Neural Processing Letters (2023) 55:12663–12717
https://doi.org/10.1007/s11063-022-10835-4

A Comprehensive Survey on Various Fully Automatic Machine Translation Evaluation Metrics

Shweta Chauhan[1] · **Philemon Daniel**[1]

Accepted: 7 April 2022 / Published online: 31 May 2022

Abstract

The fast advancement in machine translation models necessitates the development of accurate evaluation metrics that would allow researchers to track the progress in text languages. The evaluation of machine translation models is crucial since its results are exploited for improvements of translation models. However fully automatically evaluating the machine translation models in itself is a huge challenge for the researchers as human evaluation is very expensive, time-consuming, unreproducible. This paper presents a detailed classification and comprehensive survey on various fully automated evaluation metrics, which are used to assess the performance or quality of machine translated output. Various fully automatic evaluation metrics are classified into five categories that are lexical, character, semantic, syntactic, and semantic & syntactic evaluation metrics for better understanding purpose. Taking account of the challenges posed in the field of machine translation evaluation by Statistical Machine Translation and Neural Machine Translation, along with a discussion on the advantages, disadvantages, and gaps for each fully automatic machine translation evaluation metric has been provided. The presented study will help machine translation researchers in quickly identifying automatic machine translation evaluation metrics that are most appropriate for the improvement or development of their machine translation model, as well as researchers in gaining a general understanding of how automatic machine translation evaluation research evolved.

Keywords Machine translation evaluation · Machine translation · Automated metrics · Metrics

✉ Shweta Chauhan
shweta@nith.ac.in

Philemon Daniel
phildani7@nith.ac.in

[1] Department of Electronics and Communication, National Institute of Technology, Hamirpur, Himachal Pradesh 177005, India

1 Introduction

Machine Translation (MT) helps in breaking the language barrier among people and for inter-lingual communication [1]. To check the performance of MT output earlier human judgment was used but that is very expensive, time-consuming, and unrepeatable. Machine Translation Evaluation (MTE) is a very challenging task because natural languages are highly ambiguous and various languages may not necessarily convey the same information in the same way. Since there are no gold standards for translations, the automatic evaluation of the system becomes complicated. At the same time, automatic evaluations are needed for ease of implementation and reusability [2]. Different types of researchers [3–6] have shown the importance of the fully Automatic Machine Translation Evaluation (AMTE), used to compare different translation models and to identify and refine the weaknesses of a model [4]. To provide more consistent, cheaper, and much faster measurements of the performance of MT models, various fully AMTE metrics [7–9] have been proposed in recent years, that compare the human-translated references to the MT outputs in different ways. The purpose of MTE is primarily to evaluate how close the MT output is to human-translated reference.

In recent years unsupervised Statistical Machine Translation (SMT) [10, 11] and unsupervised Neural Machine Translation (NMT) ([10–12) models used only monolingual corpora that make MTE more challenging. Moreover, the improvement in MT reported by unsupervised neural and statistical models [13–18] presents new problems for MTE researchers. The less amount of training data, domain mismatch, rare words, extended sentences, word alignment, and beam search are among the obstacles listed by Koehn and Knowles [19] for NMT and SMT models. Most MT evaluators use Bilingual Evaluation Understudy (BLEU) [20] to evaluate NMT and SMT outputs in these domains. But heuristic-based MTE metrics like BLEU, TER, NIST, etc. are inadequate for capturing the nuances in the various MT tasks. Some of the automatic evaluation metrics rely on linguistic features for their performance, some focus on the lexical feature, but unfortunately, it is not successful for all the language pairs, especially morphologically rich languages, and code-mixed language. Many low resource languages fall under the category of morphologically rich languages. The evaluation of morphologically rich languages is always a challenging task if the translation is from a morphological less complex to a morphological more complex language as the translated output have out of vocabulary and data ambiguity problems.

Whenever the MT researcher updates their translation model, then evaluating the results using a human evaluator hindrance their work. Automatic metrics such as BLEU, ROUGE, METEOR, TER, and others are the most widely used evaluation metrics which quickly evaluate the MT output as compared to the human evaluator. BLEU and other AMTE metrics have remained popular despite multiple researchers [21–25] share criticism revealing that they do not correspond well with human judgments. This is depicted in Fig. 1 which shows the number of citations each year, whereas the dotted lines show the major criticism on these popular evaluation metrics, but which had little effect on their acceptance.

So, from the translator evaluator side, the goal of this review article is to provide a concise presentation of a variety of completely fully AMTE metrics, including information on their implementations, benefits, and drawbacks. However, the survey article presented by Han Lifeng [7], Chatzikoumi [8] provides a review of human and automated metrics but is less detailed for fully AMTE metrics. Another author [9, 26] provides a

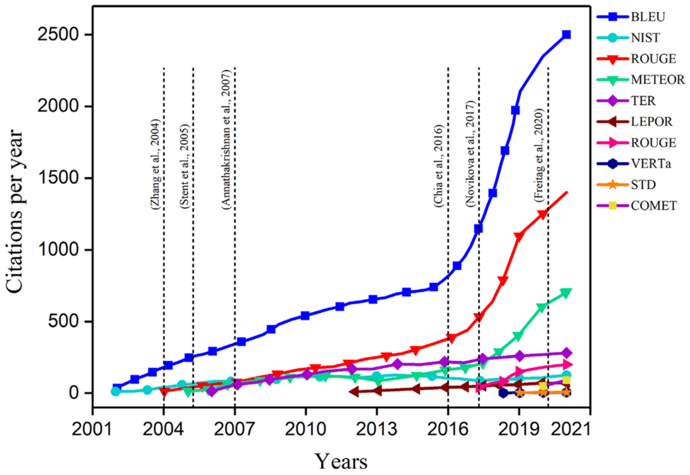

Fig. 1 Popular evaluation metrics citation year-wise and the dotted line shows the criticism of these metrics at the corresponding year of publications

survey of evaluation metrics used for natural language generation systems, but they have not considered the recent deep learning approaches faced by automatic evaluation.

However, in all of the aforementioned literature review study based on fully AMTE metrics is limited, our work is more focused on different classifications of fully AMTE metrics, which cores the detail of each evaluation metric and also discusses key features, advantages, disadvantages, and challenges faced by them. We have also shown the recent challenges faced by neural and statistical MT and their impact in the field of MTE. The impact of morphological rich and low resource language on automatic evaluation has been discussed. Finally, the paper is concluded with research gaps, challenges, and future work.

The main features of the survey paper are:

- The role of evaluation in MT and taxonomy of fully AMTE metrics based on their features that's lexical, character-based, syntactic-based, semantics-based, and finally syntactic and semantic-based evaluation metric has been created in Sect. 2 and discussed in Sect. 3.
- The evaluation properties, modified versions, features, advantages, and shortcomings of each AMTE metric have been summarized in Tables 3, 4, 5, 6 and 7 in Sect. 3.
- The challenges faced by AMTE for unsupervised NMT and SMT models have been discussed in Sect. 4. Experimental results depict the performance of popular AMTE metrics for recent unsupervised NMT and SMT models.
- The impact of language and datasets especially on morphologically and low resource languages for fully AMTE has been analyzed in Sect. 5.
- The upcoming challenges faced by AMTE and the new research ideas/goals for the improvement of evaluation metrics are discussed in Sect. 6.

This paper is organized as: Sect. 2 discusses the role and classification of fully automatic evaluation metrics in detail. Section 3 presented the detailed description of fully Automatic Machine Translation Evaluation Metrics. Language analysis for AMTE metrics is in Sect. 4. Section 5 discusses the challenges faced by automatic evaluation metrics for

NMT and SMT. Section 6 has experimental analysis for popular metrics for recent unsupervised SMT and NMT models. Section 7 discusses the research gaps, challenges, and future work for AMTE. Concluding remarks are in Sect. 8.

2 The Role and Classification of Fully Automatic Machine Translation Evaluation Metrics

2.1 The Role of Machine Translation Evaluation Methods

The role of the MTE in the MT model has been shown in Fig. 2. MT developer has to identify the possible causes of the source of errors in the translations. The MT developer focus on a single or specific subproblem to get the desired solution, then they directly implement it in the translation model. After implementation and training, the main important part is to test the updated model. If the updated models improve the performance, then the mechanism is added to the model otherwise it is discarded. Testing/Evaluation is the most important part of MT models therefore two methods are broadly used that is human evaluation and fully AMTE methods. Although human evaluations are the gold standard but to get reliable judgments, the evaluators should be professional. The manual evaluation performed by the professional translator is time-consuming and expensive. Therefore, to check the quality of MT models AMTE metrics are preferred as they are quick and cheap to use. MTE plays an important role in the improvement of MT models, the three main tasks for MTE methods are:

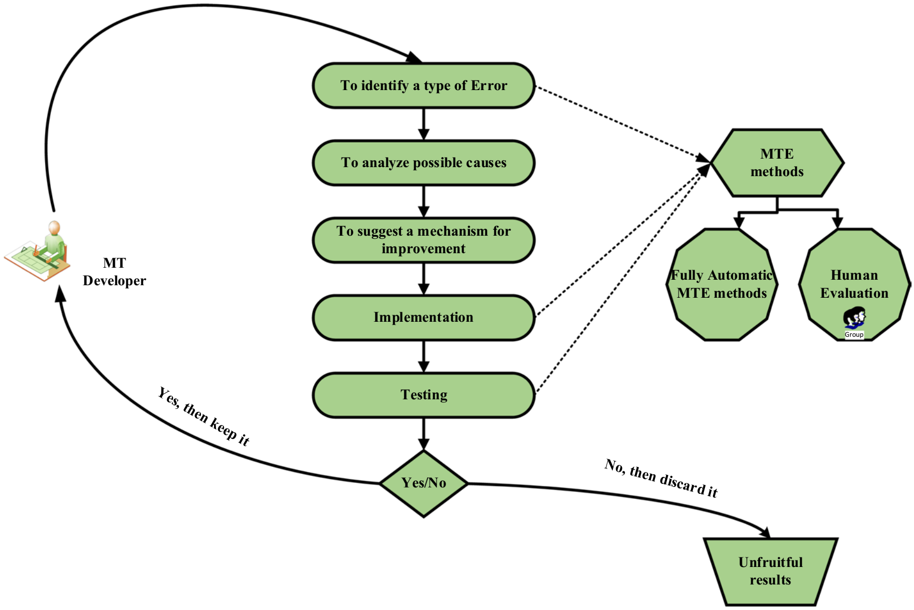

Fig. 2 The role of MTE methods in MT model development cycle

- Error Analysis: MTE helps to analyze the possible cause of the error in the models. Once the errors are defined, the developer finds the solution and implements it in the models.
- MT Model Comparison: MTE methods help to check the performance of the different MT models by comparing different MT model outputs. It will help the MT developer to check and consider the appropriate models in terms of their translation outputs performance, which helps the research community.
- Optimization of the MT model: To get the quality output translation the internal parameters are optimized or adjusted according to the translation quality measured by the MTE methods.

2.2 Classification of Fully AMTE Metrics

Manual evaluation has some drawbacks, such as being time-consuming, expensive, inflexible, and unreproducible. Fully automatic evaluation has been frequently employed for MT models, they usually compare the results of MT models against human translations. We classify automatic evaluation metrics into five groups that are lexical, character-based, syntactic-based, semantics-based, and finally syntactic and semantic-based evaluation metrics as shown in Fig. 3. Still, some MTE metrics share the properties with other groups therefore, deciding their categories is a difficult task. Edit distance, precision, recall, F-measure, and word order are some of the lexical similarity approaches. The character-based evaluation metrics include character n-gram precision and recall. Parts-Of-Speech tags, phrase kinds, and sentence structures are among the syntactic characteristics, whereas named entities, synonyms, paraphrasing, semantic, and language models are among the semantic features. We have discussed the key features, characteristics, advantages, disadvantages, challenges, and research gaps of each evaluation metric.

3 Fully Automatic Machine Translation Evaluation Metrics

3.1 Lexical Similarity-Based Machine Translation Evaluation Metrics

The edit-distance, word-order measure, recall-precision, and n-gram co-occurrence data employed by lexical AMTE metrics are discussed in this section.

3.1.1 Word Error Rate (WER)

The Word Error Rate (WER) [27] is a method for evaluating an MT model performance at the word level. WER judges the quality of translation sentences based upon edit distance. Edit distance is obtained by performing certain edits on the translation sentence so that it matches the reference sentence. The permitted edits are substitutions, deletions, and insertions. Firstly, the recognized word sequences in the translation sentences are aligned with the word sequence in the reference sentences using dynamic string alignment. Further, the edits are applied and WER is calculated using Eq. 1 as:

$$\mathrm{WER} = \frac{\mathrm{Sub} + \mathrm{Del} + \mathrm{Ines}}{\mathrm{N}} \tag{1}$$

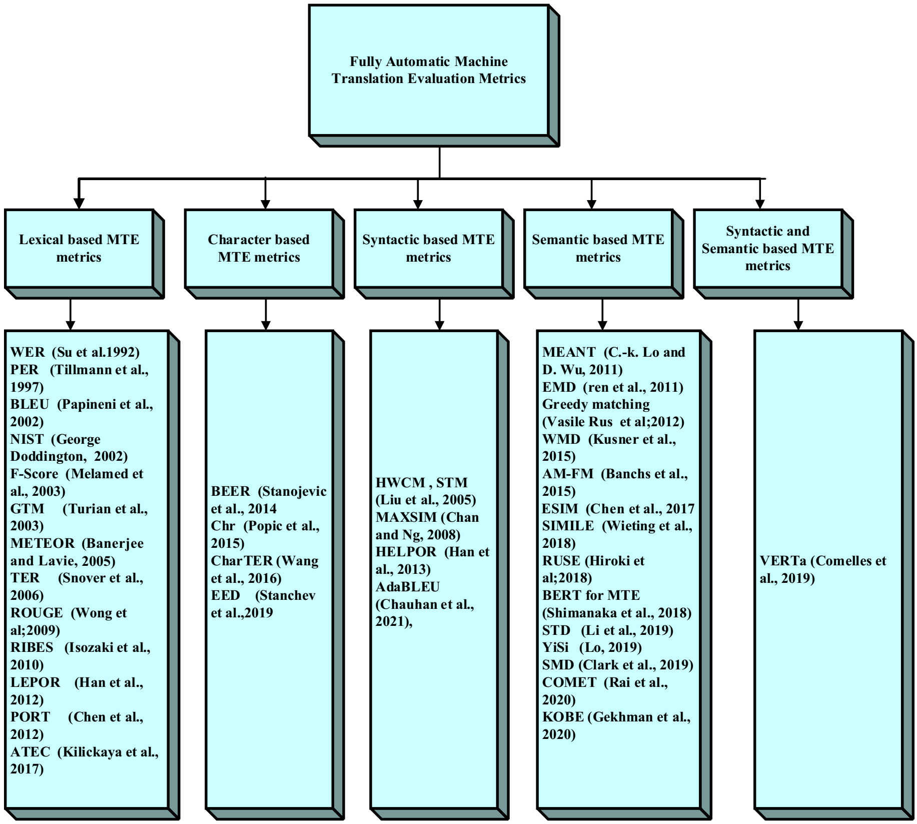
Fully Automatic Machine Translation Evaluation Metrics
Lexical based MTE metrics
Character based MTE metrics
Syntactic based MTE metrics
Semantic based MTE metrics
Syntactic and Semantic based MTE metrics
WER (Su et al.1992)
PER (Tillmann et al., 1997)
BLEU (Papineni et al., 2002)
NIST (George Doddington, 2002)
F-Score (Melamed et al., 2003)
GTM (Turian et al., 2003)
METEOR (Banerjee and Lavie, 2005)
TER (Snover et al., 2006)
ROUGE (Wong et al;2009)
RIBES (Isozaki et al., 2010)
LEPOR (Han et al., 2012)
PORT (Chen et al., 2012)
ATEC (Kilickaya et al., 2017)
BEER (Stanojevic et al., 2014
Chr (Popic et al., 2015)
CharTER (Wang et al., 2016)
EED (Stanchev et al.,2019
HWCM , STM (Liu et al., 2005)
MAXSIM (Chan and Ng, 2008)
HELPOR (Han et al., 2013)
AdaBLEU (Chauhan et al., 2021),
MEANT (C.-k. Lo and D. Wu, 2011)
EMD (ren et al., 2011)
Greedy matching (Vasile Rus et al;2012)
WMD (Kusner et al., 2015)
AM-FM (Banchs et al., 2015)
ESIM (Chen et al., 2017
SIMILE (Wieting et al., 2018)
RUSE (Hiroki et al;2018)
BERT for MTE (Shimanaka et al., 2018)
STD (Li et al., 2019)
YiSi (Lo, 2019)
SMD (Clark et al., 2019)
COMET (Rai et al., 2020)
KOBE (Gekhman et al., 2020)
VERTa (Comelles et al., 2019)

Fig. 3 Taxonomy of fully automatic evaluation metrics for MT

where Sub, Del, and Ines are the number of substitutions, deletions, insertions. Whereas the number of correct words is presented by Matched (M) and N = Sub + Del + M where N is the total number of words that occur in the reference.

Let us take an example the MT sentence is "beautiful car this is" to be evaluated on reference sentence is "This is a beautiful car". Different edits have given different weights, here insertion and deletion are weighted as 0.5 but substitution is taken as 1 (Table 1).

Possible edits are insertion (I), deletion (D), and substitution (S) of single words. Different edits can have different weights. For example, substitution is usually weighted at unity, but deletion and insertion are both weighted at 0.5.

