OpenMEVA: A Benchmark for Evaluating Open-ended Story Generation Metrics

Jian Guan[1], **Zhexin Zhang**[1], **Zhuoer Feng**[1], **Zitao Liu**[2], **Wenbiao Ding**[2], **Xiaoxi Mao**[3], **Changjie Fan**[3] **and Minlie Huang**[1*]

[1]The CoAI group, DCST; [1]Institute for Artificial Intelligence; [1]State Key Lab of Intelligent Technology and Systems;
[1]Beijing National Research Center for Information Science and Technology; [1]Tsinghua University, Beijing 100084, China.
[2]TAL Education Group. [3]Netease Fuxi AI Lab.

{j-guan19,zx-zhang18,fze17}@mails.tsinghua.edu.cn, zitao.jerry.liu@gmail.com,
dingwenbiao@100tal.com, {maoxiaoxi,fanchangjie}@corp.netease.com,
aihuang@tsinghua.edu.cn

Abstract

Automatic metrics are essential for developing natural language generation (NLG) models, particularly for open-ended language generation tasks such as story generation. However, existing automatic metrics are observed to correlate poorly with human evaluation. The lack of standardized benchmark datasets makes it difficult to fully evaluate the capabilities of a metric and fairly compare different metrics. Therefore, we propose OpenMEVA, a benchmark for evaluating open-ended story generation metrics. OpenMEVA provides a comprehensive test suite to assess the capabilities of metrics, including (a) the correlation with human judgments, (b) the generalization to different model outputs and datasets, (c) the ability to judge story coherence, and (d) the robustness to perturbations. To this end, OpenMEVA includes both manually annotated stories and auto-constructed test examples. We evaluate existing metrics on OpenMEVA and observe that they have poor correlation with human judgments, fail to recognize discourse-level incoherence, and lack inferential knowledge (e.g., causal order between events), the generalization ability and robustness. Our study presents insights for developing NLG models and metrics in further research.

1 Introduction

Significant advances have been witnessed in many NLG tasks with pretraining models (Devlin et al., 2019; Brown et al., 2020). However, existing generation models are still far behind the human-level performance to generate reasonable texts, particularly for open-ended generation tasks such as story generation (Fan et al., 2018; Guan et al., 2020). One critical obstacle is the lack of powerful metrics for measuring the quality of generation.

The standard paradigm for evaluating NLG metrics is to calculate the correlation with human judgments on manually annotated datasets (Tao et al., 2018; Sellam et al., 2020). Recent studies have discovered that the existing automatic metrics may correlate poorly with human judgments (Liu et al., 2016; Guan and Huang, 2020). Unfortunately, the lack of benchmark datasets makes it challenging to completely assess the capabilities of a metric and fairly compare different metrics. Firstly, annotated datasets usually contain innate data bias and annotation bias. Secondly, summarizing the performance with a single aggregate statistic (e.g., a correlation score) makes it difficult to probe which aspects a metric can successfully capture and which can not. Therefore, many alternative approaches have been proposed to evaluate NLG metrics, such as measuring the robustness to adversarial examples (Zhang* et al., 2020), and the generalization to quality-biased data (Sellam et al., 2020). However, these approaches only focus on an individual capability or a single task, thereby failing to fully reveal the strengths and weaknesses of a NLG metric.

Therefore, we propose OpenMEVA, a benchmark for ***Open**-ended story generation **M**etrics **Eva**luation*. We first collect a *MAN*ually annotated *S*tory dataset (MANS). The stories are generated by various generation models trained on two widely used story corpora, ROCStories (Mostafazadeh et al., 2016) and WritingPrompts (Fan et al., 2018). Therefore, MANS supports to evaluate metrics in terms of not only the correlation with human judgments, but also the generalization w.r.t *model drift* (generations from different models) and *dataset drift* (examples from different datasets).

In addition, OpenMEVA also includes an *AUTO*-constructed *S*tory dataset (AUTOS) to test the robustness and the ability to judge story coherence, namely, the semantic relations and discourse structures in the context. We construct AUTOS by per-

[*] Corresponding author

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, pages 6394–6407
August 1–6, 2021. ©2021 Association for Computational Linguistics

turbing human-written stories, and test the metrics in each single aspect (e.g., the ability to recognize inconsistency) by validating the input-output behavior (Ribeiro et al., 2020). Through such behavioral tests, AUTOS can support to reveal potential issues of metrics in multiple aspects, which would be not traceable in machine-generated examples in MANS.

We conduct extensive experiments to assess the capabilities of existing automatic metrics on OpenMEVA. We find that state-of-the-art metrics still correlate poorly (less than 0.5) with human judgments on MANS. And it is difficult for the learnable metrics to generalize to *model or dataset drift*. Through tests on AUTOS, we observe that most metrics can perform well in recognizing incoherence at *token level* (e.g., unrelated entities) and *sentence level* (e.g., semantic repetition), but fail to recognize *discourse-level* incoherence (e.g., inconsistency) and lack understanding of inferential knowledge (e.g., temporal order between events). Besides, we also show that existing metrics are not robust to a small number of typos and synonym substitution. These findings may inspire new directions for developing NLG models and designing metrics in future research.

We also provide an open-source toolkit which implements various metrics, and therefore supports the comparison and analysis of metrics. In addition, the toolkit provides data perturbation techniques for generating customized test cases beyond AUTOS, which can facilitate fast development of new automatic metrics[1].

2 Related Work

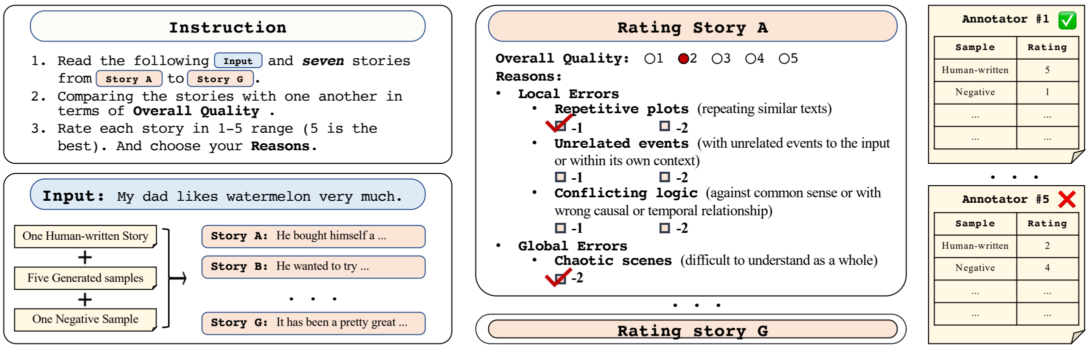
Various automatic metrics have been proposed for evaluating language generation. They can be roughly divided into referenced, unreferenced, and hybrid metrics, according to whether relying on human-written references when calculating the metric score. Referenced metrics usually measure the similarity between a sample and some references based on word-overlap (e.g., BLEU (Papineni et al., 2002), ROUGE (Lin, 2004)) or word embedding (e.g., BERTScore (Zhang* et al., 2020), MoverScore (Zhao et al., 2019)). However, referenced metrics were reported to correlate poorly with human judgments in open-ended generation tasks (Liu et al., 2016) due to the one-to-many issue (Zhao et al., 2017). To address the issue, unreferenced metrics were proposed to measure the quality of a sample without any reference, such as perplexity, discriminator-based metric (Kannan and Vinyals, 2017), UNION (Guan and Huang, 2020) and GRADE (Huang et al., 2020). Besides, hybrid metrics combine referenced and unreferenced metrics (e.g., RUBER and its variant (Tao et al., 2018; Ghazarian et al., 2019)) or learn from the human-annotated score (e.g., ADEM (Lowe et al., 2017), BLEURT (Sellam et al., 2020)).

