Evaluating Semantic Accuracy of Data-to-Text Generation with Natural Language Inference

Ondřej Dušek and **Zdeněk Kasner**
Charles University, Faculty of Mathematics and Physics
Institute of Formal and Applied Linguistics
Prague, Czechia
{odusek,kasner}@ufal.mff.cuni.cz

Abstract

A major challenge in evaluating data-to-text (D2T) generation is measuring the semantic accuracy of the generated text, i.e. checking if the output text contains all and only facts supported by the input data. We propose a new metric for evaluating the semantic accuracy of D2T generation based on a neural model pretrained for natural language inference (NLI). We use the NLI model to check textual entailment between the input data and the output text in both directions, allowing us to reveal omissions or hallucinations. Input data are converted to text for NLI using trivial templates. Our experiments on two recent D2T datasets show that our metric can achieve high accuracy in identifying erroneous system outputs.

1 Introduction

Neural models may reduce the effort for building natural language generation (NLG) systems and produce very natural outputs, at the cost of limited control over the model outputs. State-of-the-art neural D2T models are prone to omitting or hallucinating facts (Gehrmann et al., 2018; Castro Ferreira et al., 2019; Dušek et al., 2020), which restricts their real-world deployment. Recognizing these errors is thus essential for proper system evaluation and further research in D2T generation.

In general, evaluating the semantic accuracy of D2T generation outputs requires full natural language understanding. Minor changes in wording may cause major differences in the meaning of the text, making it difficult for handcrafted heuristics to cover all edge cases. Human evaluation, on the other hand, is expensive and difficult to scale.

We note that the task of checking if a generated sentence includes/entails a particular fact is very close to the task of natural language inference (NLI). NLI is a sequence classification task which takes two inputs—a *hypothesis* and a *premise*—and produces one of the possible outputs: the hypothesis is *entailed* by (follows from) the premise, *contradicts* the premise, or their relation is *neutral*. Recently, neural models for NLI (Zhang et al., 2020b; Liu et al., 2019a,b) reached near-human levels of performance and NLI was used for evaluating the output of abstractive summarization systems (Maynez et al., 2020).

This brings a question: Can we use an NLI model for evaluating the semantic accuracy of D2T outputs? The main idea of our method is to check with a general pretrained NLI model if the semantic information implied by the input data and the generated text is equal. We achieve this by using the NLI model to check for *entailment* in two directions: By inferring input facts from the generated text we can check for *omissions*, while the other direction allows us to check for *hallucinations*.[1] For instance, consider the two input facts from Figure 1: *(Blue Spice | eat_type | pub)*, *(Blue Spice | area | riverside)* and the generated text: "You can bring your kids to Blue Spice in the riverside area." A NLI system should detect that the first fact is not entailed by the text (there is no mention of Blue Spice being a pub), but the text is also not entailed by the facts (the information about kids is hallucinated).

Applying NLI for the D2T task brings a problem: The hypothesis for the standard NLI task is a natural language text, but the input for D2T generation is structured. However, we show that we can easily sidestep this issue by transforming the data into text using a trivial template for each fact.

---

[1] This check in both directions is appropriate for D2T tasks that do not include content selection, which are the focus of our experiments in this paper. If the generator is supposed to select just some of the input facts to verbalize (cf. e.g. Wiseman et al., 2017), we can either only check for hallucinations or, if the content selection is explicit, perform a two-way check with the selected facts provided.

We demonstrate that even without any human references or in-domain training and with minimal handcrafting, our approach achieves high accuracy (>90%) on the E2E Challenge data (Dušek et al., 2020), competitive with scripts specifically handcrafted for the domain, and produces useful results (>75% accuracy) on the more challenging WebNLG dataset (Gardent et al., 2017). A manual error analysis shows that some instances marked as errors were in fact assessed correctly by our metric; we also identified a few major sources of errors that can be mitigated by in-domain tuning. The experimental code for our metric is now available on GitHub.[2]

2   Related Work

Automatic Evaluation of NLG   NLG outputs were traditionally evaluated by reference-based metrics measuring *n*-gram overlap with a reference, such as BLEU (Papineni et al., 2002), ROUGE (Lin, 2004) and METEOR (Lavie and Agarwal, 2007). Alternative, referenceless quality estimation metrics based on language model scores (Kann et al., 2018) or linguistic features (Tian et al., 2018) focus on fluency and do not consider semantic accuracy. Recent works try to estimate NLG output quality with finetuned pretrained models (Zhou and Xu, 2020; Zhang et al., 2020a; Sellam et al., 2020). The score from these models can capture some aspects of semantic accuracy, but only implicitly.

