Machine Translation Evaluation

Machine Translation: Advanced Topics

Vincent Vandeghinste

KU Leuven – Brussels

2026

Outline

Machine Translation Evaluation

Machine Translation Evaluation

- Central question in MT: *How good is a translation?*
- Hard because translation is **not** single-answer:
  - multiple outputs can be equally acceptable;
  - variation in syntax, word choice, information structure.
- Evaluation paradigms (research + shared tasks):
  - Human evaluation (rating, ranking, DA, MQM, post-editing).
  - Automatic metrics (surface overlap, edit distance, neural metrics).
  - Statistical significance testing for metric differences.
  - Evaluating evaluation metrics (correlation with human judgments).
  - Task-based evaluation and ethics/bias.

Why MT Evaluation Is Difficult

Why MT Evaluation Is Difficult I

- Many distinct translations can preserve meaning adequately.
- Surface forms differ while meaning remains stable.
- Metrics and humans may disagree depending on what they reward:
  - adequacy vs fluency,
  - literalness vs paraphrase,
  - content coverage vs stylistic acceptability.

Why MT Evaluation Is Difficult II

Example of Translations

Chinese: 这个 机场 的 安全 工作 由 以色列 方面 负责 .

Translations:

- Israeli officials are responsible for airport security.
- Israel is in charge of the security at this airport.
- The security work for this airport is the responsibility of the Israel government.
- Israeli side was in charge of the security of this airport.
- Israel is responsible for the airport's security.
- Israel is responsible for safety work at this airport.
- Israel presides over the security of the airport.
- Israel took charge of the airport security.
- The safety of this airport is taken charge of by Israel.
- This airport's security is the responsibility of the Israeli security officials.

Human Evaluation

Human Evaluation: Overview

- Considered most reliable, but expensive and time-consuming.
- Applies to:
  - evaluation of **machine** translation outputs,
  - evaluation of **human** translations (quality assessment).
- Standard in shared tasks (e.g. WMT, IWSLT).

Rating-Based Evaluation: Adequacy and Fluency

- Two classic dimensions:
  - **Adequacy:** meaning preservation (needs bilingual assessor, or compare to reference).
  - **Fluency:** grammaticality / naturalness in target language (monolingual possible).
- Often collected on 1–5 Likert scales.

Adequacy	
5	all meaning
4	most meaning
3	much meaning
2	little meaning
1	no meaning

Fluency	
5	flawless target language
4	good target language
3	non-native target language
2	disfluent target language
1	incomprehensible

Exercise: Assess Translation Quality

Rank according to adequacy and fluency on a 1–5 scale.

Source: L'affaire NSA souligne l'absence totale de d´ebat sur le renseignement

Reference: NSA Affair Emphasizes Complete Lack of Debate on Intelligence

System1: The NSA case underscores the total lack of debate on intelligence

System2: The case highlights the NSA total absence of debate on intelligence

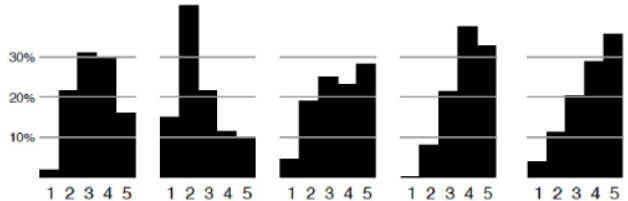
System3: The matter NSA underlines the total absence of debates on the piece of information

Likert Scales: Key Limitation (Ordinal Data)

- Likert scales are **ordinal**:
  - 4 is better than 3, but distances are not guaranteed equal.
  - difference (1,2) may not be comparable to difference (4,5).
- Treating scores as interval data can mislead:
  - averages assume equal spacing;
  - better: distributions, medians, robust summaries.

Inter-rater Agreement

- Multiple annotators often disagree substantially.
- **Inter-rater agreement:** how consistently annotators judge the same outputs.
- Low agreement:
  - subjective interpretation,
  - different tolerance for errors,
  - domain familiarity and expectations,
  - different scale usage.
- Agreement can be quantified (e.g. Cohen's κ, Krippendorff's α), often low for adequacy/fluency ratings.