Recently, there have been many criticisms for existing metrics. Garbacea et al. (2019) showed the poor generalization of discriminator-based metrics. Sai et al. (2019) demonstrated ADEM is not robust to simple attacks such as simple word substitution or random word shuffle. However, these criticisms only focus on individual metrics or capabilities. Notably, Ribeiro et al. (2020) proposed a framework CheckList to evaluate different capabilities of general language understanding models by validating the input-output behavior. The test cases are created from scratch or by perturbing an existing dataset. Similar to Checklist, OpenMEVA also employs automatically constructing examples for behavioral tests. However, CheckList only focuses on single sentences, thereby lacking the ability to test models in understanding long texts with many discourse-level features (e.g., temporal relationship). Moreover, the testing methods of CheckList are not directly applicable for NLG metrics. Specifically, CheckList measures the performance of a model by calculating the failure rate between discrete model prediction and automatic labels. Such failure rates are ineffective for measuring metrics since most metric scores are continuous. To address the above issues, we propose perturbation techniques and testing methods more applicable for story generation metrics.

3 Data Collection

We collect MANS and AUTOS based on ROCStories (**ROC** for short) (Mostafazadeh et al., 2016) and WritingPrompts (**WP** for short) (Fan et al., 2018), which are commonly used for story generation (Guan et al., 2020; Fan et al., 2019) and evaluation (Guan and Huang, 2020). ROC contains 98,162 five-sentence commonsense stories with about 50 words, while WP consists of 303,358 pairs of prompts and stories, which are usually unconstrained on writing topics. We retain about 250 words (with correct sentence boundary) for stories

[1] All the tools, data, and evaluation scripts are available at https://github.com/thu-coai/OpenMEVA



Instruction

1. Read the following Input and *seven* stories from Story A to Story G.
2. Comparing the stories with one another in terms of **Overall Quality** .
3. Rate each story in 1-5 range (5 is the best). And choose your **Reasons**.

Input: My dad likes watermelon very much.

One Human-written Story + Five Generated samples + One Negative Sample

Story A: He bought himself a ...

Story B: He wanted to try ...

. . .

Story G: It has been a pretty great ...

Rating Story A

Overall Quality: ○1 ●2 ○3 ○4 ○5

Reasons:

- **Local Errors**
  - **Repetitive plots** (repeating similar texts)
    ☑ -1 □ -2
  - **Unrelated events** (with unrelated events to the input or within its own context)
    □ -1 □ -2
  - **Conflicting logic** (against common sense or with wrong causal or temporal relationship)
    □ -1 □ -2
- **Global Errors**
  - **Chaotic scenes** (difficult to understand as a whole)
    ☑ -2

. . .

Rating story G

Annotator #1 ✓

Sample	Rating
Human-written	5
Negative	1
...	...
...	...

. . .

Annotator #5 ✗

Sample	Rating
Human-written	2
Negative	4
...	...
...	...

Figure 1: Overview for the manual annotation interface. Story A gets two points in overall quality since it gets three points deducted for its repetitive plot and chaotic scene. The ratings of Annotator #5 for the current story group are rejected because of the low score for the human-written story and the high score for the negative sample.

in WP. Although we only consider the stories in the two corpora, OpenMEVA is designed to measure the capability of NLG metrics to evaluate general linguistic features such as coherence, which may pertain to other stories. Besides, our idea that building datasets by manual annotation or automatic construction can be easily extended to evaluate specific aspects for other types of stories.

3.1 MANS: Manually Annotated Stories

We collect MANS to assess the correlation of metrics with human judgments and the generalization ability when evaluating machine-generated stories. We randomly split ROC and WP by 90%/5%/5% for training/validation/test of the generation models. We regard the first sentence for ROC and the prompt for WP as input. After training, we generate stories based on the test sets. Then, we resort to Amazon Mechanical Turk (AMT) for human judgments of the generated stories. We consider various generation models including a **Seq2Seq** model (Sutskever et al., 2014), **Fusion** (Fan et al., 2018), **Plan&Write** (Yao et al., 2019), the fine-tuned **GPT-2** (Radford et al., 2019) and **K**nowled**G**e-enhanced **GPT-2** (Guan et al., 2020). These models cover diverse network architectures and different levels of the generation ability, which support to evaluate the generalization to examples with different model biases or quality levels.

Manual Annotation We present the manual annotation interface in Figure 1. In each human intelligence task (HIT) of AMT, we show workers the input of a story paired with *seven* stories including (a) five stories generated by the above five models, (b) the human-written story, and (c) a negative example constructed by perturbing a story (e.g., repetition, shuffling) sampled from the test sets. Then we ask workers to compare the *overall quality* of the seven stories[2], and rate each story with a 5-point Likert scale. We reject an HIT if the worker rates the human-written story lower than four points or rates the negative example higher than two points. Through the quality control mechanism, we filtered about 38.7% assignments for ROC and 75.4% for WP. Finally, we ensure that there are five valid ratings for each generated story, and we regard the average rating as the final human judgment.

Considering that overall quality is often too abstract to measure, we follow previous recommendations (Belz and Hastie, 2014; van der Lee et al., 2020) to decide the overall quality by summarizing multiple separate criteria. We ask the workers to decide the rating of a story based on a point deduction policy. Specifically, a story should get punishment in points if it contains errors such as *repetitive plots*, *unrelated events* and *conflicting logic*, or globally *chaotic scenes*, which are commonly observed in existing NLG models (Guan and Huang, 2020) (several examples shown in the appendix). Intuitively, the policy can alleviate the tendency to give high scores and ensure that the judgment standard of workers is as consistent as possible during annotation. To avoid introducing extra bias in the policy, we do not impose the restriction on workers to exactly match the rating in overall quality with the deducted points.

Data Statistics We randomly sampled 200 stories from test sets of ROC and WP for story

[2] We do not ask annotation in other aspects (e.g., *interesting*) since previous work (Novikova et al., 2017) has noted that the annotation scores on different aspects are highly correlated in spite of careful design. And computing correlation scores in the entangled aspects would be unconvincing.