Semantic Accuracy   To our knowledge, there is no generally accepted automatic metric for explicitly measuring semantic accuracy of NLG outputs. The closest commonly used metric is the *slot error rate*, which is typically based on pattern matching tailored for a given dataset (Reed et al., 2018; Mi et al., 2019; Dušek et al., 2020). Recently, Goodrich et al. (2019) introduced a metric based on training a neural model on named-entity recognition and fact extraction.

Faithful NLG   Some recent neural NLG systems train specifically for semantic accuracy (Nie et al., 2019; Tian et al., 2019; Kedzie and McKeown, 2019). Similarly to us, Harkous et al. (2020) use a pretrained neural model as a classifier to detect inaccurate output, finetuning the classifier on manually augmented domain-specific data.

Unlike previous works, we use a pretrained neural model finetuned for NLI which we do not fur-

ther train on any domain-specific data.

3   Method

3.1   NLI Model

We use pretrained RoBERTa (Liu et al., 2019b) as implemented in the Transformers library (Wolf et al., 2020) for our NLI model. Specifically, we use the `roberta-large-mnli`[3] checkpoint, which was finetuned on the MultiNLI dataset (Williams et al., 2018). We use the model *as is*, without any further training. Given a premise text and a hypothesis text, the NLI model produces a probability distribution over three results: *contradiction*, *neutral* and *entailment* (cf. Section 1). We consider a NLI check as passed if the probability for *entailment* is the highest of the three.

3.2   Data Preparation

The input to our metric is a set of facts (the input for a D2T system) and the corresponding verbalization of these facts (the output from a D2T system). In our setup, the facts are RDF-like triples in the *subject-predicate-object* form.

We convert each triple to natural language using a trivial template. We consider two cases:

(1) *Default:* The templates can be handcrafted or extracted from the NLG systems' training data for each predicate.

(2) *Backoff:* We use only a single, universal "backoff" template for all the facts, in the form: *The <predicate> of <subject> is <object>*.

Hereinafter, a *fact* refers to a template filled with the values from the triple.

3.3   Evaluation Process

The generated text is said to be correct if it mentions *all* and *only* the input facts. We check if the text contains any omissions or hallucinations in two steps (see Figure 1 for an example):

(1) To check for omissions, we use the whole generated text as a premise and sequentially feed each fact as a hypothesis to the NLI model. Any failed NLI check is considered an omission. While we could use all concatenated facts in a single NLI check, our approach gives us further information about which facts are omitted.

(2) To check for hallucinations, we use a concatenation of all facts as a premise and feed the generated text as a hypothesis to the NLI model. If this NLI check fails, the text is considered to

---

Figure 1: An example of evaluating the output from a D2T system with our metric. The generated text is used as a *premise* (*P*) to check for omissions and as a *hypothesis* (*H*) to check for hallucinations. The NLI model generates probabilities for *contradiction* (*C*), *neutral* (*N*) and *entailment* (*E*).

contain hallucination. This step cannot be split into simpler NLI checks.

The final output of our metric is either 4-way (denoted as FINE): *OK* (i.e., all NLI checks passed), *omission*, *hallucination* or *omission+hallucination* (based on the failed checks), or 2-way (denoted as ROUGH) where the latter three results are collapsed into *not_OK*. The FINE 4-way output is more useful for system evaluation (we can distinguish whether the system tends to hallucinate or omit information). The ROUGH 2-way output corresponds more to a usage inside an NLG system for output reranking or filtering: any output that is *not_OK* should be penalized/filtered out. Additionally, we compute a *confidence score* of the model as the minimum of all the entailment probabilities.

4 Experimental Setup

We experiment with two recent English data-to-text datasets with a triple-like format: WebNLG (Gardent et al., 2017) and E2E (Novikova et al., 2017).[4] Since both of them were used in shared tasks, sets of system outputs and measures of semantic accuracy are available (see Supplementary for details).

For WebNLG, we compare our metric with crowdsourced human ratings of semantic adequacy (Shimorina et al., 2019). Human annotators used a three-point Likert scale (1 = Incorrect, 2 = Medium, 3 = Correct) and answers are averaged over multiple annotators. In our experiments discussed in Section 5.1, we consider a sentence correct if it achieved human rating 2.5 or higher (we also tried a threshold of 2.0, with slightly worse results).

