Automatic Evaluation of Generative Dialogue Systems: An Empirical Study

Cong Feng, Wenge Rong, Jianxin Yang, Haodong Yang,
Yuanxin Ouyang, and Zhang Xiong

School of Computer Science and Engineering, Beihang University, Beijing, China
{congfeng, w.rong, yjx17, yhd125, oyyx, xiongz}@buaa.edu.cn

Abstract. Evaluating generative dialogue systems with automatic metrics is a challenging task. It has been shown that the word-overlap metrics such as BLEU and the word-embedding metrics do not correlate well with human judgements. Furthermore, regardless of the correlation with human judgements, the scores of these metrics do not correlate well with others, which means fine-tuning a dialogue system on one set of metrics may yield inconsistency when testing with another set of metrics. This infeasibility again adds to the ineffectiveness of evaluation dialogue systems. To the end of utilizing metrics at its best, in this research, we offer several pragmatic suggestions on the use of automatic metrics to avoid the issue of metric inconsistency by empirically evaluating a set of automatic evaluation metrics.

Keywords: Automatic Evaluation Metrics, Dialogue Response Generation, Chatbot

1 Introduction

Recently the chat-oriented dialogue systems have seen a boom in the community. These systems are trained to make an appropriate response given a conversational context and can be applied to various applications. One of the fundamental techniques in such dialogue systems is generative model, which can learn language patterns and extract knowledge from the corpus in an unsupervised manner. The research community has made steady progress with the generative models and one of the most widely used techniques is Seq2Seq model [19]. Though generative models have shown promising results, there exist one essential challenge. They incline to give meaningless responses to the questions, e.g., *I don't know*, thereby making their evaluation remain an open problem [8].

Currently most generative dialogue systems are evaluated by borrowing automatic metrics from other tasks, e.g., machine translation, such as the BLEU [12], METEOR [1] and etc. However, the correlations between these borrowed metrics and human judgements were unclear and researchers generally fell back to human evaluation for better accuracy and reliability [16,17]. It is found that those metrics only correlated weakly with human judgements on the non-technical Twitter

corpus and not at all on the technical Ubuntu Dialogue Corpus [8], which calls for new automatic metrics that are more relevant to human judgements.

In this paper, we followed the previous work and further investigated the behaviors of automatic metrics without human judgements. We analyzed both system-level and example-level scores to find out possible correlations among different metrics. In particular, we drew samples from the example-level scores of different metrics and analyze the pairwise correlation of these samples. It is found that some pairs of samples have a high correlation, while other pairs show a much lower one.

Following these observations, we further clustered the metrics based on the pairwise correlations of their corresponding samples. The results show that similar metrics tend to cluster together, indicating a high pairwise correlation within the same group. According to the experiment results, it is argued that the scores given by dissimilar metrics have the risk of high inconsistency, which is yet another pitfall of using automatic metrics. Therefore, it is strongly recommended to choose consistent metrics for the ease of comparison across different settings.

2 Related Work

In dialogue response generation, to address the issue of low metric-human correlation, various semantic-based methods have been proposed. For example, the ADEM metric proposed models the human judgements with a feed-forward neural network [9]. The RUBER metric takes an asymmetric approach w.r.t the response-context pair and response-reference pair, where the former is modeled by a neural network and the latter is measured by an embedding-based metric. They combined two scores with various heuristics and achieved improvements on a Chinese corpus [20]. In brief, correlation with human judgements has been the supervised signal guiding the evolution of automatic metrics.

Other popular metrics include perplexity, which is generally used to evaluate statistical language models. Vinyals et al. revealed that their Seq2Seq dialogue model achieved a much lower perplexity than the n-gram baseline, but they also admitted the drawbacks of using such a metric [22]. Serban et al. also used perplexity to evaluate their models [15], along with other metrics. However they were also not clear about how well these metrics accounted for the grammatical correctness and semantic coherence of the responses.