Aspects	Selecting Coherent Examples	Creating Incoherent Examples
Lexical Repetition	All the human-written stories.	**(1)** Repeating a 4-gram (with "and" inserted before it). **(2)** Repeating a sentence. `Case:` ... he stepped on the stage and stepped on the stage ...
Semantic Repetition	All the human-written stories.	**(1)** Repeating a sentence with its paraphrase by back translation[3]. To ensure the semantic similarity and avoid much word overlap, we only use those paraphrases whose MoverScore is larger than 0.4 and BLEU-1 is less than 0.6 with the original sentences. We present some examples for paraphrase generation in the appendix. `Case:` he hired an attorney. he employed a lawyer ... *(MoverScore=0.57, BLEU-1=0.40)*
Character Behavior	Stories with passive voice or with personal pronouns (e.g., "him", "their") for multiple characters. `Case:` ... *it* asked *John* if *John* could ...	**(1)** Reordering the subject and object of a sentence. **(2)** Substituting a personal pronoun with another one which refers to other characters. And we do no change the grammatical case of the substituted pronoun (e.g., "my" can be substituted with "his" but not with "him"). `Case:` *John* ↔ asked *it* if *John* could ...
Common Sense	Stories with both the head and tail entities of a triple in ConceptNet[4](Speer and Havasi, 2012).	**(1)** Substituting 10% entities with its neighboring entity in ConceptNet. `Case:` today is *Halloween* ↦ *Christmas* . Jack is excited to go *trick or treating* ... ("Halloween" and "Christmas" has the relation "Antonyms")
Consistency	Stories with negated words (e.g., "not", "hardly", "inactive"). `Case:` ... Tom decided *not* to give up ...	**(1)** Substituting words with the antonyms (e.g., "happy" vs. "upset"), which are retrieved from WordNet (Miller, 1998). The antonyms are converted to the same form (e.g., verb tense) with the original words. **(2)** Inserting or Deleting negated words for 20% sentences. `Case:` she *agreed* ↦ *disagreed* to get vaccinated ...
Relatedness	Stories with weak token-level semantic relatedness within the context[5]. `Case:` Craig was diagnosed with cancer. he decided to fight it ...	**(1)** Substituting 25% nouns or verbs randomly (with correct word forms). **(2)** Substituting a sentence randomly with another sampled from the dataset. `Case:` Craig was diagnosed with cancer. he decided to fight it. ↦ Kelly wanted to put up the Christmas tree. He tried several different approaches and medications. eventually it went into remission ...
Causal Relationship	Stories with causality-related words (e.g., "because"). `Case:` ... the sky is clear. *so* he can see it .	**(1)** Reordering the cause and effect, which should be two individual sentences or two clauses connected by a causality-related conjunction; **(2)** Substituting the causality-related words with the antonyms (e.g., "reason" vs. "result"). `Case:` ... he can see it. ↔ *so* the sky is clear.
Temporal Relationship	Stories with time-related words (e.g., "before","then"). `Case:` ... Tina *then* learnt her lesson.	**(1)** Reordering two sequential events, which should be two individual sentences or two clauses connected by a time-related conjunction. **(2)** Substituting the time-related words with the antonyms (e.g., "after" vs. "before"). `Case:` ... after ↦ before eating one bite I was not hungry.

Table 1: Examples for the discrimination test to evaluate the ability to judge story coherence in different **aspects**. *Italic* words indicate the crucial keywords for the corresponding aspects. The **coherent examples** are selected from the human-written stories. The **incoherent examples** are created by perturbation including insertion, deletion and reordering, where **(1)** and **(2)** mean different perturbation techniques.

Aspects	Perturbations
Synonyms	Substituting a word with its synonym retrieved from WordNet. `Case:` ... I purchased ↦ bought my uniforms.
Paraphrases	Substituting a sentence with its paraphrase. `Case:` he hired an attorney ↦ he employed a lawyer
Punctuation	Deleting inessential punctuation marks (e.g., commas). `Case:` ... eventually, he became hungry ...
Contraction	Contracting or Expanding contraction. `Case:` ... I'll ↦ will have to keep waiting ...
Typos	Swapping two adjacent characters; Repeating or Deleting a character. We modify less than 2% words of an example to avoid much noise. `Case:` ... an orange ↦ ornage broke her nose.

Table 2: Examples for the invariance test to evaluate the robustness to **perturbations** in different **aspects**.

generation, respectively. Therefore, MANS contains $2 \times 200 \times 5 = 2,000$ annotated machine-generated stories, paired with corresponding inputs and human-written references. The Krippendorff's α (Krippendorff, 2018) of the human judgments is $0.77/0.71$ for ROC/WP, indicating a moderate inter-annotator agreement ($\alpha \in [0.67, 0.8]$). We show more statistical details in the appendix.

3.2 AUTOS: Auto-Constructed Stories

While improving correlation with human judgments is the ultimate goal for developing automatic metrics, merely relying on limited annotated data may make the true evaluation performance overestimated (Ribeiro et al., 2020). Besides, a machine-generated story may contain multiple entangled errors (e.g., repetition, unrelatedness), which do not support individual tests for metrics. Therefore, we propose to evaluate the capabilities of metrics with auto-constructed test examples (i.e., AUTOS), each of which is created to focus on a single aspect. We construct AUTOS based on the human-written stories in the test sets of ROC and WP.

Aspects We argue that an ideal metric for evaluating open-ended language generation should have at least the following capabilities: (a) the ability to *judge* story coherence, which requires recognizing **lexical** and **semantic repetition**, unreasonable **character behavior** (e.g., chaotic coreferences), violation of **common sense** (e.g., *"trick or treat"* on *"Christmas"*), poor **consistency** and **relatedness**, incorrect **causal** and **temporal relationship**; and (b) the *robustness* to perturbations, such as substituting with **synonyms** or **paraphrases**, deleting unimportant **punctuation** marks, contracting

[2] We generate paraphrases based on the back translation augmentation system of UDA (Xie et al., 2020).

[3] ConceptNet is a knowledge base including millions of commonsense triples like (h, r, t), meaning that the head entity h has a relation r with the tail entity t. Note that we only regard nouns and verbs as entities.

[4] We regard the stories with maximum inter-sentence MoverScore less than 0.1 as those which have weak token-level semantic relatedness within the context.

full expressions or expanding **contractions**, and adding **typos**. Tests in these aspects require metrics to fully understand the linguistic features at token level (e.g., synonyms), sentence level (e.g., semantic similarity), and discourse level (e.g., context relatedness in content and proper sentence orders), and possess knowledge about common sense, causality, etc., which are usually not traceable in machine-generated stories. Although these aspects are not exhaustive, it is a starting point for further research. Table 1 and 2 present some examples for the two capabilities, respectively.

Test Types We create examples with different test types to evaluate the above capabilities of metrics. Firstly, we evaluate the ability to judge story coherence by the *discrimination test*, which requires metrics to distinguish human-written coherent examples from incoherent ones. We create each incoherent example by applying perturbation within a single aspect. Besides, we also select different human-written stories as coherent examples for different aspects, as shown in Table 1. For robustness assessment, we expect the metric scores to remain the same with certain perturbations, i.e., the *invariance test*, as shown in Table 2.

However, the perturbation may inevitably introduce grammar errors. To alleviate the issue, we filter out those ungrammatical examples in AUTOS except for those used to evaluate robustness to typos using an automatic grammaticality classifier. We present the statistics of AUTOS together with the evaluation results in Table 6/ 7 for the discrimination/invariance tests, respectively. And we provide more details about the construction of AUTOS and the grammaticality classifier in the appendix.

