Re-evaluating Evaluation in Text Summarization

Manik Bhandari, Pranav Gour, Atabak Ashfaq, Pengfei Liu, Graham Neubig
Carnegie Mellon University
{mbhandar,pgour,aashfaq,pliu3,gneubig}@cs.cmu.edu

Abstract

Automated evaluation metrics as a stand-in for manual evaluation are an essential part of the development of text-generation tasks such as text summarization. However, while the field has progressed, our standard metrics have not – for nearly 20 years ROUGE has been the standard evaluation in most summarization papers. In this paper, we make an attempt to *re-evaluate the evaluation method* for text summarization: assessing the reliability of automatic metrics using *top-scoring system outputs*, both abstractive and extractive, on *recently popular datasets* for both system-level and summary-level evaluation settings. We find that conclusions about evaluation metrics on older datasets do not necessarily hold on modern datasets and systems. We release a dataset of human judgments that are collected from 25 top-scoring neural summarization systems (14 abstractive and 11 extractive): https://github.com/neulab/REALSumm

1 Introduction

In text summarization, *manual evaluation*, as exemplified by the Pyramid method (Nenkova and Passonneau, 2004), is the gold-standard in evaluation. However, due to time required and relatively high cost of annotation, the great majority of research papers on summarization use exclusively automatic evaluation metrics, such as ROUGE (Lin, 2004) , JS-2 (Louis and Nenkova, 2013), S3 (Peyrard et al., 2017), BERTScore (Zhang et al., 2020), MoverScore (Zhao et al., 2019) etc. Among these metrics, ROUGE is by far the most popular, and there is relatively little discussion of how ROUGE may deviate from human judgment and the potential for this deviation to change conclusions drawn regarding relative merit of baseline and proposed methods. To characterize the relative goodness of evaluation metrics, it is necessary to perform *meta-evaluation* (Graham, 2015; Lin and Och, 2004), where a dataset annotated with human judgments (e.g. TAC[1] 2008 (Dang and Owczarzak, 2008)) is used to test the degree to which automatic metrics correlate therewith.

However, the classic TAC meta-evaluation datasets are now 6-12 years old[2] and it is not clear whether conclusions found there will hold with modern systems and summarization tasks. Two earlier works exemplify this disconnect: (1) Peyrard (2019) observed that the human-annotated summaries in the TAC dataset are mostly of lower quality than those produced by modern systems and that various automated evaluation metrics strongly disagree in the higher-scoring range in which current systems now operate. (2) Rankel et al. (2013) observed that the correlation between ROUGE and human judgments in the TAC dataset decreases when looking at the best systems only, even for systems from eight years ago, which are far from today's state-of-the-art.

Constrained by few existing human judgment datasets, it remains unknown how existing metrics behave on current top-scoring summarization systems. In this paper, we ask the question: does the rapid progress of model development in summarization models require us to *re-evaluate* the evaluation process used for text summarization? To this end, we create and release a large benchmark for meta-evaluating summarization metrics including:

- **Outputs** from 25 top-scoring extractive and abstractive summarization systems on the `CNN/DailyMail` dataset.
- **Automatic evaluations** from several evaluation metrics including traditional metrics (e.g. ROUGE) and modern semantic matching metrics (e.g. BERTScore, MoverScore).

[1] https://tac.nist.gov/

[2] In TAC, summarization was in 2008, 2009, 2010, 2011, 2014. In 2014, the task was biomedical summarization.

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 9347–9359,
November 16–20, 2020.