For the E2E dataset, the challenge results were

checked for semantic accuracy using a handcrafted automatic script (Dušek et al., 2020),[5] we therefore use this automatic script as the ground truth for evaluating our metric in Section 5.2. We further use small sets of system outputs and human-written texts with expert annotation (provided by Dušek et al., 2019) to evaluate our approach against gold-standard annotation and to compare to existing semantic accuracy classifiers for E2E data in Section 5.3.

We evaluate the *Default* and *Backoff* approaches to acquiring templates as described in Section 3.2. The *Default* setup works with one custom template per predicate type. For WebNLG, we obtained templates by delexicalizing human references for single-triple examples from WebNLG training data.[6] For E2E, we handcrafted 8 templates. The templates are filled with values from individual input triples and concatenated for multi-triple inputs as described in Section 3.3.

5 Results Analysis

We evaluate our metric in terms of accuracy, precision, recall, and F1-measure (where *not_OK* samples are treated as positive since we focus on detecting errors). We additionally perform a manual error analysis on a random sample of 100 error examples for each dataset, i.e. examples where our metric gave a different assessment from the ground truth (provided by crowdsourced annotation for WebNLG and by a handcrafted classification script

---

[4]E2E data use attribute-value pairs relating to a restaurant; we convert them to triples where the restaurant is the subject.

[5]While the E2E challenge did include crowdsourced evaluation of semantic accuracy, the results were unreliable, overestimating the number of errors (Dušek et al., 2020). Note that unlike our metric, such a handcrafted approach to evaluating semantic accuracy is only viable for limited domains such as E2E.

[6]For each predicate, we choose randomly if more templates are found and use the backoff if no templates are found.

133

	A	R	P	F1	ρ
Default	0.775	0.772	0.796	0.784	0.628
Backoff	0.768	0.760	0.793	0.776	0.637

Table 1: WebNLG dataset results, compared to crowd-sourced human ratings (A = accuracy, R = recall, P = precision, F1 = F-measure, ρ = Spearman correlation of confidence scores with human scores).

	Af	Ar	R	P	F1
Default	0.911	0.933	0.895	0.910	0.903
Backoff	0.846	0.874	0.913	0.768	0.834

Table 2: E2E dataset results, compared to the automatic evaluation script (Af = FINE accuracy, Ar = ROUGH accuracy, R = recall, P = precision, F1 = F-measure).

for E2E as described in Section 4). In general, the results are high above the random baseline (0.5 for the ROUGH metric and 0.25 for the FINE metric) but differ between the datasets, which we discuss below.

5.1 WebNLG Analysis

The overall scores for the WebNLG dataset are summarized in Table 1. To further check whether the size of the input affects performance, we computed Spearman correlation of the number of input triples with metric errors. The resulting very low value of -0.05 ($p = 0.02$, *Default* setting) shows that the metric holds its performance even for more complex WebNLG examples.

On the other hand, the overall scores show that our metric deviates quite a lot from the human judgments. Our manual error analysis indicates several reasons for that (see Supplementary for examples): (1) The annotation is somewhat noisy and using a threshold is not ideal—many correctly rendered outputs do not reach the 2.5 threshold (while some incorrect ones do). (2) Imprecise templates can confuse the NLI (e.g., for the predicate *nationality*, our extracted template is *<subj> was <obj>*, which works well with values such as *French*, but not with *United States*). This is currently a weak point of our metric, as illustrated by the very small performance difference between the *Default* and *Backoff* setups; however, the issue can be mitigated by a better selection of the templates from training data, e.g. using language-model scoring. (3) The human annotators tend to give lower scores to accurate but ungrammatical or poorly organized texts. Our metric tends to rate these texts as *OK*. Overall,

our re-examination shows that almost half of the error examples (42 out of 100) were in fact correctly classified by our metric (i.e. their crowdsourced human annotation was incorrect), so the true performance is most likely higher than the reported numbers.

The Spearman correlation of our model's confidence scores with the average human scores is around 0.63 ($p <$1e-10). This is similar to n-gram-based metrics on this data (Shimorina, 2018 reports 0.59 for BLEU and 0.73 for METEOR), but unlike these metrics, our approach does not require human-written reference texts.

5.2 E2E Analysis

The results for the E2E dataset (shown in Table 2) are very good compared to the WebNLG dataset, with over 90% agreement with the handcrafted script. This can be attributed to lower lexical variability and less noisy texts, as well as to the better quality of the handcrafted templates (the difference between the *Default* and *Backoff* setups is much more pronounced here). Again, we observe only a very slight drop in performance for more complex E2E inputs (Spearman correlation of metric errors with the number of input triples is -0.08, $p <$1e-10 for the *Default* setting).