Table 1 WER score computations

Reference output			This	Is	A	Beautiful	Car
MT output	Beautiful	Car	This	Is			
	Insertion	Insertion	Matched	Matched	Deletion	Deletion	Deletion
WER output	$\frac{0.5*2+0.3*2}{2} = 50\%$						

WER is fairly simple to understand, easy to implement, and is reproducible. The word ordering in WER is not taken appropriately. It gives a very poor score when the word order of the MT output sentence is wrong when compared to the reference sentence. WER ignores semantic similarity, which means that many translations can be identified as accurate, but WER only considers one of them to be correct. It also does not consider the syntactic structure of language, therefore, is not suitable for morphologically rich languages.

3.1.2 Position Independent Word Error Rate (PER)

The PER is designed to ignore word order when matching MT output and reference sentences [28]. PER measures the number of times identical words appear in both phrases without considering word order. The rest of the words are either insertion or deletion words, depending on whether the MT sentence is longer or shorter than the reference translation. The PER will always be smaller than or equal to the WER.

$$\mathrm{PER} = \left(1 - \left(\frac{\mathrm{correc} - \mathrm{maximum}\left(0, \mathrm{translated\ Output}_{\mathrm{length}}, \mathrm{Reference}_{\mathrm{length}}\right)}{\mathrm{Reference}_{\mathrm{length}}}\right)\right);$$
$$\mathrm{PER} = \frac{\mathrm{Lost}_{\mathrm{number}} + \mathrm{Reduancy}_{\mathrm{number}}}{\mathrm{Reference}_{\mathrm{length}}} \tag{2}$$

Here in Eq. 2, the $\mathrm{Lost}_{\mathrm{number}}$ is the number of a word only appeared in reference sentences and not present in the MT output sentence. Whereas the $\mathrm{Reduancy}_{\mathrm{number}}$ is the redundancy number, and it is the difference between the reference sentence length and the output length.

There is another different way to remove the word order penalty in the Levenshtein distance by allowing the movement of the word sequence from one side of the output to the other side.

3.1.3 Bilingual Evaluation Understudy (BLEU)

BLEU [20] is still considered the most reliable metric and it is used extensively in the MT community to evaluate the translation quality. The translation output may change with respect to the word choice or word order. This is easily distinguishable by humans but is difficult for machines. Evaluating the translation is also a difficult task without the help of human translators. It is calculated by comparison of n-grams of the MT output with the n-grams of the reference translation and then counting the number of matches. BLEU measures the precision of unigrams, bigrams, trigrams, and 4-g with respect to the whole set of reference translations with a penalty for sentences that are too short. To calculate the BLEU score, one needs multiple high-quality reference sentences, to cover all the possible variations. The number of occurrences of a word in the reference sentences are counted. This is then clipped according to Eq. 3 and then divided by the number of words in the MT output.

$$Count_{clip} = \min(Count, MaxRefCount) \tag{3}$$

Here, Count is the number of occurrences of the word and $\mathrm{Count_{Clip}}$ is the number of occurrences in the reference sentence. We extend this calculation for different n-grams and then find the final precision score using the formula in Eq. 4.

$$p_n = \frac{\sum_{C\varepsilon\{\mathrm{candidates}\}} \sum_{n-\mathrm{gram}\varepsilon C} \mathrm{Count_{clip}}(n-\mathrm{gram})}{\sum_{C'\varepsilon\{\mathrm{candidates}\}} \sum_{n-\mathrm{gram}'\varepsilon C'} \mathrm{Count}(n-\mathrm{gram}')} \tag{4}$$

In Eq. 4 C is the entire set of candidate translations that is machine translation, $\mathrm{Count_{Clip}}$ is the n-grams that match the reference translations. The BLEU also gives a Brevity Penalty to avoid translations that have high precision but are short.

$$\mathrm{Bravity\,Penaty} = \left\{ \begin{array}{ll} 1 & \text{if } r < c \\ e^{(1-r/c)} & \text{if } r \geq c \end{array} \right\} \tag{5}$$

where c is the cumulative length of the set of translated output and r is the set of reference translations.

The final BLEU score is given by Eq. 6 where n is the number of grams.

$$\log \mathrm{BLEU} = \left(\mathrm{Minimum}\left(1 - \frac{r}{c}, 0\right) + \sum_{n=1}^{N} w_n \log p_n \right) \tag{6}$$

Let us take an example the MT sentence “A big Car this is.” to be evaluated on reference sentence is “This is a big car.” In Table 2 unigram precision, bigram, trigram, and then n-gram precision has been calculated and then combined by using geometric averaging using Eq. 4. Now the brevity penalty has been calculated using Eq. 5 if the MT sentence length is smaller than the reference sentence. The final BLEU score is calculated using Eq. 6.

The range of the BLEU score is from 0 to 1. Score 1 refers to a perfect match between reference and translated sentences whereas Score 0 refers to a complete mismatch. To deal with this difficulty, BLEU’s smoothed version uses an add-one strategy, which

Table 2 BLEU score computations

Source Sentence: यह एक बड़ी कार है					
MT sentence: A big car this is					
Reference sentence: This is a big Car					
Unigram precision	A	Big	Car	This	Is
	1	1	1	1	1
	Unigram precision = 5/5				
Bigram precision	A big	Big car	Car this	This Is	
	1	1	1	1	
	Bigram precision = 4/4				
Trigram precision	A big car		Car this is		
	1		0		
	Trigram precision = 1/2				
BLEU score	0.92 (Using Eq. 6)				

demonstrates minimal improvement over the original. Alternative techniques dealing with BLEU issues have been devised.

The main benefits of the BLEU metric are that it's relatively easy to understand and can be implemented without any complexities and it is language independent. The main disadvantage of BLEU is that it's not well-suited for morphologically rich languages due to its strict word matching [21]. Also, BLEU does not consider word order in the sentences as it does not implement recall effectively. BLEU doesn't consider the syntax structure of a language. Due to these factors, BLEU has a low correlation with human judgment.

Because each precision is averaged across all reference sentences, BLEU is referred to as a corpus-level metric, meaning it assigns a score to the entire corpus. Different variations have been proposed over a few years. SentBLEU is a smoothed variant of BLEU that has been demonstrated to better correspond with human sentence judgments. There has been an effort recently to standardize BLEU by matching the tokenization and normalization approaches. sacreBLEU [29] is the name given to this standardized version. Discriminative BLEU, also known as deltaBLEU [30], adds weights to multireference BLEU using the human reference on a scale of $(-1, +1)$. The goal is to reward n-gram matches between the MT sentences and high-quality references while penalizing n-gram matches with low-quality references. As a result, each n-gram is given a weight based on the highest scoring reference in which it appears, which can occasionally be negative.

The Tolerant BLEU [31] uses a distance metric that only needs a precise match in the center of words, not throughout the entire word. LeBLEU use a character-based distance metric.

Self-BLEU is a diversity evaluation metric proposed by Zhu et al. [32], which produces a BLEU score for each created sentence while treating the other generated sentences as references. The text's self-bleu score is the average of these BLEU scores, with a lower self-bleu value indicating greater diversity. According to several NLG articles, self-bleu achieves a high level of generation diversity [32–34]. In the other metrics when it comes to creating varied output [35] or identifying mode collapse in text translation using GAN models [36, 37].

3.1.4 National Institute of Standards and Technology (NIST)

The NIST metric is based on the BLEU evaluation metric but differs in some ways [38]. Instead of n-gram precision, the information gained from each n-gram is considered, implying that the NIST metric determines the importance of each n-gram in relation to the phrase. The importance of each n-gram is obtained with the help of weights. A correct n-gram will be given high weightage if it is quite less likely to occur or rare. The length penalty known as brevity penalty for the NIST score is calculated such that small variations in the length of translation sentences do not affect the overall score significantly. Doddington [38] suggests that the NIST metric with the information weight added to gives more weightage to those n-grams that are more informative.

$$\mathrm{NIST} = \sum_{\mathrm{n}=1}^{\mathrm{N}} \left\{ \frac{\Sigma_{\text{all n}-\text{gram match that matches}} \mathrm{Info}(\mathrm{n}-\mathrm{gram})}{\Sigma_{\mathrm{n}-\mathrm{gram}\varepsilon\mathrm{refernce}}} \right\} .\exp(\beta) \log^{2} \left[\min\left(\frac{|\rho|}{|\bar{\mathrm{r}}|}, 1 \right) \right] \tag{7}$$

where:

$$\mathrm{Info(ngram)} = \mathrm{Info}\left(\mathrm{wi}_1, \ldots \mathrm{wi}_n\right) = \log_2 \frac{\text{number of occurrences of } \mathrm{wi}_1 \ldots, \mathrm{wi}_{n-1}}{\text{number of occurrences of } \mathrm{wi}_{1\ldots,}\mathrm{wi}_n} \tag{8}$$

The number of words in the MT sentences is 2/3rd of the average number of words in the reference sentences, for β the brevity penalty factor is set to 0.5 in Eqs. 7 and 8. The information gain for an *n*-gram made up of words wi_1,..., i_n, is the information gain of n-gram words and has been computed over the reference sentences. Here, $|\rho|$ is the average number of words in a reference sentence. For example, a sentence 'The President', a correct unigram "the" which is fairly common will be given lower weightage as compared to another correct unigram "President" which is a rare n-gram. It uses the arithmetic mean instead of the geometric mean. The main advantages of the NIST metric are, it does not penalize the translation sentences for short length as compared to BLEU. NIST does not evaluate morphologically rich language efficiently as it does not implement semantic similarity and hence, synonyms are undetected. Also, NIST does not consider the syntactic structure of the language effectively and does not consider the word order of the sentences.

3.1.5 Translation Error Rate (TER)

TER [39] is an evaluation measure score that uses the edit distance to improve the lexical matching process. The edit distance is the number of adjustments that must be made to the translation sentence for it to match the reference sentence exactly. Insertion, deletion, the substitution of single words, and shifts of neighboring sequences of words are all possible in TER. All allowed edits have equal weightage. The number of edits required is calculated using dynamic programming that is used to determine edit distance based on the number of insertions, deletions, and replacements. To identify the set of shifts, a greedy search is employed, which involves selecting the shift that minimizes the number of insertions, deletions, and substitutions the greatest until no more favorable shifts exist. Several constraints are applied while finding the optimal set of shifts to get efficient computation. Also, beam search is used to reduce the time complexity. The TER score is calculated in Eq. 9

$$\mathrm{TER} = \frac{\text{Minimum no.of edits}}{\text{Average no. of reference words}} \tag{9}$$

Lower the value of TER better will be the translation quality. TER calculated using Eq. 9 can be larger than 1.0. When there are numerous references, the total number of edits for all of them is calculated, and the lowest score is chosen. TER is easy to understand, and its implementation is simple. TER is not dependent on any specific language. It is fast and cost-effective. The only downside of TER is that it is not efficient for morphologically rich languages due to no inclusion of semantic similarity concepts. To award a more realistic score to an MT output and generate a better alignment against the reference sentence. TER plus [40] enhances the TER measure by using stems, synonyms, paraphrases, and optimizable edit costs.

3.1.6 Recall Oriented Understudy for the Gisting Evaluation (ROUGE)

ROUGE [41] is a set of measures for evaluating automated text summarization and MT. ROUGE evaluates the MT output with respect to a reference sentence. The evaluation is based on a number of factors, including the number of overlapping units like n-grams, word sequences, and word pairs between the computer-generated summary and the reference summaries. ROUGE includes various methods explained below:

- ROUGE-N

ROUGE-N is the automated evaluation metric based upon recall. ROUGE- N compares the MT output to the reference translation in terms of n-gram recall. The number of overlapping n-grams in MT output and reference is known as recall and N in ROUGE-N represents n-gram. For ROUGE-1, the match rate of unigrams is measured. ROUGE-2 and ROUGE-3 use bigrams and trigrams respectively.

ROUGE-N is calculated as per the equation below:

$$\mathrm{ROUGE-N} = \left[\frac{\sum_{l\in(\mathrm{RT})}\sum_{\mathrm{gram_n}\in\mathrm{s}}\mathrm{Count_{match}}(\mathrm{gram_n})}{\sum_{l\in(\mathrm{RT})}\sum_{\mathrm{gram_n}\in\mathrm{s}}\mathrm{Count}(\mathrm{gram_n})}\right] \tag{10}$$

Here n= length of the n-gram, $\mathrm{Count_{match}}(\mathrm{gram_n})$ is a maximum number of n-grams co-occurring in a translated output and a reference sentence as shown in Eq. 10. The numerator of the equation represents the n-gram overlapping between machine translated and reference sentences and the denominator represents the total number of n-grams in reference translation. ROUGE-N can also be implemented with multiple reference translations.

- ROUGE-L

ROUGE-L is the automated evaluation metric based upon Longest Common Subsequence (LCS). Using LCS, ROUGE-L determines the longest matching sequence of words. For two given summaries, the longest common subsequence is the sequence with maximum length. ROUGE-L is calculated using Eq. 11.

$$\mathrm{ROUGE-L} = \frac{(1+\beta^2)\mathrm{R_{lcs}P_{lcs}}}{\mathrm{R_{lcs}}+\beta^2\mathrm{P_{lcs}}} \tag{11}$$

where,

$$\mathrm{R_{lcs}} = \frac{\mathrm{LCS(X,Y)}}{\mathrm{m}};\quad \mathrm{P_{lcs}} = \frac{\mathrm{LCS(X,Y)}}{\mathrm{n}}$$

In Eq. 11 X, Y are two translation of length m and n respectively. It is observed that LCS is more beneficial than n-grams because we need not know the length of n-grams in advance.

- ROUGE-S

The skip-bigram co-occurrence statistics are used to calculate the overlap of skip-bigrams between the MT and reference sentences, which is defined as the overlap of word pairs with a maximum of two gaps between them. For example, the skip-bigrams of a sentence “She goes to school.” would be “She goes”, “She to”, “She school”, “goes to” etc. ROUGE-S is calculated using Eq. 12.

$$\mathrm{ROUGE-S} = \frac{(1+\beta^2)\mathrm{R}_{\mathrm{skip2}}\mathrm{P}_{\mathrm{skip2}}}{\mathrm{R}_{\mathrm{skip2}}+\beta\mathrm{P}_{\mathrm{skip2}}} \tag{12}$$

where

$$\mathrm{R}_{\mathrm{skip2}} = \frac{\mathrm{SKIP2(X,Y)}}{\mathrm{C(m,2)}}\ ,\quad \mathrm{P}_{\mathrm{skip2}} = \frac{\mathrm{SKIP2(X,Y)}}{\mathrm{C(n,2)}}$$

where X, Y are two summaries of length m and n respectively. SKIP2(X, Y) is the number of skip-bigram matches between X and Y. C is the combination function. ß controls the relative importance of $\mathrm{P}_{\mathrm{skip2}}$ and $\mathrm{R}_{\mathrm{skip2}}$. Skip-bigrams make the metric more flexible than n-grams.

ROUGE metric helps in detecting the fluency of MT sentences and works well with single-document summarization. ROUGE provides a variety of techniques that can be quite effective in evaluation tasks. It is fast and easy to implement. But ROUGE does not work well for synonyms. ROUGE is not very useful for morphologically rich languages due to no inclusion of semantic similarity concepts [41]. ROUGE does not work well in the case of multi-document summarization.

3.1.7 Assessment of Text Essential Classification (ATEC)

The ATEC evaluation metric [42] of text-critical characteristics, includes explicit assessments of word order and word choice. In other words, the ATEC metric for machine translation evaluation is based on a matching of word choice. The word order in an MT output in comparison to its reference translations are the two most important variables in the formation of meaning for a phrase. It is based on accuracy and recall metrics, as well as a position difference penalty coefficient that has been designed. The word choice is assessed by comparing word types at many linguistic levels, including surface shape, stem, sound, and meaning, as well as measuring the informativeness of each term. Finally, each MT sentence’s ATEC score is computed as the product of the unigram F-measure and the penalty for word position difference, as shown in Eq. 13

$$\mathrm{ATEC}_{(\mathrm{MTS,RS})} = \mathrm{F-measure}_{(\mathrm{MTS,RS})} \times \mathrm{penalty}_{(\mathrm{MTS,RS})} \tag{13}$$

Here, in Eq. 13 the MTS represents the machine translated sentence and RS represents the reference sentence. The average ATEC score for all the system’s output sentences is the system’s ATEC score. By default, word matching ignores the punctuation and is case insensitive. Originally, ATEC used a two-stage matching technique, the exact match comes first and then the synonym match, with lemmatization using WordNet. However, many acceptable matches are still missed, primarily those words that are not in WordNet, have several parts of speech, or have diverse surface forms despite being phonetically similar or identical.