Ability of metrics to	Observations on existing human judgments (TAC)	Observations on new human judgments (CNNDM)
Exp-I: evaluate all systems? (Sec. 4.1)	MoverScore and JS-2 outperform all other metrics.	ROUGE-2 outperforms all other metrics. MoverScore and JS-2 performs worse both in extractive (only achieved nearly 0.1 *Pearson* correlation) and abstractive summaries.
Exp-II: evaluate top-k systems? (Sec. 4.2)	As k becomes smaller, ROUGE-2 de-correlates with humans.	For extractive and abstractive systems, ROUGE-2 highly correlates with humans. For evaluating a mix of extractive and abstractive systems, all metrics de-correlate.
Exp-III: compare 2 systems? (Sec. 4.3)	MoverScore and JS-2 outperform all other metrics.	ROUGE-2 is the most reliable for abstractive systems while ROUGE-1 is most reliable for extractive systems.
Exp-IV: evaluate summaries? (Sec. 4.4)	(1) MoverScore and JS-2 outperform all other metrics. (2) Metrics have much lower correlations when evaluating summaries than systems.	(1) ROUGE metrics outperform all other metrics. (2) For extractive summaries, most metrics are better at evaluating summaries than systems. For abstractive summaries, some metrics are better at summary level, others are better at system level.

Table 1: Summary of our experiments, observations on existing human judgments on the TAC, and contrasting observations on newly obtained human judgments on the CNNDM dataset. Please refer to Sec. 4 for more details.

- **Manual evaluations** using the lightweight pyramids method (Shapira et al., 2019), which we use as a gold-standard to evaluate summarization systems as well as automated metrics.

Using this benchmark, we perform an extensive analysis, which indicates the need to re-examine our assumptions about the evaluation of automatic summarization systems. Specifically, we conduct four experiments analyzing the correspondence between various metrics and human evaluation. Somewhat surprisingly, we find that many of the previously attested properties of metrics found on the TAC dataset demonstrate different trends on our newly collected CNNDM dataset, as shown in Tab. 1. For example, MoverScore is the best performing metric for evaluating summaries on dataset TAC, but it is significantly worse than ROUGE-2 on our collected CNNDM set. Additionally, many previous works (Novikova et al., 2017; Peyrard et al., 2017; Chaganty et al., 2018) show that metrics have much lower correlations at comparing summaries than systems. For extractive summaries on CNNDM, however, most metrics are better at comparing summaries than systems.

Calls for Future Research These observations demonstrate the limitations of our current best-performing metrics, highlighting (1) the need for future meta-evaluation to (i) be across multiple datasets and (ii) evaluate metrics on different application scenarios, e.g. summary level vs. system level (2) the need for more systematic meta-evaluation of summarization metrics that updates with our ever-evolving systems and datasets, and (3) the potential benefit to the summarization community of a shared task similar to the WMT[3] Metrics Task in Machine Translation, where systems and metrics co-evolve.

[3]http://www.statmt.org/wmt20/

2 Preliminaries

In this section we describe the datasets, systems, metrics, and meta evaluation methods used below.

2.1 Datasets

TAC-2008, 2009 (Dang and Owczarzak, 2008, 2009) are multi-document, multi-reference summarization datasets. Human judgments are available on for the system summaries submitted during the TAC-2008, TAC-2009 shared tasks.

CNN/DailyMail (CNNDM) (Hermann et al., 2015; Nallapati et al., 2016) is a commonly used summarization dataset that contains news articles and associated highlights as summaries. We use the version without entities anonymized.

2.2 Representative Systems

We use the following representative top-scoring systems that either achieve state-of-the-art (SOTA) results or competitive performance, for which we could gather the outputs on the CNNDM dataset.

Extractive summarization systems. We use CNN-LSTM-BiClassifier (CLSTM-SL; Kedzie et al. (2018)), Latent (Zhang et al., 2018), BanditSum (Dong et al., 2018), REFRESH (Narayan et al., 2018), NeuSum (Zhou et al., 2018), HIBERT (Zhang et al., 2019b), Bert-Sum-Ext (Liu and Lapata, 2019a), CNN-Transformer-BiClassifier (CTrans-SL; Zhong et al. (2019)), CNN-Transformer-Pointer (CTrans-PN; Zhong et al. (2019)), HeterGraph (Wang et al., 2020) and MatchSum (Zhong et al., 2020) as representatives of extractive systems, totaling 11 extractive system outputs for each document in the CNNDM test set.

