human evaluation scores (like MQM, DA) to predict translation quality with strong human correlation.



Metric Name	***Kendall Tau w. Human Ratings*** *(mean of all to-English lang. pairs)*
sentenceBLEU	22.7
BERTscore w. BERT-large	30.0
YiSi1 SRL	30.4
ESIM	31.6
BLEURT w. BERT-base	*33.6*
BLEURT w. BERT-large	*33.8*

BERTScore vs. BLEURT

- BERTScore: Computes cosine similarity between token embeddings in the reference and hypothesis.
- BLEURT: Uses a fine-tuned BERT model trained on human-rated translations, making it more aligned with human judgment.

Feature	BERTScore	BLEURT
Full Name	BERT-based Sentence Similarity Score	Bilingual Evaluation Understudy with Representations from Transformers
Developed By	Columbia University	Google Research
Model Type	Uses **pretrained BERT embeddings**	Fine-tuned BERT with human ratings
Scoring Method	Measures **cosine similarity** between token embeddings	Uses **a regression model** trained on human evaluation data
Needs Fine-Tuning?	❌ No	✅ Yes (pretrained BLEURT models available)
Handles Synonyms?	✅ Yes	✅ Yes
Context-Aware?	✅ Yes	✅ Yes (More human-like)
Handles Fluency & Grammar?	❌ No	✅ Yes
Best For	General **semantic similarity** evaluation	**Machine Translation (MT) & Text Generation evaluation**
Performance	Works well for **short sentences**	Works well for **longer texts**

Generated Translations:

1. **Good Translation (High Similarity)**

"The researcher released a pioneering study about global warming."

2. **Bad Translation (Unrelated Meaning)**

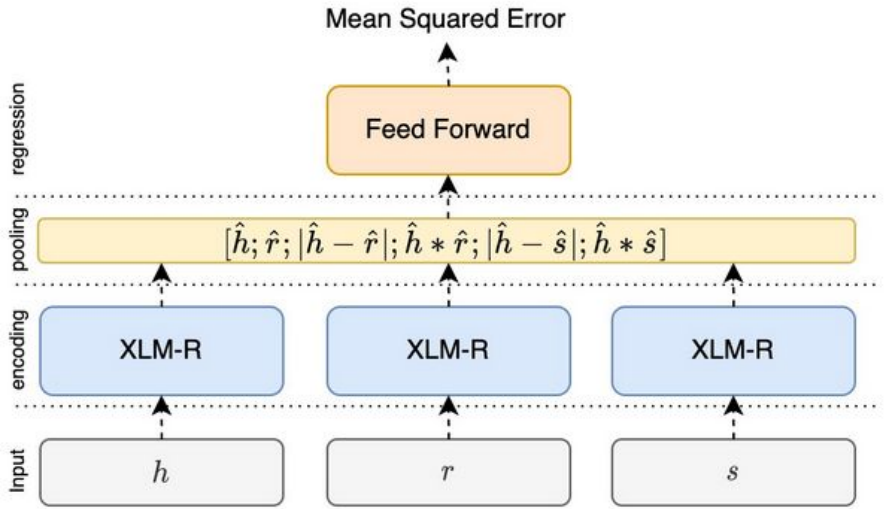
"A new book was written about space exploration."

📌 **Expected Scores for Each Metric**

Metric	Good Translation	Bad Translation
BERTScore	**0.92** (High)	**0.30** (Low)
BLEURT	**0.85** (High)	**-0.20** (Very Low)

COMET (with reference): variation 1

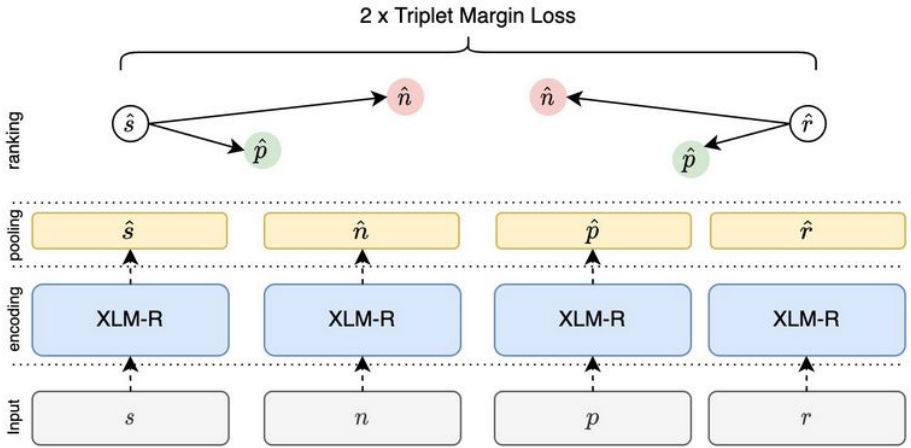
- Given a hypothesis (prediction) *h*, a reference (answer) *r*, and a source *s* as inputs
- COMET uses a multilingual encoder (i.e. XLM-R) to extract the features from the inputs.
- Concatenate them and feed it to a feed-forward regressor.
- The target score can be anything from humans, such as HTER or DA.