3.1.8 Length Penalty, Precision, n-Gram Position Difference Penalty and Recall (LEPOR)

LEPORE [43] evaluates MT quality based on a number of parameters including n-gram word order penalty, precision, recall, and sentence-length penalty. The LEPOR measure has been shown good performance on English to other similar language pairs in world machine translation-13. LEPOR series has been constructed from the LEPOR metric. LEPOR combines two changed factors that are sentence length penalty and n-gram position difference penalty as well as two traditional approaches is recall and precision. LEPOR score is calculated by:

$$\mathrm{LEPOR} = \mathrm{LP} * \mathrm{NPosPenal} * \mathrm{Harmonic}^{(\alpha \mathrm{R}, \beta \mathrm{P})} \tag{14}$$

In Eq. 15 LP means Length penalty, which is defined as the penalty for both longer and shorter MT output when compared to the reference translations, and is determined as follows:

$$\text{Length Penalty} = \left\{\begin{array}{ll} \left\{\mathrm{e}^{1-{\mathrm{RS}}/{\mathrm{MTS}}}\right. & \text{if MTS} < \mathrm{RS} \\ 1 & \text{if MTS} = \mathrm{RS} \\ \left\{e^{1-{\mathrm{MTS}}/{\mathrm{RS}}}\right. & \text{if MTS} > \mathrm{RS} \end{array}\right\} \tag{15}$$

As shown in Eq. 15 when the length of the output sentence is equal to the length of the reference sentence, then the output will be 1 which shows that there is no penalty. Here MTS represents the machine translated sentence and RS represents the reference sentence. When the length of the output sentence is greater than or less than the reference length, then the output will be less than one, resulting in a penalty. Mathematically, the exponential function characteristics say that the larger the numerical difference, the smaller will be the value. So, from Eq. 14 NPosPenal is defined as:

$$\mathrm{NPosPenal} = \mathrm{e}^{-\mathrm{NPD}} \tag{16}$$

Here, NPD is the ngram position difference penalty. The value is intended to compare the order of words in sentences between the reference and output translations. The result has been normalized and all MT output can be considered whose effective value lies in the range of 0–1. As a result, the LEPOR value will be lower in the end. According to Eq. 17, NPD is defined as follows:

$$\mathrm{NPD}_{\text{(n−gram position difference penalty)}} = \frac{1}{\text{Total Length}_{\text{output}}} \sum_{\mathrm{i}=1}^{\text{total Length}_{\text{output}}} \left|\mathrm{PD}_{\mathrm{i}}\right| \tag{17}$$

Equation 17 represents the length of the MT output sentence and the n-gram position value of aligned words between MT sentence and reference sentences. Each word from the output translation and reference translation should be aligned. It makes no difference whether the case is upper or lower. When there is no match, the value for this output translation word is set to zero by default. Because a shorter translation sentence may be semantically comparable to the reference sentence, LEPOR imposes a sentence-length penalty, which might lead to erroneous judgment. Because LEPOR cannot effectively assess synonyms and semantically comparable sentences, it has low efficiency for morphologically rich languages.

3.1.9 Precision-Order-Recall Tunable (PORT)

PORT [44] evaluation metric that makes use of precision, recall, and word order information to measure MT quality by comparing MT sentence to reference sentence. PORT does not consume a lot of resources and runs quickly. PORT applies a strict brevity penalty as well as a strict redundancy penalty. Due to the application of this strict length penalty, PORT does not perform well in the case of semantic similarity. Also, PORT is not very effective in detecting synonyms. Precision, recall, strict shortness penalty, strict redundancy penalty, and an ordering measure are the five components of PORT. PORT's architecture is based on extensive testing on a development data set.

Finally, in a harmonic mean, $Q_{mean}(N)$ in Eq. 18 and the word ordering measure are combined.

$$\text{PORT} = \frac{2}{1/Q_{mean}(N) + 1/a^{\alpha}} \tag{18}$$

Here, α is a free parameter that is adjusted based on holdout data. As it increases, the importance of the ordering measures goes up. They focused on Chinese-English data for their studies, setting it to 0.25, and maintained that value for the other language pairings.

3.1.10 General Text Matcher metrics (GTM)

GTM [45] stands for general text matcher and it measures similarity between reference and translated sentences based upon precision and recall, as well as their composite F-measure. It can also be used with multiple references. hence, GTM requires less resources and is quicker to implement. Turian et al. [45] point out models that can manipulate a metric by boosting precision or recall on an individual basis, even when the generations are bad. As a result, the authors propose that the evaluation metric should combine precision and recall, as in F-measure. GTM is an F- score-based metric and has been given which provides more weight to contiguous word sequences that are matched between the translated output to the reference sentences. Here matching is the mapping of words between the translated sentence to the reference sentence is based on their surface forms. The author has computed the weight by the square of the run length (for each run) to award larger weights to consecutive matching sequences referred to as "runs." It's worth noting that the duration of a run can also be one for a single word match. It has to be between 0 and min (|p|, |r|) in general. The matching size of L has been computed using the weights as follows.

$$\text{size}(L) = q\sqrt{\sum_{\text{run}\hat{\text{I}}\,L} \text{length}(\text{run})^{q}} \tag{19}$$

If q has more value, then more weight will have a longer run. The determining the maximum match size for $q > 1$ is NP-hard, GTM employs a greedy approximation in which the biggest non-conflicting mapped sequences are successively concatenated to construct the match. The precision and recall by using approximated maximum match size (MMS) are in Eq. 20.

$$
\begin{aligned}
&\text{Precision(p)}\frac{\text{MMS}(\rho,\text{r})}{|\rho|} \quad ; \ \text{Recall(R)} = \frac{\text{MMS}(\rho,\text{r})}{|\rho|} \\
&\text{GTM} = \ = \frac{2 * \text{Precision} \ * \text{Recall}}{\text{Precision} + \text{Recall}}
\end{aligned}
\tag{20}
$$

GTM provides a high correlation with human judgment as compared to other MT evaluation metrics like BLEU. However, GTM has grown slightly old, and its software support is no longer readily available, limiting its use. Furthermore, GTM is ineffective in morphologically rich languages since it is unable to distinguish semantically identical sentences.

3.1.11 Metric for Evaluation of Translation with Explicit Ordering (METEOR)

METEOR [46] will automatically evaluate the MT output by comparing it with one or more reference translations. METEOR provides a word similarity between the MT sentence and reference sentence produced by a human. The alignment is defined as a set of mappings between the words of the sentence pair where each word in every sentence will map to zero or one to each word of the other reference sentence. Exact matching, stem matching, and synonym matching are the word mapping modules that have been used. Each module maps words that have not been mapped to any word in any of the preceding modules.

The final score of METEOR is computed as follows: firstly, the precision (unigram) is the ratio of the number of matches in the unigram to the total number of a unigram in the reference translation. Secondary similarly the recall (unigram) is the ratio of the number of matches in the unigram to the total number unigram in the reference translation. After obtaining the final alignment, the final score is measured as the harmonic mean of unigram precision or recall as given in Eq. 21. To consider the longer sentences matching METEOR provides a penalty. The unigram is the MT output sentences and references translated are grouped into chunks. If the sentence is longer then there is a fewer chunk. Then the fragmentation penalty is computed as given in Eq. 21. Let's take one example "the Prime Minister spoke to the public" here "the Prime Minister" and "spoke to the public" are two chunks. As the number of chunks is going to be one the fragmentation penalty will be decreased because it is decided by the lower Bond as shown in Eq. 22.

METEOR score is calculated with the help of the following equations:

$$
\begin{aligned}
&\text{Precision} = \frac{\text{No. of matching unigrams}}{\text{Total no. of unigrams in hypothesis}} \\
&\text{Recall} = \frac{\text{No. of matching unigrams}}{\text{Total no. of unigrams in reference}} \\
&\text{F} - \text{Score} = \frac{10{*}\text{Precision}{*}\text{Recall}}{\text{Recall}{+}9{*}\text{Precision}} \\
&\text{Penalty} \ = \ 0.5 \ * \left[\left(\frac{\text{Number of chunks}}{\text{Number of unigrams matched}} \right)^3 \right]
\end{aligned}
\tag{21}
$$

$$
\text{METEOR score} = \ \text{F} - \text{Score} * (1 - \text{Penalty}) \tag{22}
$$

The smallest number METEOR is not well suited for most morphologically rich languages since, aside from English, WordNet has weak support for the majority of languages. The range of scores lies between 0 and 1.

Like BLEU, METEOR has several variations. METEOR-NEXT suggests that the weighted precision and weighted recall allocate weight to various matching conditions like exact, stem, and synonyms matching.

$$\text{Precison} = \frac{\sum_{\mathrm{i}\varepsilon|\{\text{matchers}\}|} \text{weight}_{\mathrm{i}}.\text{matches}_{\mathrm{i}}(\rho)}{|\rho|} \tag{23}$$

$$\text{Recall} = \frac{\sum_{\mathrm{i}\varepsilon|\{\text{matchers}\}|} \text{weight}_{\mathrm{i}}.\text{matches}_{\mathrm{i}}(\mathrm{r})}{|\mathrm{r}|} \tag{24}$$

where $matches_i$ (p) and $matches_i$ (r) show the counts of the mapped words and $weight_i$ are the corresponding weight. Here p represents the precision and R represents the recall. The F-score is calculated as

$$\mathrm{F-Score} = \frac{\mathrm{P}R}{\alpha.\mathrm{P} + (1-\alpha).\mathrm{R}} \tag{25}$$

METEOR is language-specific MTE, whereas the METEOR Universal [47], used by different languages by constructing function-word lists automatically and paraphrasing lists using parallel text in different languages. They describe weighted precision and recall to METEOR-NEXT, but with the added flexibility of weighing content and function words differently:

$$\text{Precison} = \frac{\sum_{\mathrm{i}} \mathrm{w}_{\mathrm{i}}.(\delta.\mathrm{m}_{\mathrm{i}}(\mathrm{hp}_{\mathrm{c}}) + (1-\delta).\mathrm{m}_{\mathrm{i}}(\mathrm{hp}_{\mathrm{f}})}{\delta.|\mathrm{hp}_{\mathrm{c}}| + (1-\delta).|\mathrm{hp}_{\mathrm{f}}|} \tag{26}$$

$$\text{Recall} = \frac{\sum_{\mathrm{i}} \mathrm{w}_{\mathrm{i}}.(\delta.\mathrm{m}_{\mathrm{i}}(\mathrm{re}_{\mathrm{c}}) + (1-\delta).\mathrm{m}_{\mathrm{i}}(\mathrm{re}_{\mathrm{f}})}{\delta.|\mathrm{re}_{\mathrm{c}}| + (1-\delta).|\mathrm{re}_{\mathrm{f}}|} \tag{27}$$

Here in Eqs. 26 and 27 hp_c and re_c represent content words in MT output and reference sentences. Function words are represented by hp_f and re_f and δ, w_i are parameters. These parameters have been adjusted to convey universal human preferences that have been seen across languages. METEOR + + [48] deals with words that are likely to remain the same throughout all parts of a sentence like named entities, or terms with few synonyms. These are used by METEOR + + to see if the MT output is full or inconsistent. METEOR + + 2.0 considers syntactic levels that are not necessarily contiguous.

METEOR-Hindi [49] is a modified version of METEOR for an automatic machine translation evaluation metric where the target language is Hindi. In METEOR's alignment algorithm and the scoring technique, suitable changes are made. METEOR-Hindi extended the METEOR to support the evaluation of translations for Hindi. For adapting METEOR to a new target language, two language-specific components need to be addressed, a language-specific word matching module and language-specific parameters that are Hindi-specific tools need to be used. In this, a Hindi morph analyzer is used for stemming. Synonym matches are detected using synonym sets from the Hindi WordNet.

3.1.12 F-Score

The F-score [50] is often known as the F-measure, which is the measure of accuracy. The harmonic mean of the two metrics balances the precision and recall of the created text. The F1-score is the most common form of the F-score. For the MT or text summarization, the F-score reflects the quality of the created sequence that a model would produce [51]. This score is more effective in measuring the MT quality.

Traditional accuracy and recall measurements, as well as the F-measure, may be used to evaluate MT, and these conventional measures were compared to other contemporary alternative MTE methods. The F-measure is a mixture of recall and precision, firstly it was used in information retrieval and now it is used in information extraction, MT evaluation, and other activities.

3.1.13 Rank-Based Intuitive Bilingual Evaluation Score (RIBES)

RIBES was developed [52] by NTT Communication Science Labs for Asian languages. The RIBES metric is based on word order. It compares the MT output sentences with the reference sentences to use the word order-based rank correlation coefficient. Spearman's measures the distance between differences in rank, while Kendall's measures the direction of differences in rank [52]. First, they use rank spearman coefficients which measure the distance of difference in rank. The second is the Kendall that is used to measure the direction of difference in rank. These rank-based measures can be normalized to get the positive values.

$$\text{Normalized} - \text{Spearman (NSR)}(\rho) = (\rho + 1)/2 \tag{28}$$

$$\text{Normalized Kendall (NKT)}\,(\tau) = (\tau + 1)/2 \tag{29}$$

These metrics can be modified with precision to avoid overestimating the correlation of only corresponding words in the MT and reference translations.

3.1.14 Summary of lexical Based Automatic Machine Translation Evaluation Metrics

Table 3 shows the property, advantages, disadvantages of lexical-based automatic MTE metrics.

3.2 Character-Based Machine Translation Evaluation Metric

The lexical-based evaluation is based on a word-by-word level and character-based evaluation metrics work at the character level. To identify the tokens in the sentence, the character evaluation metrics normally do not require tokenization and work directly on the reference and MT strings. It's worth noting that several of these metrics also make use of word-level information. The enhanced performance of character-based metrics in evaluating morphologically rich languages is the main justification for utilizing them.

Table 3 Lexical based automatic machine translation evaluation metrics

MT evaluation metric	Property of MT evaluation metric	Advantage	Disadvantage
WER Su et al. [27]	Edit distance includes % of substitutions, deletions, and insertions	Simple to implement and consider word order	Poor scores if word order is changed Semantic and syntactic structure does not consider
PER Tillmann et al. [28]	Number of times identical words appear in both phrases measured without considering word order	Simple to implement	Does not allow the sequence of phrases
BLEU Kishore et al. [20]	n-gram precision of the MT output with the n-grams of the reference translation and then counting the number of matches The version of BLEU: **SentBLEU:** The smoothened version of BLEU includes the tokenization and normalization approaches for the evaluation matching **sacreBLEU:** Standardized version expects detokenized MT outputs, applies its own metric-internal preprocessing, and produces the output **DeltaBLEU/Discriminative BLEU:** Added weights to multireference BLEU using human comments **Self-BLEU:** Provide a score for each generated sentence **Tolerant BLEU:** Distance metrics utilize a precise match in the center of words **LeBLEU:** Utilize character-based distance measures	Independent of language and simple to implement Most widely used metric till now	Provides strict word to word Matching Provides equal weightage to all words Does not implement recall efficiently
NIST George Doddington [38]	n-gram weight depends upon its information gain	Improve the BLEU score by rewarding the translation of infrequently Used words	Poorly implement the recall and provide the same weightage to each word

Table 3 (continued)

MT evaluation metric	Property of MT evaluation metric	Advantage	Disadvantage
TER Snover et al. [39]	Utilize edit distance to improve the lexical matching process. The edit distance is the number of adjustments that must be made to the translation sentence	Fast and cost-effective, and not language-specific	Semantic features are not included in the metric, not suitable for morphologically rich languages
ATEC Kilickaya et al. [41]	Explicit assessments of word order and word choice Word choice is assessed by comparing word types at many linguistic levels The product of the unigram F-measure and the penalty for word position difference	Depends upon the word matching and ignores the punctuation, comma, etc	Many acceptable matches are still missed, primarily those words that are not in WordNet
ROUGE Wong et al. [42]	n-gram recall-based measure includes the number of overlapping units like n-grams, word sequences, and word pairs **ROUGE-N** Automated evaluation metric based upon recall **ROUGE-L** Automated evaluation metric based upon Longest Common Subsequence **ROUGE-S** Skipbigram cooccurrence statistics used for evaluation	Contain the property of n-gram recall	Difficult to differentiate between MT output that has different semantic implications Best suited for text summarization rather than machine translation evaluation
LEPOR Han et al. [43]	Several parameters including n-gram word order penalty, precision, recall, and sentence-length penalty included	Combines two factors that are sentence length penalty and n-gram position difference penalty as well as two traditional approaches that are recall and precision	Imposes a sentence-length penalty, leading to erroneous judgment Difficult to assess synonyms and semantically comparable sentences, low efficiency for morphologically rich languages
PORT Chen et al. [44]	Precision, recall, strict shortness penalty, strict redundancy penalty, and an ordering measure property included	Run quickly and less resource are required Precision, recall, and word order information is used to measure translation quality	Strict brevity penalty, as well as a strict redundancy penalty, has been applied Poor for detecting synonyms
GTM Turian et al. [45]	Precision and recall, as well as their composite F-measure	Provide weight for contiguous word sequences	Unable to distinguish semantically identical sentences

Table 3 (continued)

MT evaluation metric	Property of MT evaluation metric	Advantage	Disadvantage
METEOR Banerjee and Lavie [46]	Unigram matching considers the synonyms match Precision and recall and fragmentation penalty for evaluation The version of METEOR are: **METEOR-NEXT:** Weighted precision and weighted recall allocate weight to various matching conditions like exact, stem, and synonyms matching **METEOR-Universal:** It automatically takes features from parallel datasets **METEOR + + :** Words are likely to remain the same throughout all parts of a sentence like named entities, or terms with few synonyms **METEOR + + 2.0:** Syntactic level paraphrase knowledge	Exact matching, stem matching, synonym matching, and word mapping modules are used	Language-specific Wordnet, part of speech, dependency parsing tags, language-specific features required that not available for all languages
METEOR-Hindi Gupta et al. [49]	Applicable for Hindi Translation Utilize Unigram matching and a Hindi-linguistic toolkit	Exact matching, stem matching, synonym matching, and word mapping modules has been used	Required high quality of wordnet and Hindi morphological analyzer
F-Score Melamed et al. [50]	Precision and recall property used for evaluation	Harmonic mean of the metrics balances the precision and recall Easy to implement and language independent	Missing word order
RIBS Isozaki et al. [52]	Word order-based rank correlation coefficient (Spearman, Kendall) Specially designed for Asian languages	Utilize word order-based rank correlation coefficient	More suitable for Asian languages

3.2.1 Character F Score (ChrF)

ChrF [53] is a character-based automatic evaluation metric and it calculates output scores based on the precision and recall of character n-grams (chrP and chR), as well as word n-grams. For character and word n-grams, the F-score is determined. The default n-gram order for characters is 6 and for words, it is 2. The F-score obtained is then averaged to give the ChrF score. ChrF is also helpful in obtaining the semantic similarity of the sentences as it functions with sub-words. ChrF is calculated using Eq. 30.

$$\mathrm{chrF}_{\beta} = (1 + \beta^2)\frac{\mathrm{chrPchrR}}{\beta^2.\mathrm{chrP} + \mathrm{chrR}} \tag{30}$$

where β indicates that recall has a weightage of β times weightage of precision. This metric compares the character n gram from the MT output and reference sentence. For the various value of n the precision and recall that is chrP and chrR are computed. They combined the data using arithmetic averaging. chrP shows the proportion matched character ngram in the MT output sentence, while chrR is the percentage of character ngram in the reference which is included in the MT output sentence considering the n = 1,2,3.0.6. ChrF was created to test MT models, but it has been used for various applications. The improved versions of chrF [54] include word n-grams as well as character n-grams.