Abstractive summarization systems. We use pointer-generator+coverage (See et al., 2017), fastAbsRL (Chen and Bansal, 2018), fastAbsRL-rank (Chen and Bansal, 2018), Bottom-up (Gehrmann et al., 2018), T5 (Raffel et al., 2019),

Unilm-v1 (Dong et al., 2019), Unilm-v2 (Dong et al., 2019), twoStageRL (Zhang et al., 2019a), preSummAbs (Liu and Lapata, 2019b), preSummAbs-ext (Liu and Lapata, 2019b) BART (Lewis et al., 2019) and Semsim (Yoon et al., 2020) as abstractive systems. In total, we use 14 abstractive system outputs for each document in the CNNDM test set.

2.3 Evaluation Metrics

We examine eight metrics that measure the agreement between two texts, in our case, between the system summary and reference summary.

BERTScore (BScore) measures soft overlap between contextual BERT embeddings of tokens between the two texts[4] (Zhang et al., 2020).

MoverScore (MScore) applies a distance measure to contextualized BERT and ELMo word embeddings[5] (Zhao et al., 2019).

Sentence Mover Similarity (SMS) applies minimum distance matching between text based on sentence embeddings (Clark et al., 2019).

Word Mover Similarity (WMS) measures similarity using minimum distance matching between texts which are represented as a bag of word embeddings[6] (Kusner et al., 2015).

JS divergence (JS-2) measures Jensen-Shannon divergence between the two text's bigram distributions[7] (Lin et al., 2006).

ROUGE-1 and ROUGE-2 measure overlap of unigrams and bigrams respectively[8] (Lin, 2004).

ROUGE-L measures overlap of the longest common subsequence between two texts (Lin, 2004).

We use the recall variant of all metrics (since the Pyramid method of human evaluations is inherently recall based) except MScore which has no specific recall variant.

2.4 Correlation Measures

***Pearson* Correlation** is a measure of linear correlation between two variables and is popular in meta-evaluating metrics at the system level (Lee Rodgers, 1988). We use the implementation given by Virtanen et al. (2020).

William's Significance Test is a means of calculating the statistical significance of differences in correlations for dependent variables (Williams, 1959; Graham and Baldwin, 2014). This is useful for us since metrics evaluated on the same dataset are not independent of each other.

2.5 Meta Evaluation Strategies

There are two broad meta-evaluation strategies: summary-level and system-level.

Setup: For each document $d_i, i \in \{1 \dots n\}$ in a dataset $\mathcal{D}$, we have J system outputs, where the outputs can come from (1) extractive systems (Ext), (2) abstractive systems (Abs) or (3) a union of both (Mix). Let $s_{ij}, j \in \{1 \dots J\}$ be the j^{th} summary of the i^{th} document, m_i be a specific metric and K be a correlation measure.

2.5.1 Summary Level

Summary-level correlation is calculated as follows:

$$K^{sum}_{m_1 m_2} = \frac{1}{n}\sum_{i=1}^{n}\Big(K\big([m_1(s_{i1}) \dots m_1(s_{iJ})], [m_2(s_{i1}) \dots m_2(s_{iJ})]\big)\Big). \quad (1)$$

Here, correlation is calculated for each document, among the different system outputs of that document, and the mean value is reported.

2.5.2 System Level

System-level correlation is calculated as follows:

$$K^{sys}_{m_1 m_2} = K\Big(\Big[\frac{1}{n}\sum_{i=1}^{n} m_1(s_{i1}) \dots \frac{1}{n}\sum_{i=1}^{n} m_1(s_{iJ})\Big], \Big[\frac{1}{n}\sum_{i=1}^{n} m_2(s_{i1}) \dots \frac{1}{n}\sum_{i=1}^{n} m_2(s_{iJ})\Big]\Big). \quad (2)$$

Additionally, the "quality" of a system sys_j is defined as the mean human score received by it i.e.

$$\text{HScore}^{sys_j}_{mean} = \frac{1}{n}\sum_{i=1}^{n}\text{humanScore}(s_{ij}). \quad (3)$$

3 Collection of Human Judgments

We follow a 3-step process to collect human judgments: (1) we *collect* system-generated summaries on the most-commonly used summarization dataset, CNNDM; (2) we *select* representative test samples from CNNDM and (3) we manually *evaluate* system-generated summaries of the above-selected test samples.