The main issues identified by our error analysis are: (1) Problems in the interpretation of some values, e.g., *price range=less than £20* is verbalized as "cheap" or *family-friendly=no* as "adult-only". These cases are classified as *not_OK* by the NLI model. (2) Missing or over-greedy patterns in the slot error script, causing annotation errors. (3) Edge cases: some expressions cannot be interpreted in a straightforward way, e.g. "high restaurant" for *pricerange=high* is deemed OK by the NLI but not by the slot error script. (4) Expressions in the outputs that do not correspond to input facts, such as "with full service", are considered hallucinations by the NLI, but ignored by the slot error script. Again, we consider about half of the error examples (45 out of 100) as correctly classified by our metric (see Supplementary for details), and thus our metric's performance is probably higher than the reported values due to erroneous annotation from the handcrafted script.

5.3 E2E MR Classifier Comparison

We used expert-annotated E2E data samples (cf. Section 4) to compare our approach to other accuracy classifiers in the E2E domain:

	Human-written (E2E training set)					System outputs (TGen)				
	Af	Ar	R	P	F1	Af	Ar	R	P	F1
Slug2Slug aligner	0.685	0.765	0.550	0.800	0.652	0.995	1.000	1.000	1.000	1.000
E2E slot error script	0.820	0.885	1.000	0.777	0.874	0.995	0.995	1.000	0.950	0.974
TGen reranker	0.110	0.435	0.975	0.413	0.579	0.220	0.278	1.000	0.116	0.208
Default	0.600	0.700	0.625	0.625	0.625	0.978	0.978	0.947	0.837	0.888
Backoff	0.530	0.640	0.675	0.540	0.600	0.833	0.833	0.974	0.359	0.525

Table 3: Semantic classifiers evaluated on expert human annotation on E2E data (see Table 2 for metrics legend).

- **Slug2Slug slot aligner** (Juraska et al., 2018) is based on keyword matches. It is carefully tuned but not designed to detect hallucination; it only checks for presence of facts from the input MR.
- **E2E slot error script** (used in Section 5.2) is based on regular expressions; it is also able to detect irrelevant facts.
- **TGen reranker** is an LSTM-based model trained on the E2E training data to rerank outputs of the TGen system (Dušek and Jurčíček, 2016) based on their semantic accuracy.

The results for all classifiers (in Table 3) are much weaker on human-written data, which exhibit much more variability than system outputs. The TGen reranker is very weak when required to detect all facts properly. Our approach is slightly less precise than both handcrafted scripts, but the difference is small on system outputs (97.8% vs. 99.5% accuracy). If we disregard the value *eat-Type=restaurant*, which is generally noisy, we get 76.5% accuracy and 97.6% recall on the human-written data. Moreover, our approach requires much less handcrafting and is more general.

6 Conclusions and Future Work

We described an automatic metric for evaluating semantic accuracy of D2T generation. With just a basic setup, without human references or training and with minimum handcrafting, our metric is able to detect omissions or hallucinations in generated texts, with results competitive with crowdsourced human ratings or handcrafted scripts customized for particular domains.

While our metric seems to scale well to more complex inputs in our experiments on the WebNLG and E2E data, we note that these datasets are still relatively limited. Further experiments are needed to evaluate this approach on long text generation and tasks where content selection is required, which we reserve for future work. We also plan to integrate our metric as a reranker into an NLG system and apply small-scale in-domain finetuning in order to further improve results. Following our findings from the error analysis on WebNLG, which showed that human ratings of semantic correctness are influenced by grammaticality, we would like to investigate the possibilities for combining our metric with a fluency/grammaticality checker (Kann et al., 2018; Tian et al., 2018), as well as ways to better separate these two criteria in human evaluation.

References

Thiago Castro Ferreira, Chris van der Lee, Emiel van Miltenburg, and Emiel Krahmer. 2019. Neural data-to-text generation: A comparison between pipeline and end-to-end architectures. 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 552–562, Hong Kong.

Ondřej Dušek, Jekaterina Novikova, and Verena Rieser. 2020. Evaluating the state-of-the-art of end-to-end natural language generation: The E2E NLG challenge. *Computer Speech & Language*, 59:123–156.

Ondřej Dušek, David M Howcroft, and Verena Rieser. 2019. Semantic Noise Matters for Neural Natural Language Generation. In *Proceedings of the 12th International Conference on Natural Language Generation (INLG 2019)*, pages 421–426, Tokyo, Japan.