Rater Effects, Training, Normalization

- Disagreement is not only about translations, but also about raters:
  - strict vs lenient raters,
  - inconsistent use of the scale,
  - adequacy and fluency conflated.
- Common practice in large-scale evaluation:
  - filter out unreliable/outlier annotators,
  - z-**normalize per rater** (mean 0, variance 1) to reduce scale-usage differences.

Direct Assessment (DA)

- WMT moved from discrete ratings to continuous scales (e.g. 0–100 sliders).
- **Direct Assessment (DA):**
  - finer-grained judgments,
  - improved agreement vs 1–5 scales,
  - strong annotation protocols.
- Widely used in WMT human evaluation since late 2010s.

Ranking-Based Evaluation

- Avoid absolute scales: ask for **relative preferences**.
- Often easier: compare translations of the same source sentence.
- Advantages:
  - less scale interpretation noise,
  - often higher agreement than adequacy/fluency ratings.
- Limitations:
  - no absolute quality notion,
  - comparisons explode with many systems.

Ranking Translations Example

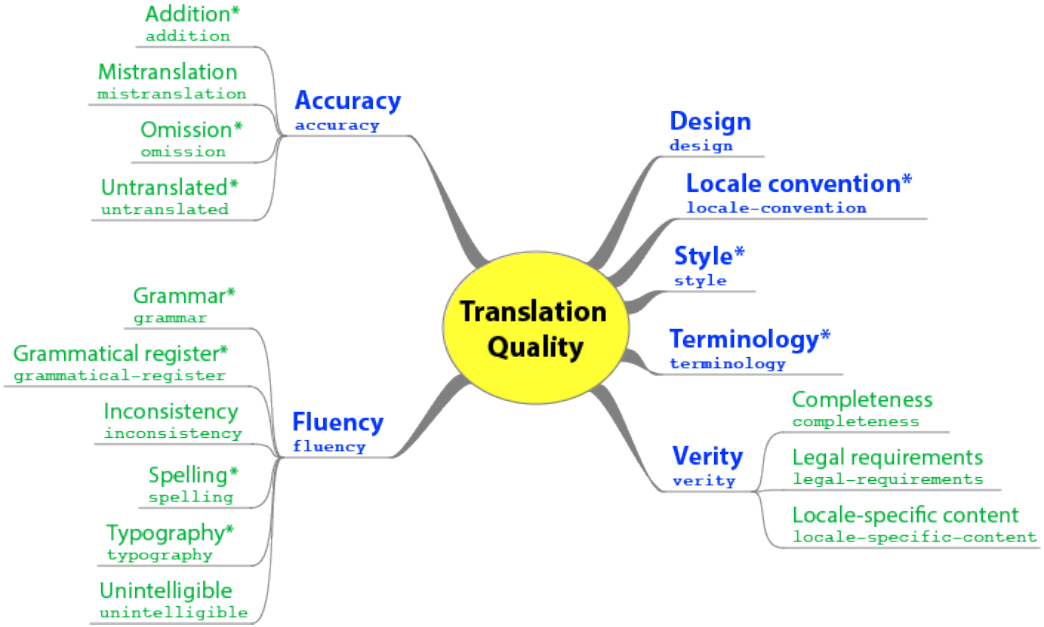
Is translation A better than, worse than, or equal to translation B?

Translation A Israeli officials are responsible for airport security.

Translation B Israel is in charge of the security at this airport.

MQM: Multidimensional Quality Metrics

- Holistic scoring $\rightarrow$ **error annotation** with a taxonomy.
- Annotators:
  - identify errors,
  - categorize them (accuracy, fluency, terminology, style, locale, ...),
  - assign severity (minor / major / critical),
  - aggregate into penalty-based scores (often normalized).
- Strength: diagnostic (*why* quality is low).
- Cost: expensive, requires training + detailed guidelines.