[4] Used code at github.com/Tiiiger/bert_score

[5] Used code at github.com/AIPHES/emnlp19-moverscore

[6] For WMS and SMS: github.com/eaclark07/sms

[7] JS-2 is calculated using the function defined in github.com/UKPLab/coling2016-genetic-swarm-MDS

[8] For ROUGE-1,2, and L, we used the python wrapper: https://github.com/sebastianGehrmann/rouge-baselines

3.1 System-Generated Summary Collection

We collect the system-generated summaries from 25 top-scoring systems,[9] covering 11 extractive and 14 abstractive systems (Sec. 2.2) on the CNNDM dataset. We organize our collected generated summaries into three groups based on system type:

- CNNDM Abs denotes collected output summaries from abstractive systems.
- CNNDM Ext denotes collected output summaries from extractive systems.
- CNNDM Mix is the union of the two.

3.2 Representative Sample Selection

Since collecting human annotations is costly, we sample 100 documents from CNNDM test set (11,490 samples) and evaluate system generated summaries of these 100 documents. We aim to include documents of varying difficulties in the representative sample. As a proxy to the difficulty of summarizing a document, we use the mean score received by the system generated summaries for the document. Based on this, we partition the CNNDM test set into 5 equal sized bins and sample 4 documents from each bin. We repeat this process for 5 metrics (BERTScore, MoverScore, R-1, R-2, R-L) obtaining a sample of 100 documents. This methodology is detailed in Alg. 1 in Sec. A.1.

3.3 Human Evaluation

In text summarization, a "good" summary should represent as much relevant content from the input document as possible, within the acceptable length limits. Many human evaluation methods have been proposed to capture this desideratum (Nenkova and Passonneau, 2004; Chaganty et al., 2018; Fan et al., 2018; Shapira et al., 2019). Among these, *Pyramid* (Nenkova and Passonneau, 2004) is a reliable and widely used method, that evaluates content selection by (1) exhaustively obtaining Semantic Content Units (SCUs) from reference summaries, (2) weighting them based on the number of times they are mentioned and (3) scoring a system summary based on which SCUs can be inferred.

Recently, Shapira et al. (2019) extended Pyramid to a lightweight, crowdsourcable method - LitePyramids, which uses Amazon Mechanical Turk[10] (AMT) for gathering human annotations. LitePyramids simplifies Pyramid by (1) allowing crowd workers to extract a subset of all possible SCUs and (2) eliminating the difficult task of merging duplicate SCUs from different reference summaries, instead using SCU sampling to simulate frequency-based weighting.

Both Pyramid and LitePyramid rely on the presence of multiple references per document to assign importance weights to SCUs. However in the CNNDM dataset there is only one reference summary per document. We therefore adapt the LitePyramid method for the single-reference setting as follows.

SCU Extraction The LitePyramids annotation instructions define a Semantic Content Unit (SCU) as a *sentence containing a single fact written as briefly and clearly as possible*. Instead, we focus on shorter, more fine-grained SCUs that contain at most 2-3 entities. This allows for partial content overlap between a generated and reference summary, and also makes the task easy for workers. Tab. 2 gives an example. We exhaustively extract (up to 16) SCUs[11] from each reference summary. Requiring the set of SCUs to be exhaustive increases the complexity of the SCU generation task, and hence instead of relying on crowd-workers, we create SCUs from reference summaries ourselves. In the end, we obtained nearly 10.5 SCUs on average from each reference summary.