3.2.2 Character Translation Error Rate (CharacTER)

This MTE metric [55] inspired by the TER metric is a character-level metric. CharacTER starts with word-level shift edits, using a lenient matching criterion that considers the word in the MT output sentence and matches with a human-translated reference sentence if a value is less than a minimum threshold value for the character-based edit distance. The Levenshtein distance is determined between the shifted MT output sentence and the reference sentence. Furthermore, normalizing by reference sentence length ignores the MT output sentence length. characTER employs the MT output sentence length to normalize the edit distance. This normalization has been demonstrated to have a greater correlation than the human-based judgments.

3.2.3 Better Evaluation as Ranking (BEER)

BEER [56] is another character-based automatic evaluation metric that uses only linear function while taking some input features. BEER obtains input features using recall, precision, and F1-score for character n-grams and word unigrams. To evaluate word order, permutation tree-based features are used. The features are obtained both for reference and MT sentences. The following equation gives the final evaluation score.

$$\mathrm{BEER} = \sum \mathrm{w_i} * \mathrm{x_i}(\mathrm{MT},\ \mathrm{RT}) \tag{31}$$

Here in Eq. 31 $\mathrm{x_i}$ is the similarity score between the machine translated and reference sentences and weights $\mathrm{w_i}$, are learned using the linear regression model.

Table 4 Character-based automatic machine translation evaluation metrics

MT evaluation metric	Property of MT evaluation Metric	Advantage	Disadvantage/future work
Chr Popic et al. [53]	Precision and recall of character n-grams (chrP and chR), as well as word n-grams For character and word n-grams, the F-score is determined. The default n-gram order for characters is 6 and for words, it is 2	Consider word unigrams and character n-grams	Investigation methods are required for carrying different weights for a particular n-gram
CharTER [55]	Word-level shift edits, using a lenient matching criterion considers the word in the MT output sentence	Levenshtein distances are used to determine between the shifted MT output sentence and the reference sentence	Levenshtein distance has been used in many evaluation metrics
BEER Stanojevic et al. [56]	Character n-gram uses F1 score precision and recall	Linear function while taking some input features	More linguistic features should be investigated that will improve the performance of the evaluation metric
EED Stanchev et al. [57]	Cover distance evaluation rate used jump operation for deletion, substitution, and insertion at the character level	Jump operation used, to get the edit distance from a different place or location, helpful for MT output that has a different word order Bound to be between 0 to 1	More meaningful analysis is required on an extension of Levenshtein distance as there are many metrics based on it Investigation of shift and jump relationship required

3.2.4 Extended Edit Distance (EED)

This metric [57] is based on the cover distance evaluation rate and adds a jump operation to the traditional edit operations that are insertions, deletions, and substitutions, but only at the character level. The main function of the jump is to get the edit distance from a different place or location, and it will be helpful for MT output who has a different word order. This procedure is only allowed on blank space characters, however, to avoid jumps. EED can be written as in Eq. 32.

$$\mathrm{EED} = \mathrm{minimum}\left(\frac{(e + \alpha.j) + \rho.v}{|r| + \rho.v}, 1\right) \tag{32}$$

Here the word e indicates the cost of traditional edit operations that include insertion and substitution having a uniform cost of 1 and deletion having a cost of 0.2. Here the value of alpha is one. In Eq. 32 the total number of jump operations is determined by j to cover the penalty term that is v. The score is kept within the range of zero, one, and that will restrict the number of times the minimum function picks the value 1. The r is the coverage penalty term added in the denominator.

3.2.5 Summary of Character-Based Automatic Machine Translation Evaluation Metrics

Table 4 shows the property, advantage, disadvantage, or future work of character-based automatic MTE metrics.

3.3 Syntactic Similarity-Based Machine Translation Evaluation Metrics

These metric measures the structural similarities between an MT output sentence and reference sentence and also determine the grammatical sentence structure similarity. Parts-of-speech tagging assigns a tag that is noun, verb, part of speech, dependency parsing, etc., to each word based on its context, morphological behavior, and syntax in corpus linguistics.

3.3.1 Maximum Similarity (MAXSIM)

The MAXSIM [58] metric computes the similarity scores between a pair of sentences (based on precision and recall). Maximum weight matching can be achieved by matching an item with only one item and not more than two items. That will help to provide comparison objects using arbitrary similarity functions and to include a variety of data such as n-grams, dependency links, and so on.

If weighted matches have been utilized, there might be many alternative methods to match an MT sentence to reference sentences. For the matching of each MT sentenced to the one of reference sentence, the items in the sentence pair are modeled as nodes in a bipartite graph and different technique is used to find a maximum weight matching between the items in polynomial time.

Another approach [59] used a maximum-weight bipartite graph. The MXPOST tagger Parts-of-speech tagging is used, as well as MAXSIM tokenization, WordNet lemmatization. The Fmean scores are obtained by matching the unigrams, bigrams, and trigrams in the phrase pair. To generate a single similarity score for this phrase pair s, the average of the three Fmean scores is taken. Then repeat the scoring method for each model reference sentence pair, averaging the results overall |S| sentence pairings to create a single sim-score for the whole system corpus.

$$\text{simscore} = \frac{1}{|S|}\sum_{s=1}^{|s|}\left[\frac{1}{N}\sum_{n=1}^{N}F_{\text{means, n}}\right] \tag{33}$$

In Eq. 33 the N = 3 has been taken to represent the calculation of unigram, bigram, and trigram scores. They calculate each similarity score between the MT output sentences and human-translated reference sentences, after that average of the results has been taken if there are more than one references sentences.

3.3.2 Harmonic Mean of Enhanced Length Penalty, Precision, n-Gram Position, Penalty, and Recall (HELPOR)

For morphological complex language HELPOR [60] uses POS tags similarity to capture syntactic information. The sentence-level and segment level score are calculated as the mathematical harmonic mean to group multi-variables is:

$$\text{Harmonic}(x_1, x_2, \dots x_n) = \frac{n}{\sum_{i=1}^{n}\frac{1}{x_4}} \tag{34}$$

Here, the number of factors is shown by n and the weighted harmonic mean for different variables is given by:

$$\text{Harmonic}\ \ (wx_1X_1, wx_2X_2, \dots, wx_nX_n)\ = \frac{\sum_{i=1}^{n} wx_s}{\sum_{i=1}^{n}\frac{wx_s}{x_i}} \tag{35}$$

where wx_iX_i is the weight of variable Xi. Finally, the developed evaluation metric HELPOR is used to get the sentence level score:

$$\text{HELPOR} = \frac{w_{ELP} + w_{NPosPenal} + w_{HPR}}{\frac{w_{\text{Enhaced length penalty}}}{\text{Enhaced length penalty}} + \frac{w_{\text{N-gram position difference penalty}}}{\text{N-gram position difference penalty}} + \frac{w_{\text{harmonic precision and recall}}}{\text{harmonic precision and recall}}} \tag{36}$$

where ELP (Enhaced length penalty), NPosPenal (N-gram position difference penalty) and HPR (harmonic precision and recall) are the three factors with tunable weights W_{ELP}, $W_{NPosPenal}$ and W_{HPR} respectively. Here MTS is machine translated sentence and RS is reference sentences.

$$\text{Enhanced Length Penalty (ELP)} = \begin{cases} e^{1-RS/MTS} & : \ MTS < RS \\ e^{1-RS/MTS} & : \ MTS \geq RS \end{cases} \tag{37}$$

$$\mathrm{NPD} = \frac{1}{\mathrm{Length}_{\mathrm{output}}} \sum_{i=1}^{\mathrm{Length}_{\mathrm{output}}} e^{-\mathrm{NPD}}$$

HPR is the Harmonic precision and recall. The system-level score is the arithmetical mean of the sentence scores are given in Eq. 38.

$$\mathrm{HELPOR} = \frac{1}{\mathrm{SentenceNumber}} \sum_{i=1}^{\mathrm{SentenceNumber}} \mathrm{HELPOR}_{I} \tag{38}$$

Here, SentenceNumber represents the sentence number of the reference text document, and HELPOR_i means the score of the ith sentence.

3.3.3 Headword Chain Based Metric (HWCM), Subtree Metric (STM), and Tree Kernal Based Metric (TKM)

The Author [61] presents three metrics that use syntactic structure and unlabeled dependency information. It computes the similarity between output and reference using the subtree kernel with the use of syntactic information like constituent labels.

To get a clear and easy evaluation of a sentence's fluency, syntax trees are utilized to generate metrics based on the closeness of the MT output tree and those of the references. This technique is based on subtree fractions since the full syntax tree of the MT output can always be found in the references.

In this work, they use the arithmetic mean rather than the geometric mean because of the sparse subtrees that lead to zero fractions. The percentages of subtrees with different depths are found for each MT phrase, and then their arithmetic mean is determined as the syntax metric, referred to as "subtree metric."

$$\mathrm{SubtreeMetric} = \frac{1}{D} \sum_{n=1}^{D} \frac{\sum_{t \in \mathrm{subtrees}_n(\mathrm{hyp})} \mathrm{count}_{\mathrm{clip}}(t)}{\sum_{t \in \mathrm{subtrees}_n(\mathrm{hyp})} \mathrm{count}(t)} \tag{39}$$

Here in Eq. 39 D denotes the maximum depth of subtrees. However, count(t) denotes the number of times subtree t appears in the MT output sentence syntax tree. $\mathrm{count}_{\mathrm{clip}}(t)$ is the clipped number of times t appears in the references sentence. The maximum number of times a subtree appears in any one reference's syntax tree cannot be exceeded by the count obtained from the MT output syntax tree for that subtree. By using the kernel approach, they can compute the cosine similarity without having to compute the complete vector of counts H. The maximum of the cosine measure across the reference sentences is defined as TKM, a kernel-based subtree metric.

$$\mathrm{TKM} = \max_{t \in \mathrm{refence}} \mathrm{cosine}(\mathrm{MT}, t) \tag{40}$$

The use of a tree kernel has the advantage that it captures the similarity of subtrees of various forms. The one flaw is that it can only utilize the reference trees one at a time, whereas STM can use them simultaneously. HWCM utilizes the same calculation technique as BLEU, but instead of n-grams in the sentence's linear order, it uses n-grams of dependency chains. The average of the percentages of headword chains that also appear in the reference dependency trees determines the final score for each MT output phrase.

$$\mathrm{HWCM} = \frac{1}{\mathrm{D}} \sum_{\mathrm{n=1}}^{\mathrm{D}} \frac{\sum_{\mathrm{g \in hyp}} \mathrm{count_{clip}(g)}}{\sum_{\mathrm{g \in hyp}} \mathrm{count(g)}} \tag{41}$$

Here D is the maximum chain length taken in Eq. 41. The author [62] calculates the similarity by using syntactic structure between the MT output sentence and reference sentence. The author [63] proposes another method by introducing the decomposition of WER and PER over different Part-of-Speech (POS) classes for obtaining more accurate details about actual translation errors in the generated output. Another metric [64] is based on sequence and tree kernel but it is complicated because it mostly depends on existing syntactic knowledge of languages. This is especially true when dealing with a little-known and headword chain-based metric complex language.

3.3.4 Adaptive Bilingual Evaluation Understudy (AdaBLEU)

To overcome the shortage of strict string matching of BLEU [65, 66] an MT evaluation score is AdaBLEU which considers the lexical and syntactical properties of the words and is useful for morphologically rich languages. AdaBLEU is a modified BLEU score, and it also does not require multiple reference sentences for evaluation. Our metric, AdaBLEU considers the Parts-of-Speech (POS) tags and Dependency Parsing (DP) tags along with the BLEU score of sentences. The proposed score covers the usage of synonyms and any resulting syntactic changes through POS and DP tags. Automatic evaluation metric AdaBLEU is an indirect measure of translation quality because it covers the usage of synonyms and any resulting syntactic changes through POS and DP tags to measure the closeness between the MT system outputs and the manually offered reference translations and is based on the calculation of correlation score with manual MT evaluation.

Just like BLEU, it evaluates the n-grams of the POS tags and DP tags. This helps in generating a decent score even if only some parts of the sentences are translated properly. For this, they evaluate the BLEU score on the POS tags and DP tags generated for the translated sentence. POS tags help with the lexical and semantic analysis and DP tags with the syntax analysis. This score helps gain information on the sentence structure and framing, but still, an unrelated verb may be used in place of the required verb. A higher weightage is given to the POS tags score. For the final score, the weighted average between the POS tag and DP tag scores and the BLEU score is considered. The following formula is used for calculating AdaBLEU score.

$$\mathrm{AdaBLEU} = \frac{W_1(BLEU(POS)) + W_2(BLEU(DP)) + W_3(BLEU(Sentence))}{W_1 + W_2 + W_3} \tag{42}$$

- BLEU (POS) means BLEU of a string of Part-of Speech Tags of the sentence.
- BLEU(DP) means BLEU of a string of Dependencies Parsing Tags of the sentence.
- BLEU(Sentence) means BLEU score on the sentence.
- W1, W2, W3 are the weights.

The weights are decided such that the correlation of AdaBLEU and human evaluation is high. They employed the Pearson correlation coefficient to measure the correlation between BLEU, AdaBLEU, and human judgment. The human evaluation is tested on two test datasets of 150 sentences of respective languages. The evaluation metric requires a

Table 5 Syntactic based automatic machine translation evaluation metrics

MT evaluation metric	Property of MT evaluation metric	Advantage	Disadvantage
MaxSIM Chen et al. [33]	Maximum weight matching can be achieved by matching an item with only one item and not more than two items A variety of data in analysis, such as n-grams, dependency parsing, and many others	Maximum weight bipartite graph, uses supervised models for single word alignment between sentences from source to the target language Contain MXPOST tagger Parts-of-speech as well as MAXSIM tokenization, WordNet lemmatization	Semantic part is not included which reduces the performance of the metric Less optimized the parameter
Helpor Han et al. [60]	Unigrams harmonic mean has been used	POS tags similarity to capture syntactic information for morphologically rich languages	Semantic information, using the distance between item pairs can also be added
HWCM, STM Liu et al. [61]	Property of syntactic structure and unlabeled dependency information It computes the similarity between output and reference using the subtree kernel with the use of syntactic information like constituent labels	Arithmetic means rather than the geometric mean has been used because of the sparse subtrees that lead to zero fractions	Missing semantic features
AdaBLEU Chauhan et al. [65, 66]	Lexical and syntactical properties of the words are useful for morphologically rich languages Automatic evaluation metric AdaBLEU is an indirect measure of translation quality because it covers the usage of synonyms and any resulting syntactic changes through POS and DP tags	POS and DP tags along with the BLEU score of the sentence has a better correlation	Difficult to analyze all the synonyms and may wrongly give a high score for words with different semantic meanings for the same POS and DP tag

tagger and parser systems for POS tags and DP tags posing a restriction on the metric due to language dependence. Also, they use POS tags to address the issue of strict matching. But this may not accurately consider all the synonyms and may wrongly give a high score for words with different semantic meanings but the same tag. To better address, the issue of strict matching use of synonym identifiers is required. The author performed metrics on Hindi, Urdu, and Telegu languages.

3.3.5 Summary of Syntactic Based Automatic Machine Translation Evaluation Metrics

Table 5 shows the property, advantages, disadvantages of syntactic-based automatic MTE metrics.

3.4 Semantic and Word Embedding Based Machine Translation Evaluation Metric

The lexical and character-based evaluation metrics rely heavily on surface-level matching, but some consider synonyms, so, they frequently overlook semantic similarities among the words. If we consider the words ’canine’ and ’dog,’ they are synonyms and related to each other. Similarly, the words "pig" and "dog" are much nearer (due to their status as domestic animals) but are not synonymous when compared with the word "dog" and "boat". Word embeddings like Word2Vec are better at capturing such commonalities. Word2Vec [67], GloVe [68] has trained on a very large dataset that will compare and capture the distributional similarity between words. As an alternative to matching words, the similarity of the MT and reference word embeddings are compared.

3.4.1 Semantic Travel Distance (STD)

Recently, Semantic Travel Distance [69] (STD) has been devised based upon a novel document representation and a semantic distance matrix consisting of n-gram word embeddings as it captures semantic similarity features of the languages. This leads to an accurate evaluation as semantics plays a significant role in morphological language structure. Firstly, the reference and the MT sentences are transformed into vector representations respectively. A novel document representation has been proposed which is normalized similarity representation. nSIM representations are better than nBOW as nSIM can better recognize the words semantically related in the reference and MT sentences. The distance between the MT output sentence and reference sentence is measured by using the earth mover’s distance solvers based on the weighted graph.