Generally, automatic metrics have been constantly doubted of their capability to reflect human judgements. In one of the earliest attempts to the response generation problem, Ritter et al. made an initial examination on the suitability of BLEU to this field leveraging the human data they collected [13]. They found that the BLEU scores were very low even on the system level and the correlation was modest. Similarly, Shang et al. also argued that BLEU did not apply as the reasonable responses evaluation [17].

An extensive study of metric-human correlation was conducted by Liu et al. [8]. It was revealed that although these metrics could distinguish state-of-art models from baseline models, none of them correlated highly with human scores.

Their work has left us two questions: 1) How can we improve the metric-human correlation? 2) Why do the existing metrics correlate badly with human judgements? Previous works that proposed enhanced metrics endeavored to answer the first question, while we try to shed some light on the second one. In particular, we are curious about what can we learn when scores from different metrics are put together and compared across various settings.

3 Experiment Configuration

Our experiment essentially involves training multiple generative models on multiple datasets, and then measure their performances with various automatic metrics. As stated, our study does not involve human evaluation. We also do not consider retrieval models as we mainly focuse on generative models. Due to the time and resource constrains, we limit our scope to three commonly used models and three datasets.

3.1 Metrics

Here are a list of popular metrics used in dialogue response evaluation included in this research.

1) BLEU [12] is a classical metric for machine translation that reports a high correlation with human scores on the system level. It owes the quality of a hypothesis to its similarity to multiple references. It computes the geometric mean of consecutive orders of n-gram precision, multiplied by a brevity penalty.

2) METEOR [1] is a metric proposed to address several issues with BLEU. It applies multiple stages of unigram matching to the hypothesis and reference, each using a different criterion, such as exact matching, WordNet synonyms, and paraphrases. An alignment is then created from these unigram matches. The score is based on the F1 of the alignment and a penalty to shorter matches.

3) ROUGE [6] is a family of metrics for automatic summarization. It is based on the F1 score and can integrate different counting units, e.g., n-gram statistics and the longest common subsequence.

4) Embedding Average is a metric based on word embedding, a distributed approach to the meaning of words [11]. The embedding of a sentence is defined as the average of the embeddings of its composing words. The similarity of two sentences is then simply defined as the cosine of the corresponding vectors.

5) Vector Extrema [2] composes the sentence embedding by taking the most extreme value (either maximum or minimum) along each dimension from its constitutive words. The intuition is that in the embedding space, common words like function words are pulled towards the origin as they appear in the context of many different words, while informative words are pushed away from the origin in either positive or negative direction, since they tend to appear in more specific context.

6) Greedy Matching [14] is an embedding-based method without calculating a sentence vector. Instead, two sentences under comparison are treated as a

weighted bipartite graph, taking their words as nodes and the embedding cosine of two words $\cos(w, w')$ as the edge weight. The metric is based on a greedy method to solve the optimal matching problem on the weighted graph:

$$G(r, \hat{r}) = \frac{\sum_{w \in r} \max_{\hat{w} \in \hat{r}} \cos(e_w, e_{\hat{w}})}{|r|} \tag{1}$$

$$\textit{Greedy-Matching} = \frac{G(r, \hat{r}) + G(\hat{r}, r)}{2} \tag{2}$$

where r and $\hat{r}$ are the reference and response, respectively.

7) ADEM [9] is an approach based on a concise nueral network. The model first embeds a context c, response $\hat{r}$, and reference r into a low-rank vector space and then predicts the human score with a linear combination of the input features:

$$s(c, r, \hat{r}) = \frac{(c^T M \hat{r} + r^T N \hat{r} - \alpha)}{\beta} \tag{3}$$

M, N are learnable parameters of the network and α, β are constants to normalize the output to a five-point score. The model is trained on a human-annotated conversation corpus to minimize the squared error with L2 regularization:

$$L = \sum_{i=1}^{K} (s_i - h_i)^2 + \gamma \left\| \theta \right\|_2 \tag{4}$$

where K is the number of samples in a batch. s_i and h_i are the model prediction and the human score, respectively. θ is the parameters of the network and γ is the regularization factor.