Translation Quality
Accuracy
accuracy
Addition*
addition
Mistranslation
mistranslation
Omission*
omission
Untranslated*
untranslated
Fluency
fluency
Grammar*
grammar
Grammatical register*
grammatical-register
Inconsistency
inconsistency
Spelling*
spelling
Typography*
typography
Unintelligible
unintelligible
Design
design
Locale convention*
locale-convention
Style*
style
Terminology*
terminology
Verity
verity
Completeness
completeness
Legal requirements
legal-requirements
Locale-specific content
locale-specific-content

Post-editing as Evaluation

- Measure quality indirectly via **effort to correct MT output**.
- Better MT $\Rightarrow$ less human editing time/effort.
- Connects evaluation to professional workflows.

- Post-editing yields an **absolute** sentence-level signal (not only relative preferences).
- Still sensitive to annotator/editing style (rater effects remain possible).

HTER: Human-targeted Translation Edit Rate

- HTER quantifies post-editing effort:

$$\text{HTER} = \frac{\#\text{edits}}{\#\text{words in post-edited translation}}$$

- Allowed operations:
  - insertion, deletion, substitution, **shift** (move a contiguous block).

HTER Calculation Example

Consider the following English–Dutch translation:
Source Yesterday, the minister announced the decision.

MT output	De	minister	kondigde	de	beslissing	aan	gisteren
Post-edited	Gisteren	kondigde	de	minister	de	beslissing	aan

To obtain the post-edited translation, the annotator performs **two shift** operations.

- shift 1: move *gisteren* to sentence-initial position;
- shift 2: move *de minister* to the postverbal position.

The post-edited sentence has 7 words, so:

$$\text{HTER} = \frac{2}{7} \approx 0.29$$

Why two edits? Each shift moves one contiguous block. Moving *gisteren* alone does not yield the final order; shifting *de minister* is also required.

Automatic Evaluation Metrics

Automatic Evaluation Metrics: Why and How

- Goal: approximate human judgments **cheaply and consistently**.
- Inputs:
  - MT output + one or more references (reference-based evaluation).
- Historical trend:
  - surface overlap $\rightarrow$ edit distance $\rightarrow$ neural/semantic metrics.

Precision and Recall as Evaluation Concepts

- Many classic metrics quantify overlap between hypothesis and reference.
- Two complementary notions:
  - **Precision:** how much of the system output is correct (avoid spurious content).
  - **Recall:** how much reference content is recovered (avoid omissions).
- With M matches, H hypothesis units, R reference units:

$$\text{Precision} = \frac{M}{H} \qquad \text{Recall} = \frac{M}{R}$$

- Combined as an F-score:

$$F_\beta = \frac{(1+\beta^2)\,P\,R}{\beta^2\,P + R}$$

Precision / Recall Example I

System A:
`Israeli officials responsibility of airport safety`

Reference:
`Israeli officials are responsible for airport security`

Assume a simple word-overlap matching with three matches: `Israeli`, `officials`, `airport`.

- **correct** $M = 3$.
- **output-length** $H = 6$ (`Israeli`, `officials`, `responsibility`, `of`, `airport`, `safety`)
- **reference-length** $R = 7$ (`Israeli`, `officials`, `are`, `responsible`, `for`, `airport`, `security`)

Precision / Recall Example II

$$P = \frac{3}{6} = 0.50 \qquad R = \frac{3}{7} \approx 0.43$$

F-measure (harmonic mean)

$$F_1 = \frac{2PR}{P+R} = \frac{2 \cdot 0.50 \cdot 0.43}{0.50 + 0.43} \approx 0.46$$

Equivalent form

$$F_1 = \frac{P \cdot R}{(P+R)/2} \approx 0.46$$

Limitation: Insensitivity to Word Order

Reference:
`Israeli officials are responsible for airport security`

System A:
`Israeli officials responsibility of airport safety`

System B:
`airport security Israeli officials are responsible`

Under simple word-overlap, System B can achieve perfect precision and recall (because it contains the same words), yet it is not well-formed due to word order and broken syntactic relations.

- Overlap counts *what* words appear, not *where* they occur.
- A translation can score perfectly and still be hard to understand.