For segment level:

$$\text{Semantic distance } = 0.5 * \text{Semantic}_{\text{unigram}} + 0.5 * \text{Semantic}_{\text{bigram}} \tag{43}$$

For system level:

$$\text{Semantic distance } = 0.3 * \text{Semanticdis}_{\text{unigram}} + 0.7 * \text{Semanticdise}_{\text{bigram}}$$

STD score is obtained in the range of (0,1) and the lower the value of STD, the better is the translation quality.

$$\text{Where Semanticdis}_{\text{unigram}} = \sum_{i,j} F^{\text{opt}}_{\text{ij\_unigram}} d_{\text{ij\_unigram}}$$

$$\text{Semanticdis}_{\text{bigram}} = \sum_{i,j} F^{\text{opt}}_{\text{ij\_bigram}} d_{\text{ij\_bigram}}$$

STD metric is efficient in evaluating the morphologically rich languages as it not only captures the semantics of sentences but also the word order. The only shortcoming of STD is that syntactic features, which are also integral feature in morphologically rich languages, are not included in its evaluation process. Higher quality of word embedding will improve the STD score. Moreover, it only depends on the availability of word embeddings, which should be available, or at least derivable for most languages.

3.4.2 Cross-Lingual Optimized Metric for Evaluation of Translation (COMET)

In earlier approaches, the quality of MT depends upon the similarity between the MT output sentences to the human translated sentences. COMET [70] is based on the Pytorch framework for customizable, multilingual MTE models that can be used as metrics. This study uses recent advances in cross-lingual language modeling [71, 72] to create predictions of human-based evaluation such as Direct-assessments (DA), human-mediated translation edit rate (HTER) [73].

In this work, they demonstrated that a high level of correlation with human judgment can be achieved without any reference translation. For this, they proposed an approach that will incorporate source language input into the MTE model.

Previously the source input has been used by the quality model while the MTE metric depends upon the reference translation [74]. COMET framework trained models with different types of human judgment that are direct assessments, multidimensional quality, and human-mediated translation edit rate. It uses a novel technique to estimate MT quality by combining information from both the source input and the target-language reference translation and shows a better correlation at a segment level.

3.4.3 Enhanced Sequential Inference Model (ESIM)

This metric [75] is based on contextual word embeddings [76, 77] that captures rich and portable representations of words in context and has shown an important signal to a variety of other natural language processing tasks. When comparing a reference to an automatic translation, they propose a simple untrained model that uses off-the-shelf contextual embeddings to compute approximate recall, as well as trained models, such as a recurrent model over reference and translation sequences that incorporate attention, and the adaptation of a natural language processing method to MT evaluation. In other words, the ESIM model is directly used for the purpose of MT evaluation. It uses a BiLSTM model that has been trained to compute sentence representations for the reference and MT. A cross-sentence attention mechanism is then used to calculate the similarity between the reference and MT sentence.

The improved representations are sent into a second BiLSTM. The final BiLSTM's max-pooled and average-pooled hidden states are utilized to forecast the ESIM score:

$$\text{ESIM} = \text{UReLU}(\text{Wx} + \text{b}) + \text{b}' \tag{44}$$

Here Eq. 44 × shows the average pooled vector of the final BiLSTM hidden states for sentence b, and U,W, b' are parameters to be learned. ReLU is the activation function.

3.4.4 Yisi

Yisi [78] method evaluates the correctness of MT model output, which is based on the intended score. It is based on shallow semantic structures and weighted distributional lexical semantic similarity. To measure lexical similarity, it takes the largest substring from the MT output sentence and reference sentence. YiSi [78] is a single semantic assessment system that brings together a number of metrics, each of which is tailored to languages with varying levels of resources.

YiSi-1 is a fully AMTE that will measure the similarity between MT sentences to the reference sentences by aggregating the weighted distribution lexical semantics similarities.

YiSi-0 is a degenerate resource-free variant that will measure the only word similarity between machine-translated sentences and the reference sentences by considering the longest common characters substring.

YiSi-2 contains the parallel and bilingual word embedding for the evaluation of cross-lingual lexical semantics similarity between MT output sentences to the reference sentences.

$$\text{YiSi} = \left(\frac{\text{Precesion} * \text{Recall}}{\alpha * \text{precesion} + (1 - \alpha) * \text{recall}} \right) \tag{45}$$

Here in Eq. 45, the weight of Alpha is set to 0.7 to make YiSi more recall oriented. But, when used for MT models resolution the optimization Alpha is taken as 0.5 so, that there is a balance between precision and recall.

3.4.5 *Automatic Evaluation Metric for Evaluating Translation Utility* via *Semantic Frames (MEANT)*

MEANT (C.k. [79]) and its variants (C.k. [80], C.k. [81]) use the semantic frame for MT output sentences to the reference sentences. The metric depends on the linguistic tools for its MT evaluation and is not applicable to every language to capture the semantic similarity. The author proposes semantic role labeling or shallow semantic parsing the process of assigning words and phrases in the sentences. This type of annotation would help in improving sentence similarity. The approach is the weighted combination of F-score and word vectors are used to calculate lexical similarity.

A different variation of MEANT 2.0 [82] has been proposed. This variation will use the inverse document frequency to provide the weight of each for keeping in mind that are phrases that have more context words has the highest scores. In the phrasal similarity, computation is changed to n-gram lexical similarity rather than the bag of words in the phrases. The vector-based similarity computes the lexical and structural similarity by using word embeddings. Another cross-lingual variant of MEANT is XMEANT (C.k. [81]) which is based on a semantic frame based on fully AMTE metric which correlates better because they use a cross-lingual objective that deeply integrates semantic frame criteria into MT training.

3.4.6 Adequacy and Fluency: A Semantic Framework for Evaluation (AM-FM)

The AM-FM [83] is a semantic framework for evaluating translation quality without the use of reference translations. The suggested framework is theoretically based on the ideas of sufficiency and fluency, and it is intended to account for these two aspects of translation quality separately. First, the adequacy component is assessed using a cross-language latent semantic indexing model that compares the output translation to the original text it was derived from. Second, the fluency component is assessed using an n-gram based language model of the target language. The measure's two components are evaluated at the sentence level, allowing for the definition and implementation of a sentence-based evaluation metric. Finally, the two components are integrated into a single measure using a weighted harmonic mean, with the weighting ratio adjustable for metric performance optimization. The cross-language latent semantic indexing approach is used to build the metric's adequacy-oriented component (AM), in which the source sentence generating the translation is employed as the evaluation reference. Because it is computed on a cross-language semantic space, the AM component can be considered primarily adequacy-oriented. An n-gram based language model technique is employed to create the proposed metric's fluency-oriented component (FM). Since it is computed on the target language side like that it is completely independent of the source language, this component can be considered primarily fluency-oriented. A weighted harmonic mean is offered for merging both components into a single metric:

$$\text{AM - FM} = \text{AM} * \text{FM}/(\text{AM} + (1 - \text{FM})) \tag{46}$$

Equation 46 can be modified to set the maximize correlation between the suggested metric AM-FM and human evaluation scores, ranging from 0 for AM component and 1 for the FM component. Two different AM-FM variants are implemented and assessed in this paper. The upgraded version describes a monolingual implementation that only works with the target language. For the adequacy-oriented component of the measure, a monolingual continuous space model is employed to assess translation outputs against translation references. This creates a more level playing field when comparing the proposed statistic to other metrics that heavily rely on translation references.

3.4.7 Semantic Similarity Evaluation (SIMILE)

SIMILE [84] is the semantic similarity between the MT output sentences to the reference sentences evaluated by the embedding models that are trained on the large size of paraphrase data. The main aim is to get a continuous metric for sentence similarity, here semantics textual similarity and embedding base module are used. Semantic textual similarity provides the similarity score between MT sentences by correlating with human judgment whereas in the case of the embedding model the final score is the cosine similarity between two sentences. SIMILE simply preferred on four different languages and the results are shown by optimizing during training gives an improvement in the same metric.

They use the max-margin loss to train a sentence encoder, g, using a collection of paraphrase pairs (from the ParaNMT corpus) to promote improved comparison of MT and reference sentences:

$$l(s, s') = \max(0, \delta - \cos(g(s),\ g(s')) + \cos(g(s),\ g(t)) \tag{47}$$

where δ is the margin, s and s' are paraphrases, and t is a negative example obtained by random sampling the other sentence pairs.

The semantic similarity is represented by SIM is the cosine similarity of the MT output sentence to the reference sentence. Here length penalty term penalized the MT output sentences if there is a length difference between an MT output sentence to the reference sentence.

$$LP(r,p) = e^{(1-\max(|r|,|p|)/(\min(|r|,|p|)))} \tag{48}$$

The final equation of SIMILE is the combination of LP and SIM given in Eq. 49.

$$SIMILE = LP(r,p)\alpha SIM(r,p) \tag{49}$$

Here the length penalty term is denoted by α.

3.4.8 Knowledge-Based Machine Translation Evaluation (KoBE)

Reference-based evaluation metric required the identical matching of reference translation with MT output sentences as in lexical metrics such as BLEU, TER, NIST. But in the case of KoBE [85] evaluation method is based on the multilingual knowledge base evaluation by taking the linkage in entities of source sentences to the MT sentences. The recall of the entities is taken in MT output sentences with reference sentences. In this approach different words of the language share the same meaning and that can be solved by the same type of entry in the knowledge base. Since the knowledge base method does not penalize in the different right translation of the same entry. This approach outperforms on different types of nine languages of WMT-19 benchmarks. This method is beneficial as an absolute quality signal rather than a relative one because it gauges the recall of the things identified in the source phrase. Here, google-knowledge graph search API3 has been used by the author that considers the entities from different perspective areas.

As single matches, entities are only counted in the source side many times, and after that, these matches are trimmed or cut according to the number of times each entity appears in the source. In the early stage or the preprocessing stage inbuilt language identifications, feature or tool is set up which will discard those entities in the MT sentence which is not part of the target language sentence. The advantage of this approach is that we can improve the outcomes at a very initial stage. Now in Eq. 50 the recall is defined as the ratio of the total number of different matched entities across the different sentence pairing and then combined by the total entities in the source sentences.

$$\text{Recall} = \frac{\sum_{\text{i}=0}^{\text{n}_\text{i}} |t\text{otal matches (sourceentities, total entitie}s)|}{\sum\limits_{\text{i}=0}^{\text{n}_\text{i}} |\text{entities(source entities)}|} \tag{50}$$

The MT entities were rejected that were not in the correct language. The entity linking system was detecting the cloned entities because it is language agnostic, resulting in erroneous matches. Precision has a smaller link as it rewards systems that output fewer entities systems and gives lower-quality translations. Because the number of entities in the source is fixed for all assessed systems and only the match count changes, recall is more stable. The author has given an entity count penalty that is taken by the BLEU brevity penalty evaluation score. This entity count penalty is taken because sometimes recall gives inflated results when over-producing entities. Here the number of entities m in the source is almost double, then Entity Count Penalty (ECP) will penalize the model that will produce the entities:

$$\text{Entity Count Penalty} = \begin{cases} 1 & \text{if } m < 2s \\ e^{(1-m/2s)} & \text{if } m \geq 2s \end{cases} \tag{51}$$

Finally:

$$\text{KoBE} = \text{ECP} * \text{recall} \tag{52}$$

Because metric depends upon the recall of things on the target side as compared to the source side, the entities in the target can be correctly detected. This metric has a problem in that it does not handle sentence semantics.

3.4.9 Regressor Using Sentence Embeddings (RUSE)

RUSE [86] is a regression model based on the Multilayer Perceptron (MLP) that combines three pre-trained sentence embeddings. InferSent [87], Quick-Thought [88], and Universal Sentence Encoder [89] are the three types of sentence embeddings employed. Using a mix of the MT and reference sentence embeddings, the multilayer perceptron predicts the regressor RUSE score.

$$\text{Sentence embedding} = (\text{InferSent}; \text{Quick} - \text{Thought}; \text{UniversalSentenceEncoder}) \tag{53}$$

$$\text{RUSE} = \text{MLP} - \text{Regressor}\,(p; r; |p - r|; p * r) \tag{54}$$

Here p is the embedding of the machine translated sentence and r is the embedding of the reference translation sentence. Pre-trained models are used for sentence embedding and MLP regressor is trained by using human judgment from the WMT shared tasks.

3.4.10 Bidirectional Encoder Representations from Transformers (BERT) for MTE

The lexical-based evaluation metrics like BLEU and ROUGE show a weak correlation with human judgment. This approach encodes the [90] concatenated all the human-translated sentences. Instead of using different sentence embedding the MT output and reference sentences are pair-encoded together.

Bilingual Evaluation Understudy with Representations from Transformers (BLEURT) [91, 92] is the evaluation metric based on BERT. Since it is a pretraining approach and outperforms when the training dataset is in a very limited amount, it uses the fully unsupervised representation. For evaluation, the BERTScore uses pre-trained embeddings from BERT and matched the words from the MT output to the reference sentences. But these pretrain embeddings are not available for every language like low resource languages.

3.4.11 Greedy Matching (GM) and Embedding Average Metric (EAM)

These metrics [93] use the cosine similarity between the token's embeddings, and it will measure the match between the reference and MT token. The final evaluation score is calculated by averaging all the tokens in the reference sentence. The score in the greedy matching is direction dependent. To ensure that the metric is symmetric, the operation is repeated in the other direction. As indicated in Eqs. 55 and 56, the final score is the average of matching in both directions.

$$\mathrm{G(p,r)} = \frac{\sum_{\mathrm{w}\in \mathrm{r}} \max_{\mathrm{w}^{\wedge}\in \mathrm{p}} \mathrm{cosine}(\overrightarrow{\mathrm{w_r}}, \overrightarrow{\mathrm{w_p}})}{|\mathrm{r}|} \tag{55}$$

$$\mathrm{GM} = \frac{\mathrm{G(p,r)} + \mathrm{G(r,p)}}{2} \tag{56}$$

Here w is word embedding and GM is greedy matching. Here p is the MT token whereas r is the reference token.

The final score is the average of matching in both directions as shown in Eq. 55. In EAM the word embeddings of the sentences can be directly computed and compared for the reference translation and MT sentences while in earlier approaches the word embedding score is calculated for the MT with reference. The embedding average metric measure the sentence level embedding s by taking the tokens of all the word embeddings. The score for an MT sentence, Embedding Average metric (EAM) is then computed as the cosine similarity between the embedding of the reference word embedding r and the word embedding of the MT.

$$\mathrm{EAM} = \mathrm{cosine}\left(\vec{\mathrm{p}}, \vec{\mathrm{r}}\right) \tag{57}$$

The cosine similarity between the reference sentence embedding and machine translated embeddings will determine a score.

3.4.12 Word Movers Distance (WMD), Earth Movers Distance (EMD), and Sentence Movers Distance (SMD)

The word mover's distance [94] is a variant of the Earth mover's distance [95] that will help in the calculation of the distance between two sequences and are individually represented by word frequencies. It blends word embedding similarity with item similarity on the bag of word histogram representations of the text. In conclusion, WMD exhibits a number of intriguing qualities as it has no hyperparameters and is simple to use. It makes use of the information embedded in the word embedding space, resulting in high retrieval accuracy.

WMD has helped to improve a variety of NLG tasks, particularly sentence-level tasks. The word movers' distance is suitable for small texts, its cost rises as the length of the documents grows longer, and the bag of words technique can be problematic since the relationship between sentences is lost as documents grow larger. The metric can't capture information conveyed in a collection of words because it simply measures word distances.

Sentence Mover's Distance is a fully automatic metric and is based on the WMD for the evaluation of the text in a continuous space using sentence embeddings [96]. Each text is represented by SMD is the collection of sentences with each sentence embedding weighted based on its length. The cumulative distance of relocating the sentence embeddings in one document to match the sentence embeddings in another document is measured by SMD. In a summarization challenge, SMD outperformed ROUGE in human evaluations [96]. Zhao et al. [97] suggested an advanced version of SMD that correlates better with human evaluation. They used word and phrase embeddings, similar to SMD, by averaging the token-based embeddings before calculating the mover's distance.