8) Distinct-N [5] measures the rate of unique n-grams in a sentence. For a sentence, it is the number of unique n-grams divided by the total number of n-grams. It is a token-level measurement of the diversity of a response.

3.2 Models

In our research, three representative models from [16] are selected, namely LSTM, HRED, and VHRED. LSTM is a simple model based on the Long Short-Term Memory [3]. HRED extends the standard Seq2Seq framework by using a hierarchical encoder. VHRED is an extension to HRED that injects randomness into the decoder to achieve higher diversity. As such, they form a hierarchy of architectural sophistication and are ideal baselines of generative models.

1) The LSTM model is a simple generative model with a single RNN acting as both an encoder and a decoder. Note that the generative LSTM model is not autoregressive as the output of the previous time step does not become the input of the current time step.

2) The HRED model [18,15] features a hierarchical encoding mechanism that takes into account the structure of dialogues. It first encodes each sentence with the *utterance encoder* into a fixed-length utterance vector e_u. Then, the utterance

vectors are processed iteratively by the *context encoder* to produce a fixed-length context vector e_d. Finally, the *utterance decoder* takes the context vector and generates the next utterance of the dialogue.

3) The VHRED model [16] extends the HRED model with a variational inference mechanism. Essentially, VHRED injects into the utterance decoder random variables that are sampled from a high-dimension normal distribution, the parameters of which are conditioned on the context vector e_d.

The configurations for different model components are shown in Table 1. We used Adam [4] as our optimizer and applied gradient clipping with a threshold of 1. The learning rate was set to 0.0002 on the Ubuntu corpus and 0.0002 on the others. All the models were trained on a Nvidia GTX for at least one week. We used random sampling when testing.

Table 1: Model Configurations

(a) Utterance Encoder

	#Embedding	#Hidden Units
LSDSCC	400	1000
OpenSubtitles	400	1000
Ubuntu	300	500

(b) Context Encoder

RNN Direction	#Hidden Units
bidirectional	1000
bidirectional	1000
unidirectional	1000

(c) Utterance Decoder

	RNN Type			#Hidden Units		
	HRED	LSTM	VHRED	HRED	LSTM	VHRED
LSDSCC	LSTM	GRU	GRU	1000	2000	1000
OpenSubtitles	LSTM	GRU	GRU	1000	2000	1000
Ubuntu	LSTM	LSTM	LSTM	500	2000	500

3.3 Datasets

In this research, we select three commonly used datasets, the statistics of which are listed in table 2. These datasets represent three common domains in the literature, namely technical support, movie subtitles and online forums.

Table 2: Statistics of the Datasets

	Training Set				Testing Set	
	Vocabulary	#Samples	#Tokens	Turns	#Samples	#Tokens
Ubuntu	20000	448833	45697699	multiple	18920	2045082
OpenSubtitles	23876	11771393	379346841	multiple	14714	474074
LSDSCC	35008	738095	32355628	single	299	10914

1) The Ubuntu Dialogue Corpus [10] is a large-scale multi-turn dyadic technical corpus collected from the Ubuntu channel of an IRC network[1]. It contains a lot of technical symbols, such as filesystem paths, commands, and URLs.

2) The OpenSubtitles dataset [7] is an enormous corpus of movie subtitles. It is collected by the OPUS project [21] from the opensubtitles website[2]. Typically, two consecutive utterances are treated as a context-response pair, as in [22,5].

3) LSDSCC [23] is a domain-specific conversation corpus collected from the movie subreddit of the Reddit forum[3]. It is believed that a more specific domain can help avoid generating universal responses [23].

4 Experimental Study

4.1 System-Level Scores

We first measured the system-level scores for all settings, shown in Table 3. The system-level score reflects the average performance of a model trained on a dataset. For metrics that do not have an explicit system-level definition, we took the arithmetic average of their example-level scores.