4 Evaluation

We evaluated existing metrics on OpenMEVA, and analyzed the strengths and weaknesses with extensive experiments.

4.1 Evaluated Metrics

We experimented with existing metrics of different types as follows: **(a) Referenced Metrics:** the word-overlap based metric sentence BLEU score (geometric mean from 1-gram to 4-gram) (Papineni et al., 2002), the contextualized embedding based metrics, BERTScore-F1 (Zhang* et al., 2020). **(b) Unreferenced Metrics:** Perplexity[6] estimated by GPT-2 (Radford et al., 2019) (including pretrained GPT-2 and GPT-2 fine-tuned on the training sets); the self-supervised metric UNION (Guan and Huang, 2020). **(c) Hybrid Metrics:** RUBER-BERT (Ghazarian et al., 2019) that improves RUBER with contextualized embeddings from BERT (Devlin et al., 2019).

In addition, we also reported the performance of the unreferenced version in RUBER-BERT, denoted as R_u-BERT. And we present results with more metrics in the appendix.

4.2 Correlation with Human Judgments

We first calculate the Pearson correlation coefficient between metric scores and human judgments on MANS. Besides, we also evaluate metrics on the other four evaluation sets constructed for individual error types (described in Section 3.1) based on MANS. Each of them contains all the reasonable samples and the unreasonable samples of some error type. A reasonable sample means its overall quality score larger than four points. For an unreasonable sample, we decide it is of some error type if there is only one error type annotated by at least three of five annotators. We assign the reasonable and unreasonable samples with binary labels 1 and 0, respectively, and calculate the correlation between metric scores and the binary labels on the four evaluation sets.

We summarize the correlation results in Table 3. As previous studies (Guan and Huang, 2020) observed, unreferenced metrics are more competitive for evaluating open-ended language generation than referenced ones. PPL (F) performs better than PPL (P) on ROC but not on WP, which may be because stories in ROC are created artificially and hence differ from the general language distribution during pretraining GPT-2. Furthermore, measuring input-output relatedness (R_u-BERT) is not enough for language generation evaluation. UNION outperforms other metrics in overall quality assessment since it learns to distinguish human-written stories from negative samples with more error types. Interestingly, it seems easier for the metrics to recognize surface errors (e.g., repetitive plots) or serious global errors (e.g., chaotic scenes). **However, the best correlation with human judgments is still fairly low, and it is difficult to recognize unrelatedness and conflicting plot**. The results indicate the huge room to improve the metrics.

To further examine to what extent the improve-

[6] We follow Guan and Huang (2020) to take the minus of perplexity to ensure a higher value means better quality.

Metrics	ROC					WP				
		46 Reasonable Samples +					35 Reasonable Samples +			
	Overall	Rept	Unrel	Conf	Chao	Overall	Rept	Unrel	Conf	Chao
	1,000	22	319	39	87	1,000	23	330	83	24
BLEU	-0.0239	0.0520	0.0192	0.1134	0.0156	-0.0537	0.1188	-0.0421	-0.0875	-0.1451
BERTScore-F1	0.1271*	0.1396	0.1240	0.0626	0.2283*	0.0329	0.1198	0.0446	0.0189	0.0634
PPL (P)	0.2547*	-0.1075	0.1105	0.1354	0.5248*	0.3033*	0.0219	**0.1853***	**0.2188**	**0.4428***
PPL (F)	0.2817*	0.2152	0.1380*	**0.2643**	**0.5910***	0.2952*	0.0179	0.1720*	0.1917	0.3182*
R_u-BERT	0.0830*	0.1160	0.0877	0.1103	0.1774	0.1666*	0.0936	0.0793	0.0162	0.0077
UNION	**0.4119***	**0.4517***	**0.2000***	0.2107	0.4695*	**0.3256***	**0.3283**	0.1738*	0.1914	0.3967*
RUBER-BERT	0.1434*	0.0813	0.1453*	0.1173	0.1723	0.2116*	0.0716	0.1132	0.0721	0.1493

Table 3: Pearson correlation with human judgments on MANS. PPL (P) and PPL (F) mean Perplexity estimated by *pretrained* and *fine-tuned* GPT-2, respectively. The best performance is highlighted in **bold**. The results contain the correlation with human judgments on all the annotated samples in MANS (**Overall**), and the correlation with the binary labels on reasonable samples and unreasonable ones of different error types. The error types include **Repe**titive plots, **Unrel**ated events, **Confl**icting logic and **Chao**tic scenes. The numbers in the table header denote the number of corresponding stories. * indicates the correlation score is significant (p-value<0.01).

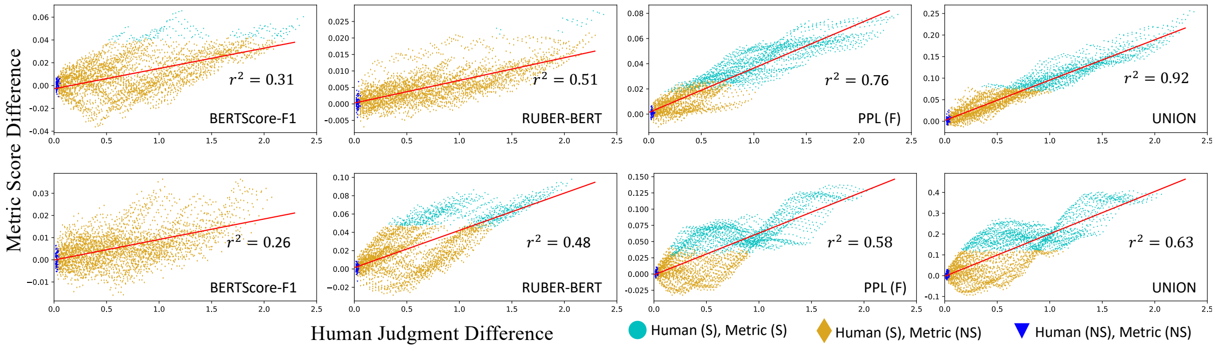


Figure 2: Correlation between human judgment difference (x-axis) and metric score difference (y-axis). Top: ROC, Bottom: WP. We only show the situation in the positive x-axis, since it is centrosymmetric with that in the negative x-axis. **Human (S)/Metric (S)** means the difference of human judgment/metric score is significant (p<0.01, t-test), while **(NS)** means insignificant difference. r^2 is the coefficient of determination for linear regression (red line), and is exactly the square of the Pearson correlation coefficient between the x-axis and y-axis.

ment in an automatic metric corresponds to the improvement in human judgments, we calculate the correlation between human judgment difference and metric score difference (Mathur et al., 2020). Specifically, we sort the 1,000 stories (for ROC and WP, respectively) in MANS by the human judgments, and then select consecutive 200 stories from the beginning and repeat the selection with a stride 10. We finally get $(1,000 - 200)/10 = 80$ story sets[7]. We decide the human judgment or metric score of each set by averaging that of the stories in the set. We calculate the human judgment difference and metric score difference between any two sets of them ($80 \times 80 = 6,400$ pairs totally), and present the correlation between the differences in Figure 2 for several typical metrics. We can see that a significant improvement in the metrics usually corresponds to a significant improvement in human judgments (cyan/dark gray part in Figure 2). However, both an insignificant drop and improvement in a metric could correspond to a significant improvement in human judgments. And worse, the improvement in human judgments may have a wide range, which is particularly evident for BERTScore-F1 and RUBER-BERT (yellow/light gray part in Figure 2). **That is, if an NLG model achieves insignificantly better scores in the two metrics, it is quite possible that the model performs significantly worse in human judgments.** The situation is improved when using PPL (F) and UNION, suggesting that they may be better to measure language generation.