Edit Distance and Word Error Rate (WER) I

- Count edits needed to transform MT output into reference:
  - insertions, deletions, substitutions (word-level Levenshtein).
- Normalize by reference length:

$$\text{WER} = \frac{E}{N}$$

- Limitations for MT:
  - legitimate reordering is penalized heavily;
  - synonyms treated as errors (exact match only).

Ref	Has	France	benefited	from	information	provided	by	the	NSA	?
MT	Did	France	profit	from	information	supplied	by	the	NSA	?
Type	S	M	S	M	M	S	M	M	M	M

Three substitutions, reference length 10:

$$\text{WER} = \frac{3}{10} = 0.30$$

BLEU: *n*-gram Precision + Brevity Penalty I

- BLEU (Papineni et al. 2002):
  - overlap of word *n*-grams (usually $n = 1..4$),
  - combine using geometric mean,
  - brevity penalty discourages overly short outputs.
- Usually computed at **corpus level** (sentence-level is unstable).
- Strengths: simple, fast, language-independent, long-standing standard.
- Weaknesses:
  - exact match only (paraphrases penalized),
  - precision-oriented (no explicit recall),
  - sensitive to tokenization and morphology.

BLEU Example: N-gram Precision and Brevity Penalty

Reference: Has France benefited from information provided by the NSA ?
MT output: Did France profit from information supplied by the NSA ?

Step 1: n-gram precisions

$$p_1 =, \quad p_2 =, \quad p_3 =, \quad p_4 = .$$

Step 2: geometric mean

$$\text{GM} = \sqrt[4]{p_1 p_2 p_3 p_4}.$$

Step 3: brevity penalty

$$c =, \quad r =, \Rightarrow \text{BP} = min(1, \frac{c}{r}) = .$$

Final

$$\text{BLEU} = \text{BP} \times \text{GM}.$$

BLEU Example: N-gram Precision and Brevity Penalty

Reference: Has France benefited from information provided by the NSA ?
MT output: Did France profit from information supplied by the NSA ?

Step 1: n-gram precisions

$$p_1 = \frac{7}{10}, \quad p_2 = \frac{4}{9}, \quad p_3 = \frac{2}{8}, \quad p_4 = \frac{1}{7}.$$

Step 2: geometric mean

$$\mathrm{GM} = \sqrt[4]{p_1 p_2 p_3 p_4}.$$

Step 3: brevity penalty

$$c = r = 10 \Rightarrow \mathrm{BP} = 1.$$

Final

$$\mathrm{BLEU} = \mathrm{BP} \times \mathrm{GM}.$$

BLEU with Multiple References

- Compare MT output to **all** references:
  - an *n*-gram match counts if it appears in *any* reference.
- Use **clipped counts**:
  - limit matches per *n*-gram by the maximum reference count.
- Brevity penalty uses **closest reference length** to output length.
- Motivation: accommodate translation variability without rewarding repetition.

BLEU: Main Limitations

- No explicit recall: omissions can be under-penalized.
- Sentence-level BLEU is unstable (sparse higher-order n-grams).
- Exact match only: penalizes synonymy and paraphrase strongly.
- Morphology breaks many n-grams in rich languages.
- Tokenization dependence complicates comparison across papers/systems.

TER and HTER (Edit Operations + Shifts)

- TER (Snover et al. 2006) extends WER with **shift** operations:
  - moving a contiguous block counts as one edit.
- Normalized by reference length.
- HTER: same operations, but reference is a **post-edited** version of MT output.
- TER measures similarity to a reference; HTER aligns with human correction effort.

chrF: Character *n*-grams + Precision/Recall I

- chrF (Popovic 2015) uses **character *n*-grams**:
  - less sensitive to tokenization,
  - more tolerant to morphology and minor orthographic variation.
  - G: multisets of character n-grams
- Explicitly models both precision and recall:

$$\text{chrP} = \frac{|G_{\text{MT}} \cap G_{\text{ref}}|}{|G_{\text{MT}}|} \qquad \text{chrR} = \frac{|G_{\text{MT}} \cap G_{\text{ref}}|}{|G_{\text{ref}}|}$$