Table 3: System-Level Scores

	LSDSCC			OpenSubtitles			Ubuntu		
	HRED	LSTM	VHRED	HRED	LSTM	VHRED	HRED	LSTM	VHRED
ADEM	**2.6178**	2.6127	2.6163	**2.6228**	2.6224	2.6219	2.6353	**2.6381**	2.635
BLEU-1	**0.08**	0.0726	0.0722	0.0672	0.0638	**0.0753**	0.1314	0.1303	**0.1365**
BLEU-2	**0.0264**	0.0181	0.0185	0.0171	0.0153	**0.0264**	0.0362	0.0345	**0.0375**
BLEU-3	**0.0105**	0.0052	0.0066	0.0062	0.0055	**0.0146**	**0.009**	0.007	0.0089
BLEU-4	**0.0053**	0.0	0.0028	0.0024	0.0022	**0.01**	**0.0029**	0.0018	0.0025
Distinct-1	**0.9577**	0.9441	0.9558	**0.973**	0.9714	0.9714	0.9074	**0.9257**	0.9113
Distinct-2	**0.8541**	0.8511	0.8497	**0.8669**	0.8594	0.8665	**0.9013**	0.8603	0.8968
Greedy	**0.3303**	0.3292	0.3267	0.3102	0.2998	**0.3145**	**0.2775**	0.2364	0.273
Average	**0.5532**	0.5467	0.5483	0.5453	0.5295	**0.5485**	**0.574**	0.5205	0.5655
Extrema	**0.2841**	0.2835	0.2814	0.3009	0.2929	**0.3061**	**0.29**	0.2663	0.2875
METEOR	**0.0296**	0.0258	0.0281	0.0248	0.0233	**0.0271**	0.1657	0.1635	**0.166**
ROUGE-1	**0.108**	0.083	0.0978	0.0784	0.075	**0.0872**	0.1644	**0.1836**	0.1683
ROUGE-2	**0.0226**	0.0049	0.0081	0.0053	0.0043	**0.0107**	0.0128	**0.0143**	0.0128
ROUGE-3	**0.0057**	0.0002	0.0035	0.0011	0.0009	**0.0053**	**0.0007**	0.0003	0.0005
ROUGE-4	0.0011	0.0	**0.003**	0.0002	0.0002	**0.0038**	**0.0002**	0.0	0.0001
ROUGE-L	**0.0956**	0.0681	0.0846	0.0742	0.0707	**0.0826**	0.1493	**0.1722**	0.1535
ROUGE-W	**0.0792**	0.0537	0.07	0.066	0.0629	**0.0734**	0.1205	**0.1391**	0.1236
PPL	**32.5599**	32.9229	37.7149	41.6392	34.2724	**33.6867**	**39.178**	46.4061	40.2641
#words	13.1605	14.0067	12.3612	8.807	8.6394	8.7798	23.0646	16.4905	21.2449

[1] https://irclogs.ubuntu.com/
[2] http://www.opensubtitles.org
[3] https://www.reddit.com/r/datasets

The system-level scores of different metrics look quite consistent in terms of the best performing model on a certain dataset. On LSDSCC, for example, the HRED model beats all the other models on all but one metrics, while on OpenSubtitles, the VHRED model wins the best of most of the metrics. The system-level score seems to be able to distinguish state-of-the-art models from baselines, since most of the metrics agree on which model is the best.

However, the consistency among the metrics does not necessarily lead to a high correlation with human judgement, as shown in [8]. Moreover, since the system-level score is calculated by accumulating over the example-level scores, it is possible that the inconsistency on the example-level is hidden away. To reveal the possible inconsistency, we performed analyses on the example-level scores.

4.2 Example-Level Scores

On the example level, a matrix $M \in R^{N \times M}$ is calculated for each model instance, where N is the number of examples and M is the number of metrics. Each row r_i of the matrix is the scores of all metrics for an example e_i, while each column c_j is the values of a metric s_j computed for all examples. All the metrics share the same set of examples within a matrix. Thus, the correlation of any two columns c_i, c_j can be understood as to how much the corresponding metrics s_i, s_j agree on their shared examples. We compute the pairwise correlations of the metrics and highlight the degree of correlation with heatmaps, as shown in Fig. 1. We used Pearson's r and all the results are statistically significant.