4.3 Generalization Ability

It is extremely important for learnable metrics to deal with *model drift* and *dataset drift* (Garbacea et al., 2019; Sellam et al., 2020). Specifically, a generalizable metric should be able to evaluate dif-

[7] We do not construct the sets by randomly sampling since it would be difficult to cover wide enough quality levels.

ferent NLG models since the generation quality or inductive bias can vary significantly across models. Besides, we also expect a metric to reliably evaluate output from different datasets even without re-training. Therefore, we assess the generalization ability of learnable metrics, including PPL (F), R_u-BERT and UNION, which are fine-tuned on the training sets of ROC and WP, respectively.

To assess the generalization to model drift, we test the metrics on stories generated by five aforementioned models in MANS, respectively (200 stories by each model). Table 4 presents the performance, which varies considerably with models. R_u-BERT only achieves a good correlation on those stories with poor relatedness (e.g., Seq2Seq on WP). PPL (F) and UNION perform comparably but neither do well in evaluating all the NLG models.

	Metrics	S2S	P&W	Fusion	GPT-2	KG-G
ROC	PPL (F)	**0.14**	0.22	**0.12**	0.14	0.25*
	R_u-BERT	-0.02	-0.08	0.04	0.12	0.06
	UNION	0.12	**0.28***	0.10	**0.15***	**0.32***
WP	PPL (F)	0.11	**0.15**	0.05	**0.12**	0.13
	R_u-BERT	**0.18***	0.08	0.14	0.07	0.02
	UNION	0.09	0.02	**0.15**	0.04	**0.15***

Table 4: Pearson correlation with human judgments to assess generalization to output from different models including Seq2Seq (S2S), Plan&Write (P&W), Fusion, GPT-2, KG-GPT-2 (KG-G). The best performance among the metrics is highlighted in **bold**.

To assess the generalization to dataset drift, we first trained the metrics on ROC and then directly used them to evaluate stories from WP, and vice versa. As shown in Table 5, all the metrics drops significantly in correlation when used for the other dataset due to the difference in length and topic. PPL (F) and UNION also have similar performance drops but are more generalizable. **The results suggest existing metrics fall short of generalization.**

Metrics	Train: ROC		Train: WP	
	Test: ROC	Test: WP	Test: ROC	Test: WP
PPL(F)	**0.2817***	0.2423*	0.2470*	**0.2952***
R_u-BERT	**0.0830***	0.0379	0.0891*	**0.1666***
UNION	**0.4119***	0.2287*	0.2128*	**0.3256***

Table 5: Pearson correlation with human judgments to assess generalization to samples from different datasets. The best performance between two test datasets (each row) for each metric is highlighted in **bold**.

4.4 Ability to Judge Story Coherence

We assess the ability of the unreferenced metrics[8] to judge story coherence based on the discrimination test set of AUTOS. We assign each test example with a binary label (1/0 for the coherent/incoherent example). Then we calculate the correlation between metric scores and the binary labels on the test examples of different aspects. The higher correlation means the better ability to judge coherence.

Table 6 presents the correlation results. We summarize the results as follows: **(1) PPL is ineffective to recognize repetition errors.** The observation is accordant with the results on MANS (Table 3). PPL (P) even has a significantly negative correlation with labels in lexical and semantic repetition. **(2) PPL (F) and UNION have better average performance than others.** R_u-BERT performs worst in almost all the aspects. UNION has the highest average performance by a large margin on ROC but underperforms PPL (F) on WP, indicating the shortage of UNION when evaluating longer stories. Besides, the results show that a powerful language model may also be a powerful evaluator (if we can alleviate its preference for repetitive texts). **(3) Existing metrics perform well in recognizing incoherence at token and sentence levels.** For example, they seem to be able to recognize unreasonable behavior for a certain character, and possess some commonsense knowledge about entity relations. However, in this work the proposed perturbation can not fully cover all possible incoherence in these aspects, which would be regarded as the future work. **(4) The metrics still struggle to recognize discourse-level incoherence.** Specifically, it is difficult to recognize inconsistent events when we insert or delete negated words, and understand the semantic relatedness across sentences. Besides, they also lack inferential knowledge about the causal and temporal relationship. The observations are also accordant with the results in Table 3 where unrelated events and conflicting logic can not be well recognized. In conclusion, we reveal various issues of the existing metrics by the isolating behavioral testing, while they achieve moderate correlation with human judgments on MANS.

[8] It is meaningless to evaluate referenced or hybrid metrics on AUTOS since the reference text of a positive example is exactly itself, which is an unfair case for unreferenced metrics.

Metrics		Lexical Repetition	Semantic Repetition	Character Behavior	Common Sense	Consistency	Relatedness	Causal Relationship	Temporal Relationship
ROC	**Cohe**	4,736	4,736	1,022	1,921	455	563	476	2,376
	Incohe	4,049	3,243	266	448	3,666	3,570	410	1,799
PPL (P)		-0.1886*	-0.0719*	0.2547*	**0.4246***	0.1357*	0.0744*	0.1002*	0.1759*
PPL (F)		0.0287*	0.2315*	**0.3595***	0.3976*	0.1630*	0.1458	**0.1568***	**0.2007**
R$_u$-BERT		0.0121	0.0543*	0.0671*	0.0478*	0.0194*	0.0764*	-0.0075	0.0135*
Union		**0.5454***	**0.5631***	0.3191*	0.3965*	**0.1676***	**0.2045***	0.1425*	0.1769*
WP	**Cohe**	9,922	9,922	3,911	2,052	2,914	497	4,552	9,408
	Incohe	9,022	8,381	173	235	6,239	851	3,057	7,092
PPL (P)		-0.0886*	-0.0461*	0.2077*	0.4782*	0.2575*	0.1328*	0.0355*	0.0763*
PPL (F)		-0.0467*	0.0986*	0.2783*	**0.4871***	0.3420*	**0.2297***	**0.1597***	**0.1788***
R$_u$-BERT		0.0098	0.0108	-0.0299	-0.0183	0.0137	0.0054	-0.0143	0.0042
Union		**0.2302***	**0.2150***	**0.3044***	0.3940*	**0.3661***	0.2107*	0.0514*	0.0459*

Table 6: Pearson correlation with automatic labels on the discrimination test set of AUTOS. The higher correlation indicates the better ability to judge story coherence in different aspects. The best performance is highlighted in **bold**. **Cohe** and **Incohe** stand for the number of coherent and incoherent examples, respectively.