- Combine via weighted F-score:

$$\text{chrF}_\beta = \frac{(1+\beta^2)\,\text{chrP} \cdot \text{chrR}}{\beta^2 \cdot \text{chrP} + \text{chrR}}$$

- Commonly $\beta = 2$ (recall weighted more than precision).

chrF example: setup

Reference: `witness for the past`

Hypothesis 1: `witness of the past`

chrF operates on **character *n*-grams**:

- spaces removed before extraction;
- character *n*-grams for $n = 1 \ldots 6$;
- repeated *n*-grams are counted *with multiplicity*.

Strings used for scoring:

- Reference: `witnessforthepast`
- Hypothesis: `witnessofthepast`

chrF example: character n-gram counts

For each n-gram order, chrF counts:

- total MT n-grams;
- total reference n-grams;
- overlapping n-grams (multiset intersection).

n	$\lvert G_{\text{MT}}^{(n)} \rvert$	$\lvert G_{\text{ref}}^{(n)} \rvert$	$\lvert G_{\cap}^{(n)} \rvert$
1	16	17	16
2	15	16	14
3	14	15	13
4	13	14	12
5	12	13	10
6	11	12	9

chrF example: micro-averaging

chrF uses **micro-averaging**:

- counts are summed across all n;
- precision and recall are computed from totals;
- per-n scores are *not* averaged.

$$\sum |G_{\cap}^{(n)}| = 74 \qquad \sum |G_{\mathrm{MT}}^{(n)}| = 81 \qquad \sum |G_{\mathrm{ref}}^{(n)}| = 87$$

$$\mathrm{chrP} = \frac{74}{81} \approx 0.91 \qquad \mathrm{chrR} = \frac{74}{87} \approx 0.85$$

chrF example: final score

chrF combines micro-averaged precision and recall using an F-measure (usually $\beta = 2$):

$$\text{chrF}_2 = \frac{(1 + 2^2) \cdot \text{chrP} \cdot \text{chrR}}{2^2 \cdot \text{chrP} + \text{chrR}} \approx 0.86$$

What this shows:

- small local changes mainly affect higher-order *n*-grams;
- lower-order *n*-grams dominate via micro-averaging;
- chrF degrades smoothly under lexical variation.

Neural Evaluation Metrics: Motivation

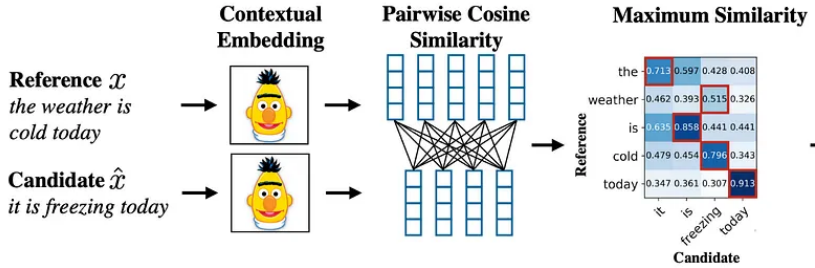
- Surface metrics (BLEU/chrF/TER) rely on overlap:
  - penalize synonymy and paraphrase,
  - miss semantic adequacy errors in fluent outputs.
- Neural metrics use pretrained models to capture semantics:
  - embedding similarity (e.g. BERTScore),
  - direct prediction of human judgments (e.g. BLEURT, COMET).
- Stronger correlation with human judgments, but:
  - more computationally expensive,
  - can be sensitive to domain shift and model assumptions.

BERTScore

- **BERTScore** is an automatic evaluation metric for machine translation.
- It measures **semantic similarity** between a system output and a reference.
- Uses **contextual word representations** from pretrained language models (e.g. BERT, RoBERTa, XLM-R).
- Unlike BLEU or chrF, it does *not* rely on exact word overlap.

Key idea: compare words based on their *meaning*, not their surface form.