Each subfigure in Fig. 1 was plotted from the correlation matrix for a model instance. The color of the cells represents the degree of correlation between the row and column labels, with red, blue, and white stand for positive, negative and zero correlation, respectively. One can observe red regions divided by white or blue lines from these plots, showing the signs of clustering.

To better observe the agreement and disagreement among the metrics, we applied hierarchical clustering to the metrics based on their correlations and the results are shown in Fig. 2. A hierarchical clustering algorithm starts with a forest of nodes and iteratively merges them into larger clusters until the root cluster is created. We used the following node-level distance $\mathit{dist}(\cdot,\cdot)$ and cluster-level distance $d(\cdot,\cdot)$:

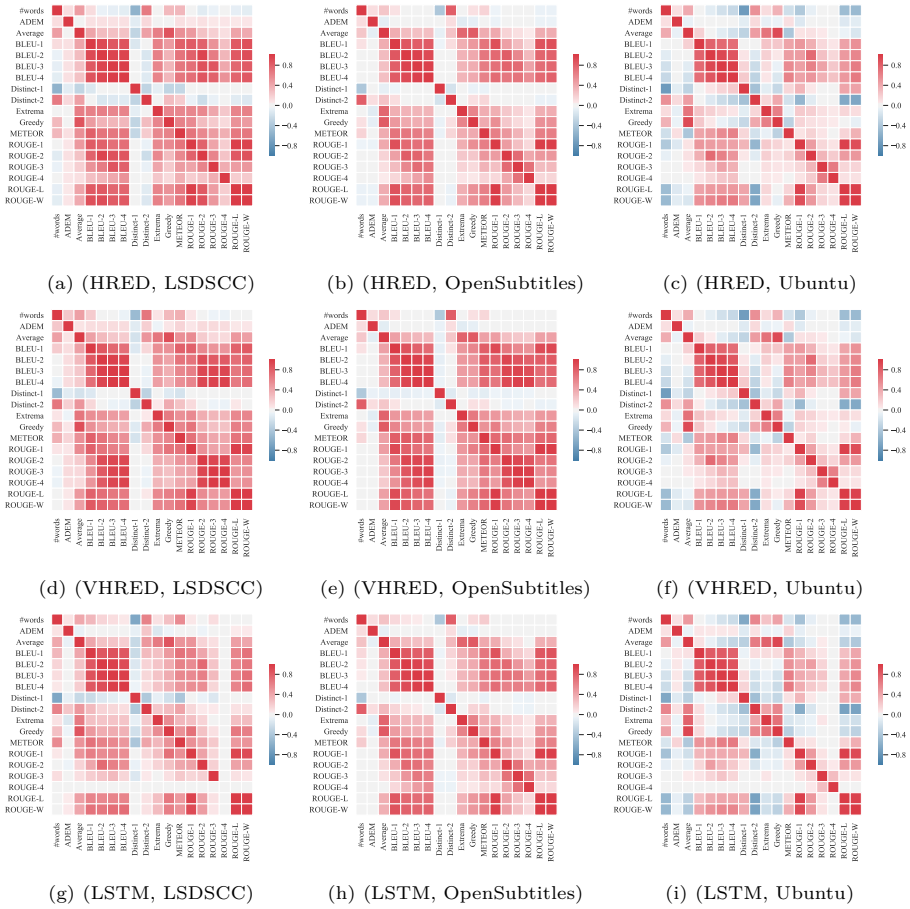
$$\mathit{dist}(i,j) = 1 - \mathit{corr}(i,j) \tag{5}$$

$$d(u,v) = \frac{\sum_{i,j} \mathit{dist}(i,j)}{|u| \cdot |v|} \tag{6}$$

where i and j are points in cluster u and v, respectively. $|u|$ and $|v|$ are the cardinalities of cluster u and v, respectively[4]. Again the correlation is based on Pearson's r.

From Fig. 2, we observe a hierarchy of agreement among different metrics. The clusters created in the earlier iterations appear closer to the leaves of the

[4] This is also known as the average method.