Metrics	Synonym		Paraphrase		Punctuation		Contraction		Typo	
	Human	**Dis**	**Human**	**Dis**	**Human**	**Dis**	**Human**	**Dis**	**Human**	**Dis**
ROC	3,777	2,395	3,174	2,194	574	171	1,602	1,208	4,755	4,763
PPL (P)	0.3162*	0.2515*	0.1450*	0.0916*	0.0922*	0.0856	-0.0557	-0.0522*	0.4124*	0.2616*
PPL (F)	0.3309*	0.2521*	0.2742*	0.2022*	0.1475*	0.0996	0.0504	0.0331*	0.4540*	0.2973*
RUBER$_u$-BERT	**0.0307***	**0.0290***	**0.0255**	**0.0263**	**0.0052**	**-0.0140**	**0.0064**	**0.0071**	**-0.0112**	**0.0042**
Union	0.2187*	0.1169*	0.1112*	0.0399*	0.0818*	0.1375*	0.0275	0.0251	0.6021*	0.4606*
WP	6,961	35,90	7,881	2,576	4,535	2,287	8,731	4,522	15,073	15,082
PPL (P)	0.2174*	0.1822*	0.0910*	0.0617*	0.2690*	0.2178*	-0.0222*	-0.0157	0.3983*	0.3885*
PPL (F)	0.2964*	0.1747*	0.2273*	0.1020*	0.3822*	0.2515*	0.0851*	0.0682*	0.4603*	0.4043*
RUBER$_u$-BERT	**-0.0013**	**0.0004**	**0.0000**	**0.0000**	**-0.0256***	**-0.0308***	**-0.0012**	**-0.0043**	**0.0133**	**0.0154***
Union	0.1077*	0.0843*	0.0389*	0.0292*	0.2182*	0.2224*	0.0185*	0.0173*	0.3812*	0.3208*

Table 7: Pearson correlation with automatic labels on the invariance test set of AUTOS. The smaller absolute value of correlation indicates the better robustness. The best performance is highlighted in **bold** and the second best is underlined. The numbers in the **ROC/WP** rows indicate how many human-written stories (**Human**) and incoherent samples from the discrimination test set (**Dis**) are perturbed.

4.5 Robustness Evaluation

A reliable metric should produce similar judgments for an example with simple perturbations or attacks in the input. Therefore, it is essential to evaluate the robustness of metrics. We test the robustness on the invariance test set of AUTOS. We assign each example with a binary label (1/0 for the original/perturbed example). Then, we calculate the correlation between metric scores and the binary labels. The original examples can be sampled either from human-written stories or from the incoherent examples in the discrimination test set.

Table 7 shows the robustness results. It is not surprising that R$_u$-BERT has the "best robustness" since the perturbations hardly influence the input-output relatedness. The result validates the relatedness is merely one side for evaluating NLG, but not means that it is a promising direction for developing robust metrics[9]. PPL is not robust to synonym substitution because the low-frequency words introduced by the perturbations (e.g., from *"happy"* to *"joyful"*) can cause significant change in PPL. Union has better robustness on average thanks to the robust contextualized representation of BERT. Furthermore, both PPL and Union perform better in contraction than in other aspects. However, they are very sensitive to a small number of typos (less than 2% words) because typos may bring some out-of-vocabulary words. Although the issue is common for almost all the (sub)word-based metrics, it is still important to handle typos since they are also common in human writing.

5 Conclusion

We present OpenMEVA, a benchmark to comprehensively assess capabilities of metrics for evaluating open-ended story generation. OpenMEVA includes test examples which are created by either annotating machine-generated stories or perturbing human-written stories in terms of each single

[9] We can imagine that a constant metric has the perfect robustness to any perturbations, but is useless for evaluation.

aspect. We evaluate a number of existing metrics on OpenMEVA and analyze their performance on each capability extensively. Experiments demonstrate that existing metrics still correlate weakly with human judgments, fail to recognize discourse-level incoherence, and lack inferential knowledge, generalization and robustness. Our study reveals the weaknesses of existing metrics and may inspire new research on designing NLG metrics.

The datasets, data augmentation tools, and implemented metrics in this paper can facilitate further research on language generation and evaluation.

Acknowledgments

This work was supported by National Key R&D Program of China, under Grant No. 2020AAA0104500. This work was jointly supported by the NSFC projects (Key project with No. 61936010 and regular project with No. 61876096), and the Guoqiang Institute of Tsinghua University, with Grant No. 2019GQG1 and 2020GQG0005. We would also like to thank the anonymous reviewers for their invaluable suggestions and feedback.

Ethics Statement

We build OpenMEVA based on two existing public story datasets ROCStories (ROC) and WritingPrompts (WP), which are widely used for story generation and evaluation. We resorted to Amazon Mechanical Turk (AMT) for manual annotation of stories in MANS. We did not ask about personal privacy or collect personal information of annotators in the annotation process. We hired five annotators and payed each annotator \$0.05 and \$0.1 for annotating each story in ROC and WP, respectively. We decided the payment according to the average story length of two datasets. We admit that there may be still unpredictable bias in MANS even though we have asked three experts to review all the annotated stories.

Besides, we selected or constructed the test examples in AUTOS based on general linguistic features. We did not adopt any selecting strategies or perturbation techniques which may introduce extra bias into AUTOS.

References

Satanjeev Banerjee and Alon Lavie. 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*, pages 65–72.

Anja Belz and Helen Hastie. 2014. Towards comparative evaluation and shared tasks for nlg in interactive systems. In *Natural Language Generation in Interactive Systems*, pages 302–350. Cambridge University Press.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*, pages 4171–4186.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898.

Angela Fan, Mike Lewis, and Yann Dauphin. 2019. Strategies for structuring story generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2650–2660, Florence, Italy. Association for Computational Linguistics.

Cristina Garbacea, Samuel Carton, Shiyan Yan, and Qiaozhu Mei. 2019. Judge the judges: A large-scale evaluation study of neural language models for online review generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3959–3972.

Sarik Ghazarian, Johnny Wei, Aram Galstyan, and Nanyun Peng. 2019. Better automatic evaluation of open-domain dialogue systems with contextualized embeddings. In *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*, pages 82–89.

Jian Guan, Fei Huang, Zhihao Zhao, Xiaoyan Zhu, and Minlie Huang. 2020. A knowledge-enhanced pretraining model for commonsense story generation. *Transactions of the Association for Computational Linguistics*, 8:93–108.

Jian Guan and Minlie Huang. 2020. UNION: an unreferenced metric for evaluating open-ended story generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 9157–9166. Association for Computational Linguistics.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Lishan Huang, Zheng Ye, Jinghui Qin, Liang Lin, and Xiaodan Liang. 2020. GRADE: automatic graph-enhanced coherence metric for evaluating open-domain dialogue systems. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 9230–9240. Association for Computational Linguistics.

Anjuli Kannan and Oriol Vinyals. 2017. Adversarial evaluation of dialogue models. *arXiv preprint arXiv:1701.08198*.

Klaus Krippendorff. 2018. *Content analysis: An introduction to its methodology*. Sage publications.

Chris van der Lee, Albert Gatt, Emiel van Miltenburg, and Emiel Krahmer. 2020. Human evaluation of automatically generated text: Current trends and best practice guidelines. *Computer Speech & Language*, page 101151.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Chia-Wei Liu, Ryan Lowe, Iulian Vlad Serban, Mike Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2122–2132.