How BERTScore Works



$$R_{\text{BERT}} = \frac{1}{|x|} \sum_{x_i \in x} \max_{\hat{x}_j \in \hat{x}} \mathbf{x}_i^\top \hat{\mathbf{x}}_j$$

$$P_{\text{BERT}} = \frac{1}{|\hat{x}|} \sum_{\hat{x}_j \in \hat{x}} \max_{x_i \in x} \mathbf{x}_i^\top \hat{\mathbf{x}}_j$$

$$F_{\text{BERT}} = 2 \frac{P_{\text{BERT}} \cdot R_{\text{BERT}}}{P_{\text{BERT}} + R_{\text{BERT}}}$$

- Each word is represented as a **contextual embedding**.
- All reference and system words are compared using **cosine similarity**.
- Each word is aligned to its *most similar* word in the other sentence.

What Does BERTScore Measure?

From the word alignments, BERTScore computes:

- **Recall**: Does the system output cover the meaning of the reference?
- **Precision**: Are the words in the system output relevant to the reference?
- **F-score**: A single score combining precision and recall.

Interpretation for translators:

- High score $\Rightarrow$ meaning is preserved, even if wording differs.
- Low score $\Rightarrow$ missing or incorrect semantic content.

BERTScore Example

Semantic Matching Beyond Exact Overlap

Reference: *The boy is riding a bicycle*

System output: *The child is riding a bike*

Although several words differ lexically, their meanings are very similar.

- *boy* $\leftrightarrow$ *child* (semantic similarity)
- *bicycle* $\leftrightarrow$ *bike* (near-synonyms)
- Function words align almost perfectly

Result: high precision, high recall, high BERTScore — even though BLEU would penalize missing exact matches.

BLEURT

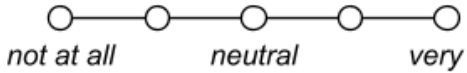
- **BLEURT** is an automatic evaluation metric for natural language generation.
- It is designed to correlate well with **human judgments of translation quality**.
- Unlike BLEU or BERTScore, BLEURT is a **learned metric**.
- It predicts a quality score rather than computing similarity by a fixed formula.

Key idea: BLEURT learns what humans consider a good or bad translation.
Images for BLEURT from `https://research.google/blog/evaluating-natural-language-generation-with-bleurt/`.

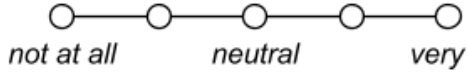
Human Judgments and Evaluation

Input: Bud Powell était un pianiste de légende.
Reference: Bud Powell was a legendary pianist.
Candidate: Bud Powell was a great pianist.

How fluent is the sentence?

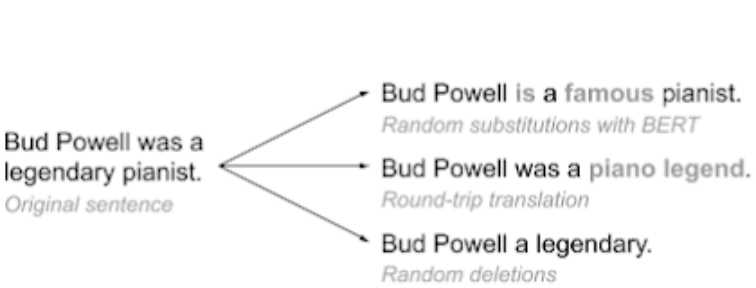


Does it accurately convey the meaning of the reference?



- Human annotators score translations for quality.
- These judgments are expensive but informative.
- BLEURT is trained to *imitate* such human scores.

Pretraining BLEURT

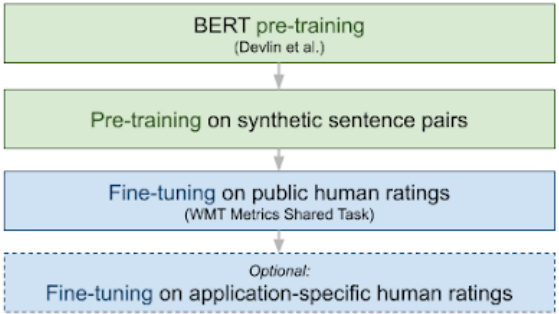


BLEU	ROUGE	...
32.1	66.7	
54.1	66.7	...
31.7	55.7	

Collection of metrics and models used as pre-training targets.