Table 6 Semantic-based automatic machine translation evaluation metrics

MT evaluation metric	Property of MT evaluation metric	Advantage	Disadvantage
STD Li et al. [70]	n-gram word embeddings property has been utilized to capture semantic similarity features of the languages Earth mover's distance solvers based on the weighted graph	Semantics sentences and the word order evaluate efficiently for morphologically rich languages Higher quality of word embedding improves the STD score	Syntactic features are not included in the evaluation process
COMET Rai et al. [69]	PyTorch-based framework for customizable, multilingual MTE models	Source input is not required for MT evaluation	Cross-lingual language modeling has been utilized to create predictions of human-based evaluation
ESIM Chen et al. [75]	BiLSTM model trained to compute the sentence representations for the reference and MT sentences Cross-sentence attention mechanism calculate the similarity between the reference and MT Attention-weighted representations are integrated to provide improved representations of the MT output and reference sentence	Contextual word embeddings that capture rich and portable representations of words in context	Syntactic features like Part of speech, Dependency parsing tags are not included

Table 6 (continued)

MT evaluation metric	Property of MT evaluation metric	Advantage	Disadvantage
Yisi Lo [78]	Shallow semantic structures and weighted distributional lexical semantic similarity has been utilized Version of YiSi are: **YiSi-0:** Degenerated resource-free variant will measure the only word similarity between machine-translated sentences and the reference sentences by considering the longest common characters substring **YiSi-1:** The similarity between MT sentences to the reference sentences is measured by aggregating the weighted distribution lexical semantics similarities **YiSi-2:** Parallel and bilingual word embedding for the evaluation of lexical semantics-based similarity between machine translated sentences to the reference sentences	Single semantic assessment system that brings together a number of metrics, each of which is tailored to languages with varying levels of resources	Lexical similarity, measure the largest substring from the MT output sentence and reference sentence
MEANT C.-k. Lo et al. [81]	Semantic frame or labeling assigns the words and phrases for MT output sentences to the reference sentences	Weighted combination of F-score and word vectors	Linguistic tools are not easily available to every language to capture the semantic similarity vector-based similarity

Table 6 (continued)

MT evaluation metric	Property of MT evaluation metric	Advantage	Disadvantage
AM-FM Banchs et al. [83]	Weighted harmonic mean, and cross-language latent semantic indexing to build the metric's adequacy-oriented component. Source sentence generated the translation is employed as the evaluation reference AM component uses adequacy-oriented because of cross-lingual semantic space. An n-gram based language model technique is employed to create the proposed metric's fluency-oriented component	Adequacy and fluency properties	Some parameters are required to be adjusted to get the maximum correlation
SIMILE Wieting et al. [84]	Embedding models that are trained on the large size of paraphrase data For continuous metric sentence similarity, semantics textual, similarity embedding module has been used	Utilize higher quality of word embeddings like BERT and Elmo	Syntactic similarity is not included
KoBE Gekhman et al. [85]	It is based on multilingual knowledge base evaluation by taking the linkage in entities of source sentences to the MT sentences Different words of the language share the same meaning and are solved by the same type of entry in the knowledge base	Outcomes improved at a very initial stage	Poor in handling sentence semantics
RUSE Hiroki et al. [86]	Regression model based on the Multilayer Perceptron that combines three pre-trained sentences embedding Infer Sent, Quick-Thought, and Universal Sentence Encoder are the three types of sentence embeddings employed Multilayer perceptron predicts the regressor RUSE score	Pre-trained models are used for sentence embedding and MLP regressor is trained by using human judgment	Embedding trained on larger dataset provide effective results rather than smaller datasets

Table 6 (continued)

MT evaluation metric	Property of MT evaluation metric	Advantage	Disadvantage
BERT Shimanaka et al. [90]	Encodes the concatenated reference sentences together. Instead of using different sentence embedding, they use pair-encoded	The pretraining approach outperforms when the training dataset is less and utilizes fully unsupervised representation	Difficult to handle long text sequences
Greedy matching (GM), embedding average metric (EAM) Rus et al. [93]	Cosine similarity between the token's embeddings, will measure the match between the reference and MT token Final score is the average of matching in both directions	Direction-dependent, the process is repeated in the other direction to verify that the metric is symmetric	Higher quality of word embedding required
WMD Kusner et al. [94]	Minimal cumulative distance calculated between the embeddings of their component words	WMD exhibits several intriguing qualities, as it has no hyperparameters and is simple to use Information embedded in the word embedding space, resulting in high retrieval accuracy	Not applicable for longer document because the bag of words technique can be problematic since the relationship between sentences is lost as documents grow larger
EMD Ren et al. [95]	Calculate the distance between two sequences and individually represent by word frequencies	EMD exhibits a number of intriguing qualities and tuning of hyperparameters is not required and is simple to use It makes use of the information embedded in the word embedding space, resulting in high retrieval accuracy	More suitable for small texts
Sentence Moves Distance (SMD) Clark et al. [96]	Each text is represented as the collection of sentences with each sentence embedding weighted based on its length Cumulative distance of relocating the sentence embeddings in one document to match the sentence embeddings in another document is measured by SMD	Cumulative distance of relocating the sentence embeddings in one document to match the sentence embeddings in another document	Problematic for the longer document because of the bag of words technique since the relationship between sentences is lost as documents grow larger

3.4.13 Summary of Semantic-Based Automatic Machine Translation Evaluation Metrics

Table 6 shows the property, advantages, disadvantages of semantic-based automatic MTE metrics.

3.5 Semantic and Syntactic Machine Translation Evaluation

In semantic and syntactic evaluation metrics the automatic evaluation metric has semantic and syntactic features.

Only verta is a fully automatic evaluation metric that requires these linguistic features.

3.5.1 VERTa: A Linguistic Approach to AMTE

VERTa [98] is a fully AMTE metric that is based on the linguistic features or information at the morphological, semantic, and synthetic levels that compares the MT output sentences with reference sentences.

VERTa has different models that are lexical, morphological, dependency, n-gram, semantic, and language models. These different modules show the different types of evaluation like fluency, adequacy, ranking, and sampling. The final score is the combination of the Precision and Recall of each module in the F-mean and that correlates better with human judgment.

- The lexical module has the function of comparison the lexical features of MT output to reference sentences. This approach is inspired by METEOR and only considers the lexical-semantic relationship. The important part is that VERTa contains the information of lemmas, where METEOR only contains stemming.
- The morphological module takes the combination of morphological information and lexical features with Parts-of-Speech tags from the annotated dataset. The main aim of this module is that it matches the words. For example, the word "love" and "loves" both have a similar meaning, but they are different in their morphological information. So, this module will check the fluency of the segment as compared to its adequacy.
- Dependency Module gives the similarity between the equivalent sentences and that shows different constituents order. This module lies on those matches that are established at the lexical level that is the word form and synonymy, lemmas, and partial lemma.
- N-gram module matches the chunks in the reference segments to the MT sentence. It can rely on the matches of the lexical and morphological modules.
- Semantic Module will play important role in MTE for adequacy. This module contains the other information at the lexical and sentence level. They also include named entity recognition and named entity linking. The language model module uses the reference segment to calculate the log probability so that the MT sentences are compared to what occurs in the corpus while building the language model.

The first module has two alignment ways, firstly it employs the matches set in the module, and secondly, it combines both morphological modules that are Parts-of-Speech and the lexical module. This sort of alignment depends on the type of MTE used and for the matching process greedy method is used. There are different ways for better alignment and

Table 7 Semantic and syntactic based automatic machine translation evaluation metrics

MT evaluation metric	Property of MT evaluation metric	Advantage	Disadvantage
VERTa Comelles et al. [98]	Based on Linguistic features or information at the morphological, semantic, and synthetic levels that compares the MT output sentences with reference sentences	Precision and recall weightage utilize number of matches like word, n-grams, and dependency parsing	Language specific feature are not available for low resource language

that can be implemented by using adjustable metric implementation except for the language model. The precision and recall utilize a weightage over a number of matches with each element like word, n-grams, and dependency that is shown in Eq. 58.

$$\begin{aligned} \text{Precision} &= \frac{\sum_{\mathrm{r}\in \mathrm{D}} \mathrm{w_r} \cdot \mathrm{nmatch_r}(\Delta(\mathrm{MTS}))}{(\Delta(\mathrm{MTS}))} \\ \text{Recall} &= \frac{\sum_{\mathrm{r}\in \mathrm{D}} \mathrm{w_r} \cdot \mathrm{nmatch_r}(\Delta(\mathrm{RS}))}{(\Delta(\mathrm{RS}))} \end{aligned} \tag{58}$$

Here, the function r will return the elements of each level that's words at the lexical level, and the set of different types of matching is given by D. Here MTS is machine translated sentence and RS is reference sentence. Lastly, the collection of weights that is from 0 to 1 is associated with each type of matching which is used to combine different types of matching. VERTa combines Precision and Recall metrics with the F-mean. If there are many references, the score is calculated using the highest F-mean of all references. The final score is a weighted average of the different module scores (F-mean) when the scores per module are calculated.

3.5.2 Summary of Syntactic and Semantic-Based Automatic Machine Translation Evaluation Metrics

Table 7 shows the property, advantages, disadvantages of semantic and syntactic-based automatic MTE metrics.

4 Languages Analysis for AMTE

AMTE metric used to check the quality of various MT output, but unfortunately, it is not successful for all the language pairs, especially morphologically rich languages. Mostly Endangered languages fall under the category of morphologically rich languages. The high resource languages like English, French, German have a linguistic tool that is easily publicly available and that makes evaluation much easy. Moreover, they are trained on large

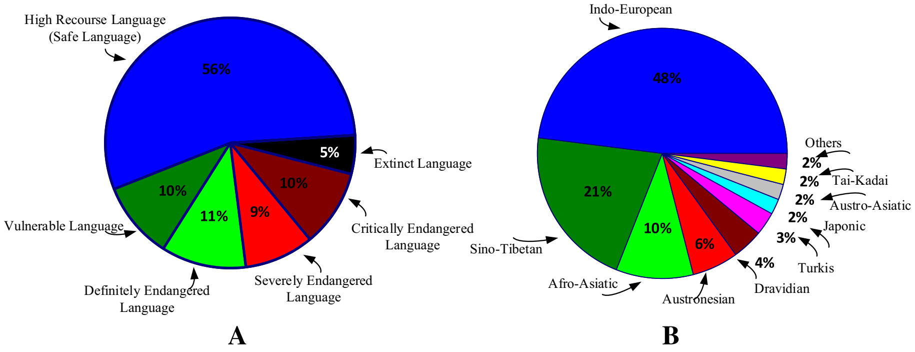

Fig. 4 **A** Endangered language around the world. **B** World Language speakers

corpora, so the output does not contain untranslated words. But low resource languages like kangri, mandyali, chambali faces out of vocabulary problems, data ambiguity problems because they do not have a sufficient amount of corpora and that makes evaluation a tough task. Polyglot and code mixture language output contain missing words and ambiguity output. So, the selection of metrics plays a crucial role in evaluation.

In today's world, the number of speakers of some languages is in billions, while it is only a few thousand for many languages. UNESCO listed around 7000 languages around the world around which 4000 languages are safe whereas 3000 languages are endangered languages as shown in Fig. 4A. The endangered languages are categorized as vulnerable, definitely endangered, severely endangered, critically endangered, and extinct languages. The evaluation of the low resource endangered language is challenging because the MT output contains out of vocabulary and data sparsity problems. These endangered languages do not have proper linguistic tools that are easily available.

Figure 4B shows the languages that are spoken around the world in which Indo-European, Sino Tibetan, Dravidian languages datasets are available publicly. The evaluation of these languages is much easier because they have a larger vocabulary, dictionary, and larger training dataset and linguistic tools are easily available. BLEU is still a widely used automatic evaluation metric, but it has a problem with morphologically rich language. BLEU, TER NIST ROUGE and recently COMET has been widely used evaluation metric besides their criticism. Moreover, the evaluation of the current unsupervised MT system has been challenging because of ambiguity issues that it cannot capture the complexity of words with multiple meanings, such as homonyms or polysemous words in the case of morphologically rich languages. This limitation can be overcome by word sense disambiguation which identifies the correct sense of a given word in a particular sentence. Many techniques were used in word sense disambiguation that is applicable for all languages. Whereas in case of low resource languages out of vocabulary problem is the main challenges due to data sparsity. So, data augmentation techniques have been used to increase the dataset of low resource languages and to remove out of vocabulary problems different techniques have been used.

- *Problematic Morphologically Rich Languages for AMTE*

The automatic evaluation of unsupervised MT models has been challenging especially for morphological rich and low resource languages as they have word ambiguity issues. Morphologically rich languages are problematic in MT and AMTE, especially if the translation is from a morphological less complex to a morphological more complex language. Endangered low resource languages in general seriously suffer from problems like lack of sufficient parallel data and morphological richness as well as word order difference due to syntactic divergence. Morphological distinctions which are not present in the source language need to be generated in the target language. Much work on morphology-aware approaches relies heavily on language-specific tools, which are not always available to each language. For example, in Hindi, considering the synonyms "पुस्तक" and "किताब", and other semantically similar words such as "पुस्तक" and "पुस्तकें", such pair of words having same meaning but expressing it differently occur frequently. The syntactic differences contribute to the difficulties for translation models, the morphological differences contribute to data sparsity. Highly inflected language and agglutinative languages are the types of morphologically rich languages. In highly inflected language there is the change of the form of some form of words when used in a sentence. Latin, Finnish languages fall in this category. While in agglutinative language words are made up of a linear sequence of distinct morphemes and each component of meaning is represented by its morpheme like Kannada, Estonian and Hungarian languages.

5 Neural and Statistical Challenges Faced by MT Evaluation

The improvement in MTE for neural and statistical models [55, 99–102] presents the new problems for MTE. The author [103] demonstrated that the actual language of the source material, the evaluator's translation proficiency, and the context are the main key factors that were not considered by Hassan et al. [104]. The author [105] has proposed the MTE a challenge set of sentences, that is a set of phrases containing the linguistic features. This is specifically designed to face the challenge of the NMT systems, in this framework. Sennrich et al. [106] has focused on linguistically intriguing phenomena that are to be problematic for MTE for a long-distance agreement, like transliteration, and polarity.

To evaluate statistical MT and neural MT models, [107] the author proposes a fine-grained manual evaluation based on the multidimensional quality metrics. This taxonomy has been adapted to the linguistic phenomena of the languages and this is used in the MTE projects as of the strategy.

Because of the extensive feedback on the language events involved, their findings suggest that this type of metric can capture the relevant elements that pose issues to NMT evaluation.

NMT and SMT systems suffer issues like, including as scaling to larger word vocabularies and slow training speed of models. Furthermore, to train NMT and SMT systems to get good performance a large amount of corpus is required. Different obstacles for NMT have been presented by Koehn and Knowles [19] that will include the mismatching of the domain, rarest word, long and extended sentences, beam search, the word alignment, and the total amount of training and test dataset for NMT. Mostly, the BLEU evaluation metric has been used to evaluate NMT outputs in these domains, but they customize the task for each domain, by changing the training circumstances of the models.

As a successor to SMT, NMT has been evolved and it has advanced rapidly and is already making inroads into the MT industry. NMT, SMT is based on the deep learning method of MT and will employ a massive neural network based on word vector representations. Since neural networks share statistical evidence between comparable phrases, the prediction potential of NMT is more promising than that of SMT. There are still major translation problems, such as mistranslated words or named entities, omissions, and incorrect word order, in current NMT systems. Additionally, neural machine translation inconsistency between training, inference, and its evaluation is a major issue in most sequence creation jobs. This subject has been effectively addressed in the machine translation community [108, 109] but it is still worth investigating, particularly in terms of technical efficiency. Designing NMT that is both comprehensible and durable. The SMT and NMT model is like a black box and using it in many cases where we need to know how and why the translation result is generated is quite problematic. The contribution of each input to the output translation was visualized in Ding et al. [110]. Nonetheless, it would be fantastic to dig more into the NMT models explanations or develop an explainable MT architecture. Furthermore, contemporary NMT systems are vulnerable to input perturbation. Prism [111] provides a large, pre-trained multilingual NMT model. They [112] provided publicly a large collection of MT output sentences human judgement. Their main objective is to find an AMTE metric that is best suited for the pairwise ranking of different models.

Table 8 Machine translation evaluation score for unsupervised NMT model

Language pair	BLEU	NIST	ROUGE	METEOR	TER	AdaBLEU
English-Hindi	0.298	0.05	0.204	0.08	0.09	0.299
English-Telegu	0.192	0.01	0.172	0.02	0.07	0.199
English-Tamil	0.131	0.03	0.120	0.01	0.06	0.132
Hindi-Kangri	0.01	0.001	0.001	0.000	0.001	0.02

Table 9 Machine translation evaluation Score for unsupervised SMT model

Language pair	BLEU	NIST	ROUGE	METEOR	TER	AdaBLEU
English-Hindi	0.418	0.09	0.214	0.10	0.08	0.420
English-Telegu	0.212	0.12	0.191	0.11	0.12	0.214
English-Tamil	0.202	0.10	0.170	0.11	0.10	0.234
Hindi-Kangri	0.05	0.01	0.01	0.00	0.01	0.07

The NMT models to handle noisy inputs were presented in Chen [108, 109]. However, real-world input noise is very difficult to predict, and designing strong NMT and SMT models that are immune to real-world noise remains a major difficulty.

6 Experimental Analysis

In this section NMT, SMT Translation Model with their training choice setup, dataset, and results are provided for different evaluation metrics.

- *NMT, SMT Translation Model*

    - Shared encoder with back translation based Unsupervised NMT system [10, 11]
    - Unsupervised phrase-based Moses SMT System [10, 11]

First model has encoder with a bidirectional recurrent neural network (RNN) is shared between two decoders. Training is performed between denoising and back-translation. The second model uses the standard SMT model has three subtasks namely language model (LM), translation model (TM), and, decoding. The TM and LM are built on the Bayesian rule. The usual phrase based SMT (PBSMT) approach infers word alignment first from a bilingual corpus, after which phrase pairs are generated and a statistical table with their translation probability is created. To extract phrase pairs, a distortion model is used. The usual PBSMT approach infers word alignment first from a bilingual corpus, after which phrase pairs are generated and a statistical table with their translation probability is created. To extract phrase pairs, a distortion model is used.

- Unsupervised NMT Training Model Setup

We employed an encoder and decoder with two layers, 600 hidden units each. Bath size is set to 50. Encoder has bidirectional RNN, whereas the decoder has RNN (two layer). Adam optimizer has been utilized and learning rate is 0.0002. SoftMax activation function has been utilized. Dropout regular probability is set at 0.3. We trained all three systems with a fixed number of 3,00,000 iterations.

- Unsupervised SMT Training Model Setup

Trained the PBSMT model given by the Moses toolbox [113]. FastAlign is used to align the source language segmented words with the target language segmented words. Symmetrized heuristic and Grow-diag-final for the alignments used KenLM [114] to train the 5-g Language model with modified Kneser–Ney discounting. The Moses de- coder (v0.4) was used to decode the data, and the Minimum Error Rate Training (MERT) was utilized to optimize the decoder settings. Moses default settings were utilized in all of the experiments.

- Dataset and Language Pair

The dataset has been taken from World Asian Translation-2020 (WAT-2020). For training, the AI4Bharat [115] corpus (English, Hindi, Telegu, Tamil) and Kangari low resource endangered language [65, 66] has been taken. We have compiled test datasets of 1000 sentences for respective languages. We then translated them on respective NMT and SMT translation models.