Ryan Lowe, Michael Noseworthy, Iulian Vlad Serban, Nicolas Angelard-Gontier, Yoshua Bengio, and Joelle Pineau. 2017. Towards an automatic turing test: Learning to evaluate dialogue responses. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1116–1126.

Nitika Mathur, Timothy Baldwin, and Trevor Cohn. 2020. Tangled up in BLEU: Reevaluating the evaluation of automatic machine translation evaluation metrics. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4984–4997, Online. Association for Computational Linguistics.

George A Miller. 1998. *WordNet: An electronic lexical database*. MIT press.

John Morris, Eli Lifland, Jin Yong Yoo, Jake Grigsby, Di Jin, and Yanjun Qi. 2020. Textattack: A framework for adversarial attacks, data augmentation, and adversarial training in nlp. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 119–126.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of NAACL-HLT*, pages 839–849.

Jekaterina Novikova, Ondřej Dušek, Amanda Cercas Curry, and Verena Rieser. 2017. Why we need new evaluation metrics for nlg. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2241–2252.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8).

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, Online. Association for Computational Linguistics.

Ananya B Sai, Mithun Das Gupta, Mitesh M Khapra, and Mukundhan Srinivasan. 2019. Re-evaluating adem: A deeper look at scoring dialogue responses. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6220–6227.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Robert Speer and Catherine Havasi. 2012. Representing general relational knowledge in conceptnet 5. In *LREC*, pages 3679–3686.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Chongyang Tao, Lili Mou, Dongyan Zhao, and Rui Yan. 2018. Ruber: An unsupervised method for automatic evaluation of open-domain dialog systems.

In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Alex Warstadt, Amanpreet Singh, and Samuel R Bowman. 2019. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.

Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. 2020. Unsupervised data augmentation for consistency training. *Advances in Neural Information Processing Systems*, 33.

Lili Yao, Nanyun Peng, Ralph Weischedel, Kevin Knight, Dongyan Zhao, and Rui Yan. 2019. Plan-and-write: Towards better automatic storytelling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7378–7385.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Tiancheng Zhao, Ran Zhao, and Maxine Eskenazi. 2017. Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 654–664.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M Meyer, and Steffen Eger. 2019. Moverscore: Text generation evaluating with contextualized embeddings and earth mover distance. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 563–578.

A Details for MANS

A.1 Story Collection

Data Processing We collect machine-generated stories based on ROC and WP. To achieve better generation and generalization performance, we follow Guan et al. (2020) to delexicalize stories in ROC by masking all the names with placeholders, and retain about 250 words (with correct sentence boundary) from the beginning and truncated the rest in WP.

Story Generation After training, we use the generation models to generate stories based on the test sets of ROC and WP. We adopt nucleus sampling (Holtzman et al., 2020) with $p = 0.9$ for story generation to avoid as many repetition errors as possible, since such cases are easier for recognition and simulation (we cover the repetition errors mainly with the test examples in AUTOS).

A.2 Manual Annotation

Table 8 shows examples of four error types and corresponding deduction points. The error types includes repetitive plots, unrelated events, conflicting logic and chaotic scenes.

Input: My dad likes watermelon very much.	**Error Types**
My mom bought him a watermelon. He *picked it up and picked it up*. He *picked it up and picked it up*. Now he have a big smile on his face.	**Repetitive plots** (-1)
He *ate sweets and sweets. He ate it all day. He ate it all. He ate it all.*	**Repetitive plots** (-2)
One day he decided to try a new *recipe*. He bought all the *ingredients*. He followed the *recipe*. It was the best *sauce* he ever tasted.	**Unrelated events** to the beginning (-1)
He decided to buy a *banana*. He picked up a big *oak tree*. He put it in the kitchen. He is happy with the watermelon.	**Unrelated events** to the beginning and within its context (-2)
He had a watermelon this morning. *He wanted another one*. He went to buy one. *He didn't want to eat watermelons*.	**Conflicting logic** (-1)
I buy a watermelon for him. It is pretty great for my dad. *He doesn't like it*. He finally asked me to *be his girlfriend*.	**Conflicting logic** (-2)
I had a watermelon when I was a child. I was feeding him fruits. I picked it up and put it in the house. He asked me to be his son.	**Chaotic scenes** (-2)

Table 8: Examples of four error types and corresponding deduction points (in the parentheses) given the same input. *Italic* words indicate the keywords crucial for the errors.

A.3 Statistics

The Krippendorff's α is 0.77 for ROC and 0.71 for WP, indicating a moderate inter-annotator agreement according to the interpretation in Table 9. We present the distribution of human judgments for different models in Figure 3 and other statistics in Table 10. The results show the diversity of the stories in length and quality.

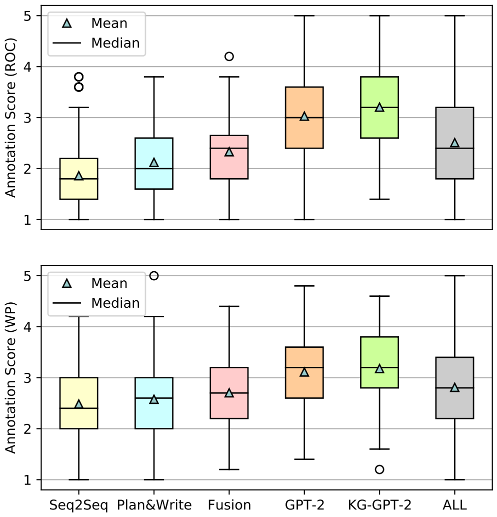


Figure 3: Boxplot of human judgments for each story source (Top: ROC, Bottom: WP).

α	**Interpretation**
< 0.67	not good
$0.67 \sim 0.8$	allowing tentative conclusions to be drawn
> 0.8	good reliability

Table 9: Interpretation of Krippendorff's α.

Statistics	**ROC**	**WP**
Unique Inputs	200	200
Generated Stories (per Input)	5	5
Generated Stories (totally)	1,000	1,000
Average Input Tokens	9.26	22.09
Average Reference Tokens	39.54	238.51
Average Story Tokens	39.38	232.51

Table 10: Statistics of MANS. Text is tokenized with spaCy tokenizer[10].

A.4 Correlation with Human Judgments

We experimented with more popular metrics as follows: `ROUGE-L` (Lin, 2004), `METEOR` (Banerjee and Lavie, 2005), embedding based metrics (including `Greedy Matching`, `Embedding Average` and `Vector extrema` (Liu et al., 2016)) with BERT embedding, `BERTScore` (inlcuding `Precision` and `Recall`), and `MoverScore`.

[10] `https://spacy.io/api/tokenizer`

RUBER, and the supervised metric BLEURT which is fine-tuned on the released annotation results from Guan and Huang (2020). The experiment results is shown in Table 11.