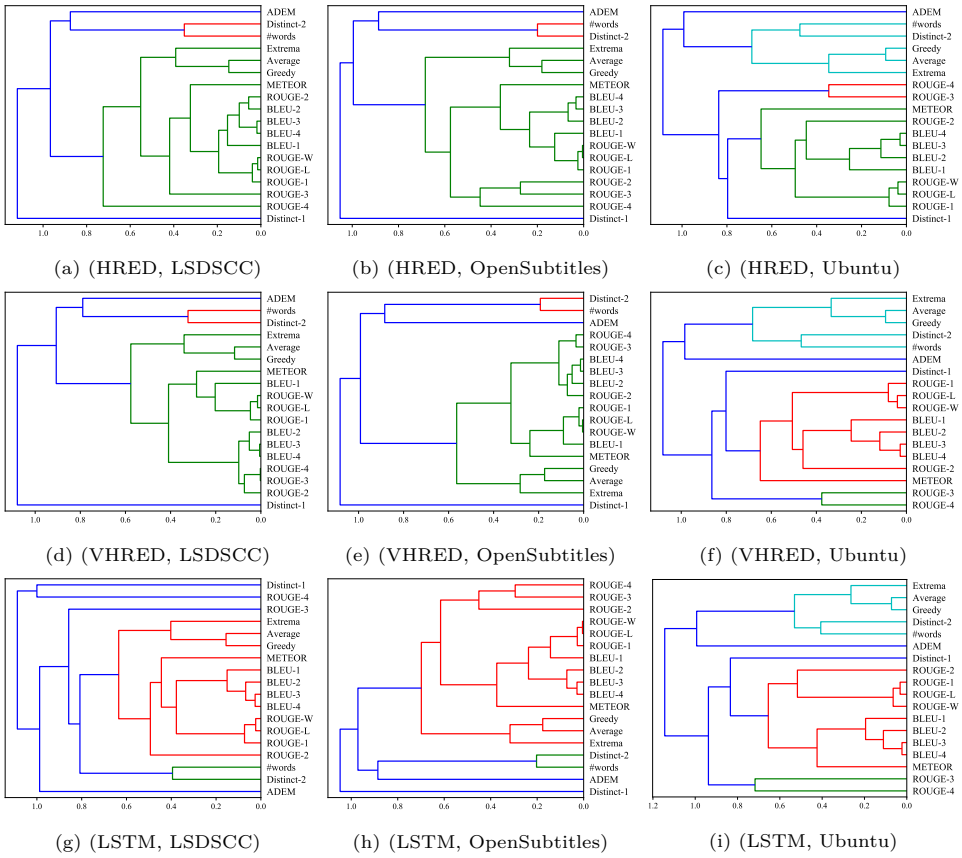


(a) (HRED, LSDSCC) (b) (HRED, OpenSubtitles) (c) (HRED, Ubuntu)

(d) (VHRED, LSDSCC) (e) (VHRED, OpenSubtitles) (f) (VHRED, Ubuntu)

(g) (LSTM, LSDSCC) (h) (LSTM, OpenSubtitles) (i) (LSTM, Ubuntu)

Fig. 1: Heatmaps of Pearson's r



(a) (HRED, LSDSCC) (b) (HRED, OpenSubtitles) (c) (HRED, Ubuntu)

(d) (VHRED, LSDSCC) (e) (VHRED, OpenSubtitles) (f) (VHRED, Ubuntu)

(g) (LSTM, LSDSCC) (h) (LSTM, OpenSubtitles) (i) (LSTM, Ubuntu)

Fig. 2: Hierarchical Clusterings with Pearson's r

tree, which indicates higher pairwise correlations or stronger inter-metric agreement. As a cluster grows larger, its children become more loosely connected with greater distances. It is interesting to find that the dendrograms in Fig. 2 share some regular structure, which has an observable correspondence to the category of the metrics. Across all the settings, we generally observe these independent clusters, named after the shared category or representative element:

1) Word-Overlap: a large cluster that contains many subclusters formed by different word-overlap metrics, such as BLEU, ROUGE, and METEOR.