https://github.com/Unbabel/COMET

COMET (with reference): variation 2

- With a different architecture, it can also rank two different translations
- Given a worse translation *n*, a better translation *p*, a reference (answer) *r*, and a source *s* as inputs
- COMET uses a multilingual encoder (i.e. XLM-R) to extract the features from the inputs.
- Optimizes on the triplet loss.



What if we don't have a reference (answer)?
Quality Estimation

https://github.com/Unbabel/COMET

Quality Estimation

An actively researched field where one attempts to **estimate the quality of a translation** *without* access to a reference translation.

This is a particularly hard task since neural networks are known to often be confident while giving wrong answers.

	Direct Assessment												
Encoder	**km-en**	**ps-en**	**en-ja**	**en-cs**	**en-mr**	**ru-en**	**ro-en**	**en-zh**	**en-de**	**et-en**	**si-en**	**ne-en**	**avg.**
Baseline (Zerva et al., 2021)													
XLM-R	0.615	0.601	0.295	0.535	0.419	0.703	0.828	0.513	0.500	0.806	0.565	0.793	0.598
Pretrained models													
InfoXLM	0.619	0.603	0.328	0.510	0.462	0.731	0.829	0.554	0.516	0.803	0.561	0.777	0.608
RemBERT	0.600	0.621	0.338	0.525	0.447	0.680	0.818	0.487	0.491	0.810	0.525	0.747	0.591
XLM-R	0.610	0.579	0.325	0.503	0.405	0.715	0.832	0.541	0.514	0.782	0.540	0.740	0.591
Sentence-level only													
XLM-R	0.628	0.591	0.350	0.531	0.551	0.761	0.859	0.577	0.568	0.800	0.565	0.796	0.631
InfoXLM	0.629	0.623	0.348	0.515	0.574	0.747	0.858	0.586	0.551	0.828	0.568	0.790	0.635
RemBERT	0.634	0.631	0.346	0.570	0.564	0.754	0.862	0.534	0.531	0.822	0.550	0.782	0.632
Few-shot Language Adaptation													
XLM-R	0.650	0.619	0.352	0.551	0.546	0.753	0.852	0.571	0.554	0.813	0.562	0.798	0.635
InfoXLM	0.641	0.650	0.367	0.549	0.549	0.751	0.855	0.591	0.565	0.824	0.563	0.803	0.642
RemBERT	0.625	0.641	0.367	0.568	0.563	0.756	0.857	0.540	0.527	0.824	0.568	0.796	0.636
Sentence + word-level training													
InfoXLM	0.617	0.586	0.344	0.532	0.572	0.761	0.865	0.586	0.579	0.829	0.576	0.804	0.637
RemBERT	0.634	0.628	0.356	0.564	0.571	0.762	0.860	0.541	0.553	0.826	0.564	0.799	0.638
Few-shot Language Adaptation													
InfoXLM	0.643	0.632	0.335	0.557	0.560	0.766	0.860	0.575	0.582	0.833	0.578	0.809	0.644
RemBERT	0.644	0.645	0.356	0.567	0.568	0.759	0.856	0.545	0.552	0.835	0.561	0.804	0.641
Final Ensemble													
Ensemble 6x	**0.664**	**0.669**	**0.380**	**0.591**	**0.593**	**0.782**	**0.871**	**0.597**	**0.593**	**0.845**	**0.588**	**0.820**	**0.666**

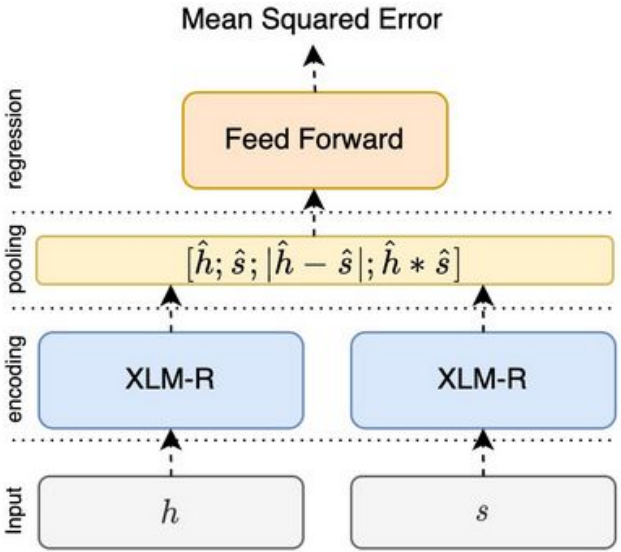
Even the best system does not exactly correlate with human judgement