Metrics	ROC	WP
Referenced Metrics		
BLEU	-0.0239	-0.0537
ROUGE-L	0.0188	-0.0107
METEOR	0.0155	-0.0079
Greedy Matching	**0.1892***	-0.0510
Vector Average	0.1840*	-0.0429
Vector Extrema	0.1021*	-0.0241
BERTScore-P	0.1538*	0.0857*
BERTScore-R	0.0838*	-0.0215
BERTScore-F1	0.1271*	0.0329
MoverScore	0.1294*	-0.0586
R$_r$-BERT	0.0808*	**0.1567***
Unreferenced Metrics		
PPL (P)	0.2547*	0.3033*
PPL (F)	0.2817*	0.2952*
R$_u$-BERT	0.0830*	0.1666*
UNION	**0.4119***	**0.3256***
Hybrid Metrics		
RUBER	0.0119	-0.0527
RUBER-BERT	0.1434*	**0.2116***
BLEURT	**0.3163***	0.1738*

Table 11: Pearson correlation with human judgments on MANS. The best performance for each type of metrics is highlighted in **bold**. The correlation scores marked with * indicate the result significantly correlates with human judgments (p-value<0.01).

B Details for AUTOS

B.1 Construction

We list some technical details for constructing AUTOS within different aspects as follows:

- **Semantic Repetition** and **Paraphrases**: We present several examples for paraphrase generation in Table 14. We adopt MoverScore and BLEU-1 to measure the semantic similarity and word overlap between the paraphrases and the original sentences, respectively. We finally only use the paraphrase whose MoverScore is larger than 0.4 and BLEU-1 is less than 0.6 with the original sentence, because they achieve both high semantic similarity and low word overlap.

- **Character Behaviour**: We recognize the personal pronouns in a story following Table 13. We select those stories which contain at least three types of person (i.e., at least three pronouns from different rows) as the coherent examples. And when substituting the pronouns to create incoherent examples, we only perform the substitution in the same column (e.g., *"my"* can be only substituted with *"our"*, *"your"*, etc.) for better grammaticality.

- **Consistency**, **Causal and Temporal Relationship**: We present the negated words, causality-related words and the time-related words in Table 12.

B.2 Grammaticality Classifier

We train a binary classifier on the CoLA corpus (Warstadt et al., 2019) to learn to judge the grammaticality, and then filter out those examples that are classified as ungrammatical (the classifier score less than 0.5). For simplicity, we directly use the public model from TextAttack (Morris et al., 2020) as a classifier to filter out those examples in AUTOS with poor grammaticality. The classifier is fine-tuned on the CoLA corpus based on BERT and achieves an accuracy of 82.90% on the test set of CoLA. Furthermore, if we suppose that all of the human-written stories in ROC and WP are grammatical, the accuracy of the classifier on the stories would be 96.48% and 65.68% for ROC and WP, respectively. The results are intuitive since stories in WP may contain much informal English (e.g., website link). We present several examples in Table 15 to further indicate the usefulness of the classifier. We can see that the classifier can detect the grammar errors in multiple aspects such as verb forms (e.g., *"head"* should be *"heads"* for case 1) and sentence elements (e.g., the predicate is missing for case 3). And the classifier would give the grammatical sentences high scores although they may be unreasonable in logic (e.g., repetitive texts for case 4 and conflicting plot for case 5). Finally, we filter out about 21.69% and 50.15% examples for ROC/WP, respectively.

B.3 Statistics

We show the statistics of the discrimination test set and the invariance test set in AUTOS in Table 16 and Table 17, respectively.

Types	Conjunction, Preposition, Adverb	Noun, Verb, Adjective
Negated	no, not, never, neither, hardly, unlikely, rarely, seldom, impatiently, uncertainly (incomplete listing, 215 in total)	none, nobody, nothing, disable, disagree, disappear, illegal, inability, inactive, unhappy, unfortunately (incomplete listing, 164 in total)
Causality-related	so, because, since, therefore, why	cause, reason, result, effect, purpose, aim, sake, consequence, causal
Time-related	after, before, previously, simultaneously, currently, meanwhile, then, now, ever, again, once, anytime, when, while, never, always, usually, often, sometimes, usually, early, lately, already, forever, ago, yesterday, today, tomorrow	ending, beginning, previous, simultaneous, current, temporary, contemporary, temporal, second, minute, hour, day, month, year, century, past, future, present, delay, night, evening, morning, afternoon, noon, morning

Table 12: Negated words, causality-related words, time-related words which are used to create test examples within the aspects "Consistency", "Causal Relationship" and "Temporal Relationship", respectively.

Subj	Obj	Poss (A)	Poss (N)	Ref
i	me	my	mine	myself
we	us	our	ours	ourselves
you	you	your	yours	yourself
you	you	your	yours	yourselves
he	him	his	his	himself
she	her	her	hers	herself
it	it	its	its	itself
they	them	their	theirs	themselves

Table 13: Personal pronouns which are used to create test examples within the aspect "Character Behaviour". Each row specifies one type of person, which has five forms: **subj**ective pronouns, **obj**ective pronouns, **poss**essive **a**djectives, **poss**essive **n**ouns and **ref**lexive pronouns.

Original Sentences	Paraphrases	M	B
I filled it with the sodas.	I put music into the world and enjoy it.	0.05	0.40
He went several more miles out of his way.	He has made kilometers more.	0.16	0.26
She screamed loudly to attract the attention of her audience.	She yelled out loud for the attention of the public.	0.42	0.45
He hired an attorney.	He employed a lawyer.	0.57	0.40
She watched a video of the play later.	She later watched a video of the play.	0.75	0.89

Table 14: Examples for paraphrase generation. **M** and **B** mean the MoverScore and BLEU-1 between the paraphrases and the original sentences, respectively.

Cases	S
1. She *head* to the city.	0.07
2. A strange elderly woman *and* called his name.	0.20
3. They walked home several more times *whenever that*.	0.41
4. One day Mary needed to leave the airport . She had no idea on how to get a taxi though. Asking for some help she learned about lyft. She had no idea how to get a taxi. Within a hour she was at home, happy with her decision.	0.66
5. Jack was invited to a holiday party. He wanted to bring his hostess a gift. But he had no clue what! Before googling, he decided on a bottle of wine . his hostess was very pleased with it.	0.95

Table 15: Examples for the grammaticality classifier. The examples are sentences or stories selected from the incoherent examples of the discrimination test set of AUTOS. **S** means the classifier score$\in [0, 1]$ (1 is the best). The *italic* words are ungrammatical, and the underlined ones are unreasonable in logic but grammatical.

Datasets	Coherent		Incoherent	
	Input	**Story**	**Input**	**Story**
ROC	8.76	39.28	8.88	40.39
WP	30.02	235.83	30.28	228.04

Table 16: Statistics of the discrimination test set in AUTOS. **Input** and **Story** is the average number of tokens in the inputs and stories. **Coherent** means the coherent examples which are selected from the human-written stories. **Incoherent** means the incoherent examples which are automatically constructed by perturbing the human-written stories.

Datasets	Human		Dis	
	Input	**Story**	**Input**	**Story**
ROC	9.03	40.66	9.23	40.57
WP	29.65	211.19	29.60	234.40

Table 17: Statistics of the invariance test set in AUTOS. **Input** and **Story** is the average number of tokens in the inputs and stories. **Human** and **Dis** means the human-written coherent stories and incoherent samples (sampled from the discrimination test set) to be perturbed, respectively.