- Start from high-quality reference sentences.
- Automatically introduce errors:
  - word deletions
  - substitutions
  - reordering
- The model learns to distinguish good from degraded text.

BLEURT Training steps



Fine-tuning BLEURT

- After pretraining, BLEURT is fine-tuned on human evaluation data.
- Input: reference + system output.
- Target: human quality score.
- The model learns which errors humans find serious.

BLEURT therefore combines:

- large-scale synthetic training,
- small but high-quality human judgments.

Comparing Evaluation Metrics

- **BLEU / chrF**: surface overlap
- **BERTScore**: semantic similarity
- **BLEURT**: learned human judgment

BLEURT goes one step further by learning what quality means.

COMET

- **COMET** (Crosslingual Optimized Metric for Evaluation of Translation) is an automatic evaluation metric for machine translation.
- It is designed to correlate strongly with **human judgments of translation quality**.
- Like BLEURT, COMET is a **learned metric**.
- It predicts a quality score rather than computing similarity by a fixed formula.

Key idea: COMET learns how humans judge translation quality.

Which Metric Should Be Reported?

- No single metric captures all aspects of translation quality.
- Common contemporary practice: report multiple complementary metrics:
  - **BLEU** (continuity and comparability),
  - **chrF** (robustness to morphology/tokenization),
  - **Neural metrics** (better correlation with human judgments), often **COMET** as primary signal in recent work.
- Best practice: state clearly
  - which metric is used for model selection,
  - which metrics are reported for comparison.

What Makes COMET Different?

- COMET explicitly uses the **source sentence**.
- It compares:
  - the source sentence,
  - the system output,
  - and optionally a reference translation.
- This allows COMET to focus strongly on **adequacy**.

Intuition: A good translation should preserve the meaning of the source.

Training COMET

- COMET is trained using **human evaluation data**.
- Each training example includes:
  - a source sentence,
  - a system output,
  - a reference translation,
  - a human quality score.
- The model learns to predict the human score.

Over time, COMET learns which translation errors humans consider serious.

What Is COMET Good At?

- Detecting **meaning shifts** and omissions.
- Penalising fluent but incorrect translations.
- Handling paraphrasing better than surface-based metrics.
- Showing strong correlation with human judgments.

COMET is particularly effective for evaluating **adequacy**.

Comparing Evaluation Metrics

- **BLEU / chrF**: surface overlap
- **BERTScore**: semantic similarity
- **BLEURT**: learned human judgment
- **COMET**: learned human judgment + source sentence

COMET explicitly checks whether the source meaning is preserved.

Interpreting COMET Scores

- Higher COMET score $\Rightarrow$ better translation quality.
- Scores are relative: mainly useful for **system comparison**.
- COMET does not explain *why* a translation is good or bad.

Important: COMET is a black-box metric, but a very strong predictor of human judgments.

Statistical Significance Testing

Why Significance Testing?

- Metric scores are single numbers but depend on the sampled test set.
- Small differences may be due to sampling noise, not real improvements.
- Significance testing estimates whether observed differences are likely meaningful.

Bootstrap Resampling (Concept)

- Create many pseudo-testsets by sampling sentence pairs *with replacement*.
- Compute the metric on each pseudo-testset $\Rightarrow$ score distribution.
- Use distribution to derive confidence intervals (e.g. 95% interval).

1. Sample n sentence pairs with replacement.
2. Compute metric score on sample.
3. Repeat many times (e.g. 1,000–10,000).
4. Sort scores; discard lowest/highest 2.5% for a 95% interval.

Paired Bootstrap for System Comparison

- Compare systems A and B fairly:
  - use the **same bootstrap sample** for both systems.
- For each sample compute:

$$\Delta = \text{metric}(A) - \text{metric}(B)$$

- Estimate significance as:

$$p = \Pr(\text{metric}(A) > \text{metric}(B))$$

- If $p > 0.95$, evidence for significance at the 5% level.