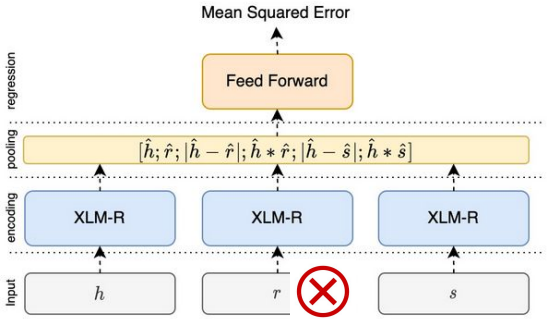
Table 1: Results for sentence-level QE in terms of Spearman correlation for DA.

https://github.com/Unbabel/COMET

COMET (without reference)

- **Everything is the same with the first variation of COMET with reference, except you don't give it a reference text.**
- Given a hypothesis *h* and a source *s* as inputs
- COMET uses a multilingual encoder (XLM-R) to extract the features from the inputs.
- Concatenate them and feed it to a feed-forward regressor.
- The target score can be anything from humans, such as HTER or DA.



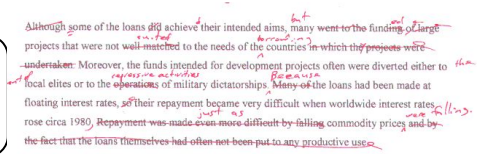


https://github.com/Unbabel/COMET

Human Judgement

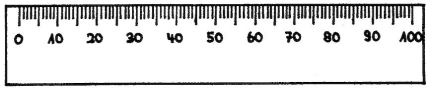
HTER

Human-targeted Translation Edit Rate: Similar to TER, but instead of using a fixed reference translation, it uses an edited version of the machine translation that has been manually corrected by a human.



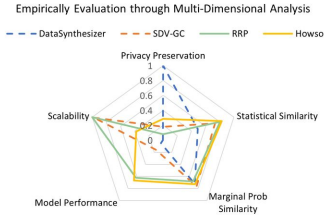
DA

Human annotators score translations on a continuous scale (0-100).



MQM

Multidimensional Quality Metrics is a more detailed framework used in professional translation assessment, assigning a score for each dimensions.



Human Evaluation

Human judgements of MT quality usually come in the form of segment-level scores, such as:

1. **Human-targeted Translation Edit Rate (HTER) [1]**
   a. the MT outputs are manually corrected (#edit)
   b. then the original outputs are compared to the edited ones by computing TER.

$$\text{TER} = \frac{\#\text{ of edits}}{\text{average \# of reference words}}$$

2. **Direct Assessment (DA)**
   a. A quality score (satisfaction score) of 0 to 100 is given for a translation by human

3. **Multidimensional Quality Metrics (MQM)**

	A	B	C	D	E	F	G	H
1	MQM Scorecard: Top-Level Error Typology with 4 Severity Levels							
2								
3		*Error Severity Levels:*	Neutral	Minor	Major	Critical	Error Type Penalty Total	
4		*Severity Penalty Multipliers:*	0	1	5	25		
5	ET Nos	Error Types	Error Counts				ET Weights	ETPTs
6	1	Terminology	2	7	7	0	1.0	42.0
7	2	Accuracy	4	14	7	1	1.0	74.0
8	3	Linguistic conventions	1	23	9	0	1.0	68.0

[1] https://www.cs.umd.ed

https://themqm.org/error-types-2/1_scorecards/values-and-scores/

	A	B	C	D	E	F	G	H
1	MQM Scorecard: Top-Level Error Typology with 4 Severity Levels							
2								
3		*Error Severity Levels:*	Neutral	Minor	Major	Critical	Error Type Penalty Total	
4		*Severity Penalty Multipliers:*	0	1	5	25		
5	ET Nos	Error Types	Error Counts				ET Weights	ETPTs
6	1	Terminology	2	7	7	0	1.0	42.0
7	2	Accuracy	4	14	7	1	1.0	74.0
8	3	Linguistic conventions	1	23	9	0	1.0	68.0
9	4	Style	5	7	3	0	1.0	22.0
10	5	Locale convention	1	12	5	0	1.0	37.0
11	6	Audience appropriateness	0	2	1	0	1.0	7.0
12	7	Design and markup	0	6	1	0	1.0	11.0
13	8	Custom						
14							Absolute Penalty Total:	261.00
15								
16		Evaluation Word Count:	10184				Per-Word Penalty Total:	0.0256
17		Reference Word Count:	1000				Overall Normed Penalty Total:	25.63
18		Scaling Parameter (SP):	1.00				Overall Quality Score:	97.44
19		Max. Score Value:	100.00					
20		Threshold Value:	85.00				Pass/Fail Rating:	Pass