2) Word-Embedding: a small cluster that contains Vector-Average, Vector-Extrema, and Greedy-Matching.

3) ADEM: a standalone cluster formed by ADEM.

4) Distinct-1: a metric that might belong to some cluster or become a standalone cluster depending on the dataset involved.

5) Distinct-2 and #words: a generally observable cluster.

Those clusters confirm our hypothesis on the inconsistency among different metrics. Luckily, there are still some forms of agreement that the same cluster of metrics can reach. Unfortunately, the exact reasons why the clusters are formed this way are unclear to us. Although our experiments have a simple procedure, it should be noted that the mechanism behind each step has a complex nature. For example, the dialogue datasets are intrinsically very diverse and the way the generative models work is not well-understood. Besides, it is also unclear how these metrics reflect the desirable properties of a response. Hence we can only conclude that similar metrics tend to have consistent results on the example-level. The similarity of metrics generally refers to their mechanisms, such as the way to extract overlap units or derive semantics from an utterance, or the way to combine different components.

Specifically, we find that metrics based on word overlap tend to have high pairwise correlations since they all make use of the n-gram statistics somehow. We also find that ADEM does not correlate well with all the other metrics since it has a much higher correlation with human judgement while other metrics do not.

4.3 Qualitative Analysis

Table 4: An Example from OpenSubtitles

Context	You will be in classes does she know that? As if you' re gonna have all this free time to pal around with her.		
Reference	okay sure we will start there.		
Model	LSTM	HRED	VHRED
Response	C all this time.	Her room starts in half 50 percent.	I figured this was gonna be a fun stretch.
ADEM	2.5987	**2.6411**	2.6276
BLEU-1	0	0	0
Distinct-1	1.0	1.0	1.0
Distinct-2	0.75	0.8571	**0.8889**
Greedy	**0.3520**	0.2336	0.3412
Average	0.4979	0.3468	**0.6607**
Extrema	0.2791	0.1341	**0.3995**
METEOR	0	**0.0105**	0
ROUGE-L	0	0	0
ROUGE-W	0	0	0
#words	4	7	**9**

Table 4 serves as an example of the inter-metrics inconsistency. The response from HRED mentions "her room", which is relevant to the subject "she" in the context and the verb "starts" matches the "start" in the reference. The response

from VHRED does not share any token with the context or reference, but it remarks the event mentioned by the context so it is quite a reasonable one.

All three responses share no n-grams with either the context or the reference in terms of exact matching. Thus, all the word-overlap metrics yield a zero value. However, all the word-embedding metrics give non-zero value, making them incompatible with the word-overlap metrics.

In terms of semantic relevance, the responses from both HRED and VHRED are somehow related to the topic of the dialogue, while the response from LSTM looks grammatically incomplete. One will mostly agree that the responses from HRED and VHRED are equally better than that from LSTM. Nonetheless, the ranks given by the word-embedding metrics do not agree with our manual inspection. For example, they all give higher scores to LSTM than HRED.

5 Conclusion and Future Work

In this paper, we followed the work of [8] and try to understand the reasons behind low metric-human correlations. We investigated the system-level scores and leveraged statistical analyses to reveal the inter-metrics correlation on the example level. Our study shows a high consistency of metrics on the system level, as shown by others [8,16,22]. We further show that on the example level, similar metrics tend to score the examples more consistently and the degree of correlations forms a hierarchical cluster.

Intuitively, different metrics judge a dialogue from different angles, while human beings judge it from a quite comprehensive perspective. This might explain why the metrics do not correlate well with human judgements. Our discovery of the example-level correlation-based hierarchical clusterings of metrics is a novel contribution.

Based on the observations, it is recommended to avoid using a set of metrics that have low pairwise correlations since that will make the example-level scores divergent and hard to explain. We also urge against the use of metrics that are known to have poor correlations with human judgements. In the future, we would like to look deeper into the mechanisms behind the metric-human correlations.