Interpretation

- Statistical significance $\neq$ practical relevance.
- Lack of significance can mean:
  - differences are small,
  - test set is too small/noisy to detect differences reliably.
- Bootstrap is most common for corpus-level metrics (BLEU, chrF), but can be applied to neural metrics with care (variance/domain sensitivity).

Evaluating Evaluation Metrics

Evaluating Metrics

- Metrics are useful only if they correlate with human judgments.
- WMT runs annual **Metrics Shared Tasks**:
  - compare metrics against human annotations,
  - often using MQM-style judgments in recent years.

Correlation with Human Judgments

- Common correlation measures:
  - **Pearson:** linear correlation,
  - **Spearman:** rank correlation,
  - **Kendall's τ:** robust rank-based comparison (system ranking).
- High correlation helps, but does not guarantee coverage of all quality aspects.

System-level vs. Segment-level

- **System-level:** average scores per system (typically higher correlation).
- **Segment-level:** sentence-level scores (harder; judgments noisier).
- Neural metrics typically outperform overlap metrics in both settings, especially at segment level.

Implications for Metric Choice

- No universal best metric across languages, domains, and tasks.
- Domain shift matters: metrics trained on news may degrade elsewhere.
- Practical guidance:
  - Use system-level metrics for benchmarking.
  - Use source-aware neural metrics when adequacy/faithfulness is critical.
  - Treat metric scores as proxies, not ground truth.

Task-Based Evaluation

Task-Based Evaluation

- Evaluate MT by downstream utility rather than similarity to references.
- Core idea: measure whether MT helps users achieve a task.

Post-editing Productivity (Task-based)

- Ask translators to post-edit MT output to a target standard.
- Measure productivity:
  - editing time,
  - number of edits (TER/HTER),
  - throughput (words/hour).
- Note: gains are not always linear (fluent but wrong translations can increase cognitive effort).

Content Understanding

- Evaluate usefulness for **gisting**:
  - users answer comprehension questions based on MT output.
- Relevant for:
  - information access,
  - triage,
  - crisis/humanitarian settings.
- Hard to design: controlled questions + experimental conditions.

Bias and Ethical Considerations

Bias and Ethical Considerations

- Evaluation is not value-neutral:
  - MT can reflect/amplify societal biases in training data.
  - Standard metrics may miss harms (stereotypes, offensive renderings, sensitive-domain errors).

Hungarian (gender neutral)	**English MT output**
ő egy ápoló	she is a nurse
ő egy tudós	he is a scientist
ő egy vezérigazgató	he is a CEO
ő egy esküvőszervező	she is a wedding organizer

Example from (Prates, Avelar, and Lamb 2019).

Bias: Why It Matters for Evaluation

- Potential harms:
  - amplification of gender/social stereotypes,
  - misleading translations in high-stakes domains (healthcare, legal, immigration),
  - erosion of user trust.
- Responsible evaluation can include:
  - bias-aware test sets,
  - targeted challenge cases,
  - uncertainty/confidence estimation,
  - abstention or "don't know" behavior when evidence is insufficient.

Summary

Summary

- MT evaluation is hard because multiple outputs can be valid.
- Human evaluation:
  - adequacy/fluency ratings are intuitive but noisy (ordinal scales, low agreement),
  - ranking reduces scale problems,
  - DA improves reliability with continuous scoring,
  - MQM gives diagnostic error analysis,
  - post-editing aligns evaluation with real effort (HTER).
- Automatic evaluation:
  - surface metrics (WER/BLEU/chrF/TER) are fast but limited,
  - neural metrics (BERTScore/BLEURT/COMET) capture semantics better.
- Significance testing (bootstrap) helps judge whether score differences are real.
- Evaluation must also consider usefulness (task-based) and ethical risks (bias).

Hands-on MT Evaluation

Google Colab

The implementation of the metrics and techniques in this section can be found **HERE**.