Figure 1: Scorecard with Top-Level MQM Error Types

	A	B	C	D	E	F	G	H
1	MQM Scorecard: Top-Level Error Typology with 4 Severity Levels							
2								
3		Error Severity Levels:	Neutral	Minor	Major	Critical	Error Type Penalty Total	
4		Severity Penalty Multipliers:	0	1	5	25		
5	ET Nos	Error Types	Error Counts				ET Weights	ETPTs
6	1	Terminology	2	7	7	0	1.0	42.0
7	2	Accuracy	4	14	7	1	1.0	74.0
8	3	Linguistic conventions	1	23	9	0	1.0	68.0
9	4	Style	5	7	3	0	1.0	22.0
10	5	Locale convention	1	12	5	0	1.0	37.0
11	6	Audience appropriateness	0	2	1	0	1.0	7.0
12	7	Design and markup	0	6	1	0	1.0	11.0
13	8	Custom						
14							Absolute Penalty Total:	261.00
15								
16		Evaluation Word Count:	10184				Per-Word Penalty Total:	0.0256
17		Reference Word Count:	1000				Overall Normed Penalty Total:	25.63
18		Scaling Parameter (SP):	1.00				Overall Quality Score:	97.44
19		Max. Score Value:	100.00					
20		Threshold Value:	85.00				Pass/Fail Rating:	Pass

Figure 1: Scorecard with Top-Level MQM Error Types

The scorecard can be confusing on first glance. It is important to understand the different values entered on the scorecard, how an evaluator uses the card, and how the math functions

Which Scores to Use?

- **BLEU** & **ROUGE (syntactic)** → If you need a quick, traditional metric.
- **METEOR (syntactic)** → If BLEU seems too rigid (captures synonyms), languages with WordNet support.
    - Language with WordNet support:

      English, Spanish, French, German

Language	Exact Match	Stem Match	Synonym Match	Paraphrase Match	Tuned Parameters
English	Yes	Yes	Yes	Yes	Yes
Arabic	Yes	No	No	Yes	Yes
Czech	Yes	No	No	Yes	Yes
French	Yes	Yes	No	Yes	Yes
German	Yes	Yes	No	Yes	Yes
Spanish	Yes	Yes	No	Yes	Yes

- **ChrF (syntactic)** → F1-Score for unsupported languages & morphologically rich languages (Thai)
- **BERTScore / COMET / BLEURT (semantic)** → If you care about semantic meaning.
- **Human Evaluation** (HTER, MQM, DA) (human) → Best for high-quality MT assessment.

METEOR supported languages https://www.cs.cmu.edu/~alavie/METEOR/README.html

Which Scores to Use?

- A paper from Microsoft [1] advocates for **COMET** and **ChrF** (for automatic metrics).

To Ship or Not to Ship:
An Extensive Evaluation of Automatic Metrics for Machine Translation

Tom Kocmi **Christian Federmann** **Roman Grundkiewicz** **Marcin Junczys-Dowmunt** **Hitokazu Matsushita** **Arul Menezes**

Microsoft
1 Microsoft Way
Redmond, WA 98052, USA
{tomkocmi,chrife,rogrundk,marcinjd,himatsus,arulm}@microsoft.com

Abstract

Automatic metrics are commonly used as the exclusive tool for declaring the superiority of one machine translation system's quality over another. The community choice of automatic metric guides research directions and industrial developments by deciding which models are deemed better. Evaluating metrics correlations with sets of human judgements has been limited by the size of these sets. In this paper, we corroborate how reliable metrics are in contrast to human judgements on – to the errors (Freitag et al., 2021), and thus may mislead system development by incorrect judgements. Therefore, it is important to study the reliability of automatic metrics and follow best practices for the automatic evaluation of systems.

Significant research effort has been applied to evaluate automatic metrics in the past decade, including annual metrics evaluation at the WMT conference and other studies (Callison-Burch et al., 2007; Przybocki et al., 2009; Stanojević et al., 2015; Mathur et al., 2020b). Most research has fo-

13 Sep 2021

Based on our findings, we suggest the following best practices for the use of automatic metrics:

1. Use a pretrained metric as the main automatic metric; we recommend COMET. Use a string-based metric for unsupported languages and as a secondary metric, for instance ChrF. Do not use BLEU, it is inferior to other metrics, and it has been overused.

[1] https://arxiv.org/pdf/2107.10821.pdf