References

1. Banerjee, S., Lavie, A.: METEOR: an automatic metric for MT evaluation with improved correlation with human judgments. In: Proceedings of the Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization@ACL 2005. pp. 65–72 (2005)
2. Forgues, Pineau, J., Larcheveque, J., Tremblay, R.: Bootstrapping dialog systems with word embeddings. In: Workshop on Modern Machine Learning and Natural Language Processing (2014)
3. Hochreiter, S., rgen Schmidhuber, J.u.: Long short-term memory. Neural Computation **9**(8), 1735–1780 (1997)
4. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: 3rd International Conference on Learning Representations (2015)

5. Li, J., Galley, M., Brockett, C., Gao, J., Dolan, B.: A diversity-promoting objective function for neural conversation models. In: NAACL HLT 2016. pp. 110–119 (2016)
6. Lin, C.Y.: Rouge: A package for automatic evaluation of summaries. In: Text Summarization Branches Out (2004)
7. Lison, P., rg Tiedemann, J.o.: Opensubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016 (2016)
8. Liu, C..W., Lowe, R., Serban, I., Noseworthy, M., Charlin, L., Pineau, J.: 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. pp. 2122–2132 (2016)
9. Lowe, R., Noseworthy, M., Serban, I.V., Gontier, N.A.., Bengio, Y., Pineau, J.: Towards an automatic turing test: Learning to evaluate dialogue responses. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. pp. 1116–1126 (2017)
10. Lowe, R., Pow, N., Serban, I., Pineau, J.: The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. In: Proceedings of the SIGDIAL 2015 Conference. pp. 285–294 (2015)
11. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. In: 1st International Conference on Learning Representations (2013)
12. Papineni, K., Roukos, S., Ward, T., Zhu, W..J.: Bleu: a method for automatic evaluation of machine translation. In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics. pp. 311–318 (2002)
13. Ritter, A., Cherry, C., Dolan, W.B.: Data-driven response generation in social media. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. pp. 583–593. Association for Computational Linguistics (2011)
14. Rus, V., Lintean, M.C.: A comparison of greedy and optimal assessment of natural language student input using word-to-word similarity metrics. In: Proceedings of the Seventh Workshop on Building Educational Applications Using NLP. pp. 157–162 (2012)
15. Serban, I.V., Sordoni, A., Bengio, Y., Courville, A.C., Pineau, J.: Building end-to-end dialogue systems using generative hierarchical neural network models. In: Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence. pp. 3776–3784 (2016)
16. Serban, I.V., Sordoni, A., Lowe, R., Charlin, L., Pineau, J., Courville, A.C., Bengio, Y.: A hierarchical latent variable encoder-decoder model for generating dialogues. CoRR **abs/1605.06069** (2016)
17. Shang, L., Lu, Z., Li, H.: Neural responding machine for short-text conversation. 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. pp. 1577–1586 (2015)
18. Sordoni, A., Bengio, Y., Vahabi, H., Lioma, C., Simonsen, J.G., Nie, J..Y.: A hierarchical recurrent encoder-decoder for generative context-aware query suggestion. In: Proceedings of the 24th ACM International Conference on Information and Knowledge Management. pp. 553–562 (2015)
19. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014. pp. 3104–3112 (2014)

20. Tao, C., Mou, L., Zhao, D., Yan, R.: RUBER: an unsupervised method for automatic evaluation of open-domain dialog systems. In: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence. pp. 722–729 (2018)
21. Tiedemann, J.: News from OPUS—A Collection of Multilingual Parallel Corpora with Tools and Interfaces, vol. 5, pp. 237–248 (2009)
22. Vinyals, O., Le, Q.V.: A neural conversational model. CoRR **abs/1506.05869** (2015)
23. Xu, Z., Jiang, N., Liu, B., Rong, W., Wu, B., Wang, B., Wang, Z., Wang, X.: LSDSCC: a large scale domain-specific conversational corpus for response generation with diversity oriented evaluation metrics. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 2070–2080 (2018)