- Results for Different AMTE Metrics

The performance of some popular MT evaluation metrics BLEU, NIST, TER, ROUGE, METEOR, AdaBLEU has been evaluated for three high resource Indian languages and one low resource Kangri language shown in Table 8. From Tables 8 and 9 we conclude that the AdaBLEU evaluation metrics provide the highest correlation among all lexical-based evaluation metrics for unsupervised NMT and SMT models. AdaBLEU evaluation metrics evaluate the translation by considering the POS tags and DP tags that help in covering the lexical and syntactical properties. It avoids the strict string to string matching of lexical-based evaluation metrics. Among all the evaluation metrics BLEU and its variant evaluation metrics are still popular. For low resource language, the evaluation score for unsupervised NMT and SMT models is low because the output sentences have untranslated words or out of vocabulary words. Here low score does not mean that evaluation metrics perform poorly, it only evaluated the MT output with reference sentence. If the quality of reference sentences is good and multiple references have been provided, then fully AMTE metrics provide better results.

In general, the role of fully AMTE metrics is the assessment of MT output with reference sentences. If the MT is for similar languages e.g., English-to-French, they are high resource languages and have a large amount of quality dataset. Therefore, the Lexical-based evaluation metrics like BLEU, TER, NIST will provide better correlation. The MT is performed for morphologically low resource language to morphologically rich language, then semantic or syntactic based MTE metrics provide better correlation. MTE also depends upon the quality of the reference sentence. If the quality of reference sentences is good and multiple references have been provided, then most of the fully AMTE metrics will evaluate

Table 10 Sample sentences with different evaluation scores

Sr. no	Source sentence	Reference sentence	Translated sentence	BLEU	METEOR	TER	NIST	ROUGE	AdaBLEU
1.	I have not finished the task yet.	मैंने अभी काम ख़तम नहीं किया है।	मैंने अभी तक काम पूरा नहीं किया है।	0.48	0.12	0.19	0.24	0.02	0.73
2.	Why did you come early.	आप इतनी जल्दी क्यों आ गए	तुम इतने जल्दी क्यों आ आया	0.43	0.02	0.30	0.21	0.08	0.87
3.	I hope Mowgli learned something from that experience.	ఆ అనుభవం నుండి మోగ్లీ ఏదో నేర్చుకున్నాడని నేను నమ్ముతు	నేను మోగ్లీ ఆ అనుభవం నుండి ఏదో నేర్చుకున్నాడు అనుకుంటున్న	0.75	0.08	0.10	0.11	0.07	0.95
		న్నాను							
4.	I am excited about the summer, too.	నేను వేసవి గురించి కూడా సంతోషిస్తున్నాను.	నేను చాలా, వేసవి గురించి సంతోషిస్తన్నాము చేస్తున్నాను.	0.84	0.01	0.06	0.13	0.07	0.90
5.	मगर दहमाचली भाषा तो पहले से बोली जा रही है।	अपर म्हाचली भासा तां पैहले ते बलोंदी औआ कर दी।	भासा म्हाचल <oov> तां <oov> आ।	0.01	0.00	0.01	0.01	0.01	0.02
6.	आम आदलमयों का जीना न्याय लेना खाना-पीना और रहना बैठना मुक्ककल हो गया है।	आम आदमी दा जीण, न्यां लैणा, खाणा पीणा कनैं रैणा बौहणा मुश्कल होई गेया ऐ।	<oov> जीण, <oov> पीण <oov> रैणा <oov> गेया ऐ।	0.01	0.00	0.01	0.00	0.01	0.03

MT model translated sentences better. The MT is performed for low resource languages which have less dataset, and their training is by unsupervised learning method. The linguistic features like wordnet, POS, DP tags, etc. are available then semantic, syntactic-based evaluation metrics correlated better with human evaluation for low resource language. Otherwise, the lexical-based evaluation metric will provide a better result.

- Sample sentences

In Table 10, we report and discuss some sample sentences which show the performance of popular evaluation metrics.

- In sentence 1, The morphological aspect of Hindi has been considered. The translated and reference sentences have several different words which are synonyms. Other metrics do not consider synonyms extensively and hence, have a lower correlation. AdaBLEU scores the sentence accurately as it is semantically and syntactically correct up to some extent provide better correlation.
- In sentence 2, based on the variation in pronouns आप and तुम, BLEU and other metrics rate the sentence lower score as compared to AdaBLEU which covers the variation and. AdaBLEU covers the difference in the usage of words इतने instead of इतनी in translation and reference sentence respectively.
- In sentence 3, the translated output is correct semantically as the words, నమ్ముతున్నాను and అనుకుంటున్న in reference and translation sentences respectively, are synonyms. Hence, the AdaBLEU score is closer to human evaluation than other metrics.
- In sentence 4, the word చేస్తున్నాను in translation has no semantic context, but the remaining translation sentence is accurate. Other metrics penalize the score due to the unknown word while AdaBLEU does not penalize strictly and hence, it is closer to human evaluation.
- In sentences 5 and 6 many words are missing in the translated sentence, so evaluation metrics provide low correlation.

7 Research Issues, Challenges, and Future Directions Faced by AMTE Metrics

7.1 Research Issues

- Automatic evaluation metrics are indirect measures of translation quality (Moran and Lewis 2011). This Fully AMTE uses the different string-matching distance algorithms that measure the closeness between the MT outputs and the human reference translations. These metrics calculate the correlation scores with the human evaluated MTE.
- The existing fully AMTE metrics do not bother about the relevance of the words in the sentences. Most of the fully AMTE metrics give an equal weightage to the punctuation and other symbols whereas more weightage should be given to named entities and parts-of-speech tags. Some of the AMTE metrics do not consider the same meaning of words and provide a low evaluation score.
- The existing fully AMTE metrics sometimes produce a meaningless score for the translated sentences, for example, the score—05,234 produced by the MTeRater metric in Parton et al. [116] and 112. 98 scores by ROSE [117] does not convey any information.
- Some of the existing automatic metrics only use the surface word information without any linguistic features, which results in low correlation with human judgment and receives much criticism from the linguists. Whereas some metrics utilize too many language-specific linguistic features, which makes it difficult for other language pairs.

7.2 Challenges

MT evaluation researchers have been inventing numerous novel evaluation approaches daily in recent years, thus building a standard code is a challenging issue. The MTE metrics developed by the researchers are compared with the prior approaches which require their implementation that's aggravating. Therefore, a common platform must be developed where new evaluation metrics results can be compared with existing fully AMTE metrics. This will be helpful in critically examining the new MTE metrics. Most of the evaluation metrics work extremely well for lesser complex languages like English but do not produce a reliable result for more complex languages i.e., Indic. Since there are no task-specific or language-specific evaluation metrics. Therefore, developing a task-specific MTE metric is challenging. Low cost, tunable, consistent, meaningful, correct, and robust for different language pairs are other challenges faced in fully AMTE metrics.

7.3 Future Work

- It is essential to provide a human judgment-based evaluation platform to analyze fully AMTE metrics. The dataset availability includes the tuples of the following form: context, reference response, suggested response, and human scores for the proposed solution that is required for the development of automatic evaluation metrics. The suggested response is produced by a synthetically generated variant of a reference response (e.g., eliminating stopwords or modifying synonyms, etc.) or a natural language generation system. These recommended replies are assessed and awarded a score (say, on a scale of 0–1), resulting in a dataset consisting of a mix of some good and bad replies, along with the human ratings.
- Developing a task-specific MTE metric is another challenge as most of the evaluation metrics work extremely well for lesser complex languages but do not produce a reliable result for more complex languages.
- Some of the fully MTE metrics like TER, NIST, BLEU, METEOR, ROUGE, etc., are critically examined by researchers in various studies. But still, there are most of the fully AMTE metrics have yet to be analyzed for different languages. Therefore, a benchmark for the fully MTE is required.

8 Conclusion

Fully automatic MT evaluation measures are meant to be more objective, faster, and less expensive than human evaluation. The usage of this sort of evaluation has grown in popularity over the last few decades among MT developers since it allows them to conduct quick automatic evaluations of their MT models and use the results to improve them in the right direction. Fully AMTE metrics provide a comparison of the MT output to the reference translation. If the quality of reference sentences is good and multiple references have been provided, then most of the fully AMTE metrics will evaluate MT model translated sentences better. In this survey paper, we have reviewed all the fully AMTE metrics that provide a comparison of the MT output to the reference translation by categorizing them in a coherent taxonomy. The challenges and performance of popular automatic evaluation for unsupervised NMT and SMT models and the impact of language and datasets on the evaluation have also been provided. The presented work has provided a thorough examination and classification of various fully automatic evaluation criteria that are crucial in evaluating MT output. The taxonomy's classification of current metrics aids MT researchers in selecting more appropriate evaluation metrics.

There is still need a for robust automatic evaluation metrics, but developing such metrics is a challenging task due to a wide variety of criteria. To build a standard code or a common platform where new evaluation metrics results can be compared with existing evaluation metrics that will help to critically examine the new evaluation metrics. Low cost, tunable, consistent, meaningful, correct, and robust for different language pairs are other challenges faced in fully AMTE metrics.

Most of the automatic evaluation metrics give an equal weightage to the punctuation and other symbols whereas more weightage should be given to named entities and parts-of-speech tags. Some of the existing automatic metrics only use the surface word information without any linguistic features, and result in low correlation with human judgment, whereas some metrics utilize too many language-specific linguistic features, which makes it difficult for other language pairs. Finally, we present the possible future approaches based on our broad survey on completely AMTE measures. To provide a uniform code base for repeatable research and to expand the use of metrics. Second, new task-specific datasets with fine-grained human judgements will be created. Furthermore, providing more interpretable scores that can give exact development directions and build rigorous standards for critically assessing suggested measures to identify their limitations and biases.

Declarations

Conflict of interests The author does not have any conflict of interest.

References

1. Blatz J, Fitzgerald E, Foster G, Gandrabur S, Goutte C, Kulesza A, Sanchis A, Ueffing N (2004) Confidence estimation for machine translation. In: Proceedings of the 20th international conference on computational linguistics, Geneva, Switzerland, pp 315–321
2. Mariani J (2005) Developing language technologies with the support of language resources and evaluation programs. Lang Resour Eval 39(1):35–44
3. Bentivogli L, Cettolo M, Federico M, Federmann C (2018) Machine translation human evaluation: an investigation of evaluation based on post-editing and its relation with direct assessment. In: Proceedings of the international workshop on spoken language translation, Bruges, Belgium, pp 62–69
4. Gonzàlez M, Giménez J (2014) Asiya. An open toolkit for automatic machine translation (meta-)evaluation. Technical Manual, version 3.0. TALP Research Center, LSI Department, Universitat Politècnica de Catalunya. http://asiya.lsi.upc.edu/Asiya_technical_manual_v3.0.pdf
5. Graham Y, Baldwin T, Moffat A, Zobel J (2015) Can machine translation systems be evaluated by the crowd alone. Nat Lang Eng 23(1):3–30
6. Zhou M, Wang B, Liu S, Li M, Zhang D, Zhao T (2008) Diagnostic evaluation of machine translation systems using automatically constructed linguistic check-points. In: Proceedings of the 22nd international conference on computational linguistics (Coling 2008), Manchester, United Kingdom, pp 1121–1128
7. Han L (2016) Machine translation evaluation resources and methods: a survey. arXiv:1605.04515v8. Cornell University Library
8. Chatzikoumi E (2020) How to evaluate machine translation: a review of automated and human metrics. Nat Lang Eng 26(2):137–161
9. Sai AB, Mohankumar AK, Khapra MM (2020) A survey of evaluation metrics used for NLG systems. arXiv preprint arXiv:2008.12009

10. Mikel A, Gorka L, Eneko A, Kyunghyun C (2018) Unsupervised neural machine translation. In: Proceedings of the 6th international conference on learning representations (ICLR), Vancouver, Canada
11. Mikel A, Gorka L, Eneko A (2018) Unsupervised statistical machine translation. In: Proceedings of the 2018 conference on empirical methods in natural language processing, Brussels
12. Lample G, Conneau A, Denoyer L, Ranzato MA (2017) Unsupervised machine translation using monolingual corpora only. In: Proceedings of the 6th international conference on learning representations (ICLR) Canada, arXiv preprint arXiv:1711.00043
13. Burlot F, Yvon F (2019) Using monolingual data in neural machine translation: a systematic study. In: Proceedings of the third conference on machine translation, 2019, Brussels, Belgium. arXiv preprint arXiv:1903.11437
14. Dalvi F, Durrani N, Sajjad H, Vogel S (2018) Incremental decoding and training methods for simultaneous translation in neural machine translation. arXiv preprint arXiv:1806.03661
15. Ramesh A, Parthasarathy VB, Haque R, Way A (2021) Comparing statistical and neural machine translation performance on hindi-to-tamil and english-to-tamil. Digital 1(2):86–102
16. Wang X, Tu Z, Zhang M (2018) Incorporating statistical machine translation word knowledge into neural machine translation. IEEE/ACM Trans Audio Speech Lang Process 26(12):2255–2266
17. Xia Y (2020) Research on statistical machine translation model based on deep neural network. Computing 102(3):643–661
18. Yang Z, Chen W, Wang F, Xu B (2018) Unsupervised neural machine translation with weight sharing. In: 56th Annual meeting of the association for computational linguistics, Melbourne, Australia arXiv preprint arXiv:1804.09057
19. Koehn P, Knowles R (2017) Six challenges for neural machine translation. arXiv preprint arXiv:1706.03872
20. Kishore P, Roukos S, Ward T, Zhu WJ (2002) BLEU: a method for automatic evaluation of machine translation. In: Proceedings of the 40th annual meeting of the association for computational linguistics, Philadelphia, Pennsylvania, USA, pp 311–318
21. Ananthakrishnan R, Bhattacharyya P, Sasikumar M, Shah RM (2007) Some issues in automatic evaluation of english-hindi mt: more blues for bleu. In: Proceeding of 5th international conference on natural language processing, Hyderabad, India
22. Freitag M, Grangier D, Caswell I (2020) BLEU might be guilty, but references are not innocent. arXiv preprint arXiv:2004.06063
23. Liu CW, Lowe R, Serban IV, Noseworthy M, Charlin L, Pineau J (2016) How not to evaluate your dialogue system: an empirical study of unsupervised evaluation metrics for dialogue response generation. arXiv preprint arXiv:1603.08023
24. Stent A, Marge M, Singhai M (2005) Evaluating evaluation methods for generation in the presence of variation. In: International conference on intelligent text processing and computational linguistics, Springer, Berlin, Heidelberg, pp 341–351
25. Zhang Y, Vogel S, Waibel A (2004) Interpreting BLEU/NIST scores: how much improvement do we need to have a better system?. In: Fourth international conference on language resources and evaluation, Portugal
26. Celikyilmaz A, Clark E, Gao J (2020) Evaluation of text generation: a survey. arXiv preprint arXiv:2006.14799
27. Su KY, Wu MW, Chang JS (1992) A new quantitative quality measure for machine translation systems. In: COLING 1992 volume 2: the 14th international conference on computational linguistics
28. Tillmann C, Vogel S, Ney H, Zubiaga A, Sawaf H (1997) Accelerated DP based search for statistical translation. In: Proceeding of EuroSpeech, Rhodes, Greece, pp 2123–2126
29. Post M (2018) A call for clarity in reporting BLEU scores. In: Proceedings of the third conference on machine translation: research papers, WMT 2018, Belgium, Brussels, October 31–November 1, 2018, Ondrej Bojar, Rajen Chatterjee, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno-Yepes, Philipp Koehn, Christof Monz, Matteo Negri, Aurélie Névéol, Mariana L. Neves, Matt Post, Lucia Specia, Marco Turchi, and Karin Verspoor (Eds.). Association for Computational Linguistics, pp 186–191. https://doi.org/10.18653/v1/w18-6319
30. Galley M, Brockett C, Sordoni A, Ji Y, Auli M, Quirk C, Mitchell M, Gao J, Dolan B (2015) deltaBLEU: a discriminative metric for generation tasks with intrinsically diverse targets. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26–31, 2015, Beijing, China, Volume 2: Short Papers. The Association for Computer Linguistics, pp 445–450.https://doi.org/10.3115/v1/p15-2073