Ondřej Dušek and Filip Jurčíček. 2016. Sequence-to-Sequence Generation for Spoken Dialogue via Deep Syntax Trees and Strings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 45–51, Berlin.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017. The WebNLG

challenge: Generating text from RDF data. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 124–133.

Sebastian Gehrmann, Falcon Z. Dai, Henry Elder, and Alexander M. Rush. 2018. End-to-End Content and Plan Selection for Data-to-Text Generation. In *Proceedings of the 11th International Conference on Natural Language Generation*, Tilburg, The Netherlands.

Ben Goodrich, Vinay Rao, Mohammad Saleh, and Peter J. Liu. 2019. Assessing The Factual Accuracy of Generated Text. In *KDD*, Anchorage, AK, USA.

Hamza Harkous, Isabel Groves, and Amir Saffari. 2020. Have your text and use it too! end-to-end neural data-to-text generation with semantic fidelity. *arXiv preprint arXiv:2004.06577*.

Juraj Juraska, Panagiotis Karagiannis, Kevin K. Bowden, and Marilyn A. Walker. 2018. A Deep Ensemble Model with Slot Alignment for Sequence-to-Sequence Natural Language Generation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 152–162, New Orleans, LA, USA.

Katharina Kann, Sascha Rothe, and Katja Filippova. 2018. Sentence-Level Fluency Evaluation: References Help, But Can Be Spared! In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pages 313–323, Brussels, Belgium.

Chris Kedzie and Kathleen McKeown. 2019. A good sample is hard to find: Noise injection sampling and self-training for neural language generation models. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 584–593, Tokyo, Japan.

Alon Lavie and Abhaya Agarwal. 2007. Meteor: An Automatic Metric for MT Evaluation with High Levels of Correlation with Human Judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 228–231, Prague, Czech Republic. Association for Computational Linguistics.

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

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019a. Multi-task deep neural networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4487–4496, Florence, Italy.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv preprint arXiv:1907.11692*.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, Online.

Fei Mi, Minlie Huang, Jiyong Zhang, and Boi Faltings. 2019. Meta-learning for low-resource natural language generation in task-oriented dialogue systems. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 3151–3157.

Feng Nie, Jin-Ge Yao, Jinpeng Wang, Rong Pan, and Chin-Yew Lin. 2019. A simple recipe towards reducing hallucination in neural surface realisation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2673–2679, Florence, Italy.

Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. 2017. The E2E dataset: New challenges for end-to-end generation. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 201–206, Saarbrücken, Germany.

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 of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA.

Lena Reed, Shereen Oraby, and Marilyn Walker. 2018. Can neural generators for dialogue learn sentence planning and discourse structuring? In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 284–295, Tilburg University, The Netherlands.

Thibault Sellam, Dipanjan Das, and Ankur P. 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.

Anastasia Shimorina. 2018. Human vs Automatic Metrics: on the Importance of Correlation Design. In *WiNLP Workshop*, New Orleans, LA, USA.

Anastasia Shimorina, Claire Gardent, Shashi Narayan, and Laura Perez-Beltrachini. 2019. WebNLG challenge: Human evaluation results. Technical report, LORIA.

Ran Tian, Shashi Narayan, Thibault Sellam, and Ankur P Parikh. 2019. Sticking to the facts: Confident decoding for faithful data-to-text generation. *arXiv preprint arXiv:1910.08684*.

Ye Tian, Ioannis Douratsos, and Isabel Groves. 2018. Treat the system like a human student: Automatic naturalness evaluation of generated text without reference texts. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 109–118, Tilburg University, The Netherlands.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana.

Sam Wiseman, Stuart M. Shieber, and Alexander M. Rush. 2017. Challenges in Data-to-Document Generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2243–2253, Copenhagen, Denmark.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. HuggingFace's Transformers: State-of-the-art Natural Language Processing. *arXiv preprint arXiv:1910.03771*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020a. BERTScore: Evaluating Text Generation with BERT. In *ICLR*, Online.

Zhuosheng Zhang, Yuwei Wu, Hai Zhao, Zuchao Li, Shuailiang Zhang, Xi Zhou, and Xiang Zhou. 2020b. Semantics-aware BERT for language understanding. In *Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI-2020)*.

Wangchunshu Zhou and Ke Xu. 2020. Learning to Compare for Better Training and Evaluation of Open Domain Natural Language Generation Models. In *AAAI*, New York, NY, USA.