31. Libovický J, Pecina P (2014) Tolerant BLEU: a submission to the WMT14 metrics task. In: Proceedings of the ninth workshop on statistical machine translation, pp 409–413
32. Zhu Y, Lu S, Zheng L, Guo J, Zhang W, Wang J, Yu Y (2018) Texygen: A benchmarking platform for text generation models. In: The 41st international ACM SIGIR conference on research & development in information retrieval, pp 1097–1100
33. Chen L, Dai S, Tao C, Zhang H, Gan Z, Shen D, Zhang Y, Wang G, Zhang R, Carin L (2018) Adversarial text generation via feature-mover's distance. In: Advances in neural information processing systems vol 31
34. Lu S, Zhu Y, Zhang W, Wang J, Yu Y (2018) Neural text generation: past, present and beyond. arXiv preprint arXiv:1803.07133
35. Caccia M, Caccia L, Fedus W, Larochelle H, Pineau J, Charlin L (2018) Language gans falling short. In: ICLR 2020—proceedings of the seventh international conference on learning representation Canada. arXiv preprint arXiv:1811.02549
36. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. In: Advances in neural information processing systems, vol 27
37. Semeniuta S, Severyn A, Gelly S (2018) On accurate evaluation of gans for language generation. In: Seventh international conference on learning representations, United States, 2019 URL https://openreview.net/forum?id=rJMcdsA5FX
38. Doddington G (2002) Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In: Proceedings of the second international conference on human language technology research March 2002, pp 138–145
39. Snover M, Dorr B, Schwartz R, Micciulla L, Makhoul J (2006) A study of translation edit rate with targeted human annotation. In: Proceedings of the 7th conference of the association for machine translation in the Americas: technical papers, pp 223–231
40. Snover MG, Madnani N, Dorr B, Schwartz R (2009) Ter-plus: paraphrase, semantic, and alignment enhancements to translation edit rate. Mach Transl 23(2):117–127
41. Kilickaya M, Erdem A, Ikizler-Cinbis N, Erdem E (2016) In Proceedings of the 15th conference of the European chapter of the association for computational linguistics: volume 1, Long Papers. Association for Computational Linguistics, 2017. https://doi.org/10.18653/v1/e17-1019
42. Wong B, Kit C (2009) ATEC: automatic evaluation of machine translation via word choice and word order. Mach Transl 23(2–3):141–155
43. Han AL, Wong DF, Chao LS (2012) LEPOR: a robust evaluation metric for machine translation with augmented factors. In: Proceedings of COLING 2012: Posters, pp 441–450
44. Chen B, Kuhn R, Larkin S (2012). Port: a precision-order-recall MT evaluation metric for tuning. In: Proceedings of the 50th annual meeting of the association for computational linguistics, volume 1: Long Papers, Jeju Island, Korea, pp 930–939
45. Shen L, Turian JP, Melamed ID (2003) Evaluation of machine translation and its evaluation. In: Proceedings of MT Summit IX, New Orleans, U.S.A.
46. Banerjee S, Lavie A (2005) METEOR: an automatic metric for MT evaluation with improved correlation with human judgments. In: Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization
47. Denkowski M, Lavie A (2010). METEOR-NEXT and the METEOR paraphrase tables: improved evaluation support for five target languages. In: Proceedings of the joint fifth workshop on statistical machine translation and MetricsMATR, WMT@ACL 2010, Uppsala, Sweden, July 15–16, 2010, Chris Callison-Burch, Philipp Koehn, Christof Monz, Kay Peterson, and Omar Zaidan (Eds.). Association for Computational Linguistics, pp 339–342. https://www.aclweb.org/anthology/W10-1751/
48. Guo Y, Ruan C, Hu J (2018) Meteor++: incorporating copy knowledge into machine translation evaluation. In: Proceedings of the third conference on machine translation: shared task paper, pp 740–745
49. Gupta A, Venkatapathy S, Sangal R (2010) METEOR-Hindi: automatic MT evaluation metric for hindi as a target. In: Proceedings of ICON-2010: 8th international conference on natural language processing, Macmillan Publishers. India
50. Melamed ID, Green R, Turian J (2003) Precision and recall of machine translation. In: Companion volume of the proceedings of HLT-NAACL 2003-Short Papers, pp 61–63
51. Aliguliyev RM (2008) Using the F-measure as similarity measure for automatic text summarization. Вычислительные технологии 13(3):5–14
52. Isozaki H, Hirao T, Duh K, Sudoh K, Tsukada H (2010) Automatic evaluation of translation quality for distant language pairs. In: Proceedings of the 2010 conference on empirical methods in natural language processing, pp 944–952
53. Popović M (2015) chrF: character n-gram F-score for automatic MT evaluation. In: Proceedings of the tenth workshop on statistical machine translation, WMT@EMNLP 2015, 17–18 September 2015,

Lisbon, Portugal. The Association for Computer Linguistics, pp 392–395. https://doi.org/10.18653/v1/w15-3049

54. Popović M (2017) chrF++: words helping character n-grams. In: Proceedings of the second conference on machine translation, WMT 2017, Copenhagen, Denmark, September 7–8, 2017
55. Wang W, Peter JT, Rosendahl H, Ney H (2016) Character: translation edit rate on character level. In: Proceedings of the first conference on machine translation: Volume 2, Shared Task Papers, pp 505–510
56. Stanojević M, Sima'an K (2014) Beer: better evaluation as ranking. In: Proceedings of the ninth workshop on statistical machine translation, WMT@ACL 2014, June 26–27, 2014, Baltimore, Maryland, USA. The Association for Computer Linguistics, pp 414–419. https://doi.org/10.3115/v1/w14-3354
57. Stanchev P, Wang W, Ney H (2019) EED: extended edit distance measure for machine translation. In: Proceedings of the fourth conference on machine translation (Volume 2: Shared Task Papers, Day 1). Association for Computational Linguistics, Florence, Italy, pp 514–520.https://doi.org/10.18653/v1/W19-5359
58. Chan YS, Ng HT (2008) MAXSIM: a maximum similarity metric for machine translation evaluation. In: Proceedings of ACL-08: HLT, Columbus, Ohi, pp 55–62
59. Taskar B, Lacoste-Julien S, Klein D (2005) A discriminative matching approach to word alignment. In: Proceedings of human language technology conference and conference on empirical methods in natural language processing, pp 73–80
60. Han ALF, Wong DF, Chao LS, He L, Lu Y, Xing J, Zeng X (2013) Language-independent model for machine translation evaluation with reinforced factors. In: Proceedings of the 14th international conference of machine translation summit, pp 215–222
61. Liu D, Gildea D (2005) Syntactic features for evaluation of machine translation. In: Proceedings of the ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization, pp 25–32
62. Collins M, Duffy N (2001) Convolution kernels for natural language. In: Advances in neural information processing systems vol 14
63. Popović M, Ney H (2007) Word error rates: decomposition over POS classes and applications for error analysis. In: Proceedings of the second workshop on statistical machine translation. pp 48–55
64. Duma M, Menzel W (2017) UHH submission to the WMT17 quality estimation shared task. In: Proceedings of the second conference on machine translation, pp 556–561
65. Chauhan S, Daniel P, Mishra A, Kumar A (2021) AdaBLEU: a modified BLEU score for morphologically rich languages. IETE J Res 12:1–12
66. Chauhan S, Saxena S, Daniel P (2021) Monolingual and parallel corpora for kangri low resource language. arXiv preprint arXiv:2103.11596
67. Mikolov T, Chen K, Corrado G, Dean J (2013) Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781
68. Pennington J, Socher R, Manning CD (2014). Glove: global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pp 1532–1543
69. Li P, Chen C, Zheng W, Deng Y, Ye F, Zheng Z (2019) STD: An automatic evaluation metric for machine translation based on word embeddings. IEEE/ACM Trans Audio, Speech Lang Process 27(10):1497–1506
70. Rei R, Stewart C, Farinha AC, Lavie A (2020) COMET: a neural framework for MT evaluation. In: Conference on empirical methods in natural language processing 2020 (online) arXiv preprint arXiv:2009.09025
71. Artetxe M, Schwenk H (2019) Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. Trans Assoc Comput Linguist 7:597–610
72. Lample G, Conneau A (2019) Cross-lingual language model pretraining. arXiv preprint arXiv:1901.07291
73. Lommel A, Uszkoreit H, Burchardt A (2014) Multidimensional quality metrics (MQM): a framework for declaring and describing translation quality metrics. Rev Tradumàtica Tecnol Trad 12:455–463
74. Fonseca E, Yankovskaya L, Martins AF, Fishel M, Federmann C (2019) Findings of the WMT 2019 shared tasks on quality estimation. In: Proceedings of the fourth conference on machine translation (volume 3: Shared Task Papers, Day 2), pp 1–10, Florence, Italy. Association for Computational Linguistics
75. Chen Q, Zhu X, Ling Z-H, Wei S, Jiang H, Inkpen D (2017) Enhanced LSTM for natural language inference. In: Proceedings of the 55th annual meeting of the association for computational linguistics, ACL 2017, Vancouver, Canada, July 30–August 4, volume 1: Long Papers, Regina Barzilay and

Min-Yen Kan (Eds.). Association for Computational Linguistics, pp 1657–1668. https://doi.org/10.18653/v1/P17-1152

76. Devlin J, Chang MW, Lee K, Toutanova K (2018) BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, Volume 1 (Long and Short Papers), pp 4171–4186, Minneapolis, Minnesota
77. Peters ME, Neumann M, Iyyer M, Gardner M, Clark C, Lee K, Zettlemoyer L (2018) Deep contextualized word representations. In: Proceedings of the 2018 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (Long Papers), pp 2227–2237
78. Lo CK (2019) YiSi-a unified semantic MT quality evaluation and estimation metric for languages with different levels of available resources. In: Proceedings of the fourth conference on machine translation (volume 2: Shared Task Papers, Day 1), pp 507–513, Florence, Italy, August 2019. Association for Computational Linguistics. doi:https://doi.org/10.18653/v1/W19-5358. URL https://www.aclweb.org/anthology/W19-5358
79. Lo CK, Wu D (2011) MEANT: an inexpensive, high-accuracy, semiautomatic metric for evaluating translation utility via semantic frames. In: proceedings of the 49th annual meeting of the association for computational linguistics, human language technologies, vol 1, pp 220–229
80. Lo CK, Beloucif M, Saers M, Wu D (2014). XMEANT: better semantic MT evaluation without reference translations. In: Proceedings of the 52nd annual meeting of the association for computational linguistics, Short Papers, 2014, vol 2, pp 765–771.
81. Lo CK, Dowling P, Wu D (2015) Improving evaluation and optimization of MT systems against meant. In: Proceedings of the 10th workshop on statistical machine translation, pp 434–441, Lisbon, Portugal
82. Lo CK (2017) MEANT 2.0: accurate semantic MT evaluation for any output language. In: Second conference on World machine translation, Denmark
83. Banchs RE, D'Haro LF, Li H (2015) Adequacy–fluency metrics: evaluating mt in the continuous space model framework. IEEE/ACM Trans Audio Speech Lang Process 23(3):472–482
84. Wieting J, Berg-Kirkpatrick T, Gimpel K, Neubig G (2019) Beyond BLEU: training neural machine translation with semantic similarity. In: Proceedings of the 57th conference of the association for computational linguistics, ACL 2019, Florence, Italy, July 28–August 2, 2019, Volume 1: Long Papers, Anna Korhonen, David R. Traum, and Lluís Màrquez (Eds.). Association for Computational Linguistics, pp 4344–4355. https://doi.org/10.18653/v1/p19-1427
85. Gekhman Z, Aharoni R, Beryozkin G, Freitag M, Macherey W (2020) KoBE: knowledge-based machine translation evaluation. arXiv preprint arXiv:2009.11027
86. Hiroki S, Tomoyuki K, Mamoru K (2018) RUSE: regressor using sentence embeddings for automatic machine translation evaluation. In: Proceedings of the third conference on machine translation: shared task papers, WMT 2018, Belgium, Brussels, October 31–November 1, 2018, Ondrej Bojar, Rajen Chatterjee, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno-Yepes, Philipp Koehn, Christof Monz, Matteo Negri, Aurélie Névéol, Mariana L. Neves, Matt Post, Lucia Specia, Marco Turchi, and Karin Verspoor (Eds.). Association for Computational Linguistics, pp 751–758. https://doi.org/10.18653/v1/w18-6456
87. Conneau A, Kiela D, Schwenk H, Barrault L, Bordes A (2017) Supervised learning of universal sentence representations from natural language inference data. In: Proceedings of the 2017 conference on empirical methods in natural language processing. Association for Computational Linguistics, Copenhagen, Denmark, pp 670–680. https://doi.org/10.18653/v1/D17-1070
88. Logeswaran L, Lee H (2018) An efficient framework for learning sentence representations. In: 6th International conference on learning representations, ICLR 2018, Vancouver, BC, Canada, April 30—May 3, 2018
89. Cer D, Yang Y, Kong SY, Hua N, Limtiaco N, John RS, Kurzweil R (2018) Universal sentence encoder. arXiv preprint arXiv:1803.11175
90. Shimanaka H, Kajiwara T, Komachi M (2019) Machine translation evaluation with bert regressor. arXiv preprint arXiv:1907.12679
91. Sellam T, Das D, Parikh AP (2020) BLEURT: learning robust metrics for text generation. In: 58th annual meeting of the association for computational linguistics. arXiv preprint arXiv:2004.04696
92. Sellam T, Das D, Parikh AP (2020) BLEURT: learning robust metrics for text generation. arXiv preprint arXiv:2004.04696
93. Rus V, Lintean M (2012) An optimal assessment of natural language student input using word-to-word similarity metrics. In: International conference on intelligent tutoring systems. Springer, Berlin, Heidelberg, pp 675–676

94. Kusner MJ, Sun Y, Kolkin NI, Weinberger KQ (2015) From word embeddings to document distances. In: Proceedings of the 32nd international conference on machine learning, Lille, France, 2015
95. Ren Z, Yuan J, Zhang Z (2011) Robust hand gesture recognition based on finger-earth mover's distance with a commodity depth camera. In: Proceedings of the 19th ACM international conference on Multimedia, pp 1093–1096
96. Clark E, Celikyilmaz A, Smith NA (2019) Sentence mover's similarity: automatic evaluation for multi-sentence texts. In: Proceedings of the 57th annual meeting of the association for computational linguistics, Italy, pp 2748–2760
97. Zhao W, Peyrard M, Liu F, Gao Y, Meyer CM, Eger S (2019) MoverScore: text generation evaluating with contextualized embeddings and earth mover distance. arXiv preprint arXiv:1909.02622
98. Comelles E, Atserias J (2019) VERTa: a linguistic approach to automatic machine translation evaluation. Lang Resour Eval 53(1):57–86
99. Bahdanau D, Cho K, Bengio Y (2014) Neural machine translation by jointly learning to align and translate. In: Proceedings of ICRL 2014, San Diego, USA
100. Cho K, Van Merriënboer B, Bahdanau B, Bengio Y (2014) On the properties of neural machine translation: encoder-decoder approaches. In: Proceedings of SSST-8, eighth workshop on syntax, semantics and structure in statistical translation, Doha, Qatar, pp 103–111
101. Kalchbrenner N, Blunsom P (2013) Recurrent continuous translation models. In: Proceedings of the 2013 conference on empirical methods in natural language processing, Seattle, Washington, USA, pp 1700–1709
102. Sutskever I, Vinyals O, Le Q (2014) Sequence to sequence learning with neural networks. In: Proceedings of advances in neural information processing systems, Montreal, Canada, pp 3104–3112
103. Toral A, Castilho S, Hu K, Way A (2018) Attaining the unattainable? Reassessing claims of human parity in neural machine translation. In: Proceedings of the third conference on machine translation (WMT), Volume 1: Research Papers, Association for Computational Linguistics, Brussels, Belgium, pp 113–123
104. Hassan H, Aue A, Chen C, Chowdhary V, Clark J, Federmann C, Huang X, Junczys-Dowmunt M, Lewis W, Li M, Liu S, Liu T, Luo R, Menezes A, Qin T, Seide F, Tan X, Tian F, Wu L, Wu S, Xia Y, Zhang D, Zhang Z, Zhou M (2018) Achieving human parity on automatic Chinese to English news translation. arXiv:1803.05567
105. Isabelle P, Cherry C, Foster G (2017) A challenge set approach to evaluating machine translation. In: Proceedings of the 2017 conference on empirical methods in natural language processing, Copenhagen, Denmark, pp 2486–2496
106. Sennrich R (2017) How grammatical is character-level neural machine translation? Assessing MT quality with contrastive translation pairs. In: 15th conference of the European chapter of the association for computational linguistics, Spain arXiv:1612.04629v3
107. Klubička F, Toral A, Sánchez-Cartagena VM (2018) Quantitative fine-grained human evaluation of machine translation systems: a case study on English to Croatian. Mach Transl 32(3):195–215
108. Cheng Y, Jiang L, Macherey W (2019) Robust neural machine translation with doubly adversarial inputs. In: Proceedings of the annual meeting of the association for computational linguistics. Florence, pp 4324–4333
109. Cheng Y, Tu Z, Meng F, Zhai J, Liu Y (2018) Towards robust neural machine translation. In: Proceedings of the annual meeting of the association for computational linguistics. Melbourne pp 1756–1766
110. Ding Y, Liu Y, Luan H et al (2017) Visualizing and understanding neural machine translation. In: Proceedings of the annual meeting of the association for computational linguistics. Vancouver, pp 1150–1159
111. Thompson B, Post M (2020) Automatic machine translation evaluation in many languages via zero-shot paraphrasing. arXiv preprint arXiv:2004.14564
112. Kocmi T, Federmann C, Grundkiewicz R, Junczys-Dowmunt M, Matsushita H, Menezes A (2021) To ship or not to ship: an extensive evaluation of automatic metrics for machine translation. arXiv preprint arXiv:2107.10821
113. Koehn P, Hoang H, Birch A, Callison-Burch C, Federico M, Bertoldi N, Cowan B, Shen W, Moran C, Zens R, Dyer C, Bojar O, Constantin A,Herbst E (2007). Moses: open source toolkit for statistical machine translation. In: Proceedings of the 45th annual meeting of the association for computational linguistics companion volume proceedings of the demo and poster sessions, pp 177–180
114. Heafield K (2011) KenLM: faster and smaller language model queries. In: Proceedings of the sixth workshop on statistical machine translation, Edinburgh, Scotland, pp 187–197
115. Kunchukuttan A, Kakwani D, Golla S, Bhattacharyya A, Khapra MM, Kumar P (2020) Ai4bharat-indicnlp corpus: monolingual corpora and word embeddings for indic languages. arXiv preprint arXiv:2005.00085

116. Parton K, Tetreault J, Madnani N, Chodorow M (2011) E-rating machine translation. In: Proceedings of the 6th workshop on statistical machine translation, Edinburgh, Scotland, UK, pp 108–115
117. Song X, Cohn T (2011) Regression and ranking based optimisation for sentence level MT evaluation. In: Proceedings of the sixth workshop on statistical machine translation, pp 123–129

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.