System Evaluation During system evaluation the full set of SCUs is presented to crowd workers. Workers are paid similar to Shapira et al. (2019), scaling the rates for fewer SCUs and shorter summary texts. For abstractive systems, we pay \$0.20 per summary and for extractive systems, we pay \$0.15 per summary since extractive summaries are more readable and might precisely overlap with SCUs. We post-process system output summaries before presenting them to annotators by true-casing the text using Stanford CoreNLP (Manning et al., 2014) and replacing "unknown" tokens with a special symbol "□" (Chaganty et al., 2018).

Tab. 2 depicts an example reference summary, system summary, SCUs extracted from the reference summary, and annotations obtained in evaluating the system summary.

Annotation Scoring For robustness (Shapira et al., 2019), each system summary is evaluated by 4 crowd workers. Each worker annotates up to 16 SCUs by marking an SCU "present" if it can be

[9] We contacted the authors of these systems to gather the corresponding outputs, including variants of the systems.

[10] https://www.mturk.com/

[11] In our representative sample we found no document having more than 16 SCUs.

(a) Reference Summary: Bayern Munich beat Porto 6 - 1 in the Champions League on Tuesday. Pep Guardiola's side progressed 7 - 4 on aggregate to reach semi-finals. Thomas Muller scored 27th Champions League goal to pass Mario Gomez. Muller is now the leading German scorer in the competition. After game Muller led the celebrations with supporters using a megaphone.

(b) System Summary (BART, Lewis et al. (2019)): Bayern Munich beat Porto 6 - 1 at the Allianz Arena on Tuesday night. Thomas Muller scored his 27th Champions League goal. The 25 - year - old became the highest - scoring German since the tournament took its current shape in 1992. Bayern players remained on the pitch for some time as they celebrated with supporters.

(c) SCUs with corresponding evaluations:

- Bayern Munich beat Porto. ✓
- Bayern Munich won 6 - 1. ✓
- Bayern Munich won in Champions League. ✓
- Bayern Munich won on Tuesday. ✓
- Bayern Munich is managed by Pep Guardiola. ×
- Bayern Munich progressed in the competition. ✓
- Bayern Munich reached semi-finals. ×
- Bayern Munich progressed 7 - 4 on aggregate. ×
- Thomas Muller scored 27th Champions League goal. ✓
- Thomas Muller passed Mario Gomez in goals. ×
- Thomas Muller is now the leading German scorer in the competition. ✓
- After the game Thomas Muller led the celebrations. ×
- Thomas Muller led the celebrations using a megaphone. ×

Table 2: Example of a summary and corresponding annotation. (a) shows a reference summary from the representative sample of the CNNDM test set. (b) shows the corresponding system summary generated by BART, one of the abstractive systems used in the study. (c) shows the SCUs (Semantic Content Units) extracted from (a) and the "*Present*(✓)"/"*Not Present*(×)" marked by crowd workers when evaluating (b).

inferred from the system summary or "not present" otherwise. We obtain a total of 10,000 human annotations (100 documents × 25 systems × 4 workers). For each document, we identify a "noisy" worker as one who disagrees with the majority (i.e. marks an SCU as "present" when majority thinks "not present" or vice-versa), on the largest number of SCUs. We remove the annotations of noisy workers and retain 7,742 annotations of the 10,000. After this filtering, we obtain an average inter-annotator agreement (Krippendorff's alpha (Krippendorff, 2011)) of 0.66.[12] Finally, we use the majority vote to mark the presence of an SCU in a system summary, breaking ties by the class, "not present".

4 Experiments

Motivated by the central research question: "does the rapid progress of model development in summarization models require us to *re-evaluate* the evaluation process used for text summarization?" We use the collected human judgments to meta-evaluate current metrics from four diverse viewpoints, measuring the ability of metrics to: (1) evaluate *all* systems; (2) evaluate top-k strongest systems; (3) compare *two* systems; (4) evaluate individual summaries. We find that many previously attested properties of metrics observed on TAC exhibit different trends on the new CNNDM dataset.

[12]The agreement was 0.57 and 0.72 for extractive and abstractive systems respectively.

4.1 Exp-I: Evaluating *All* Systems

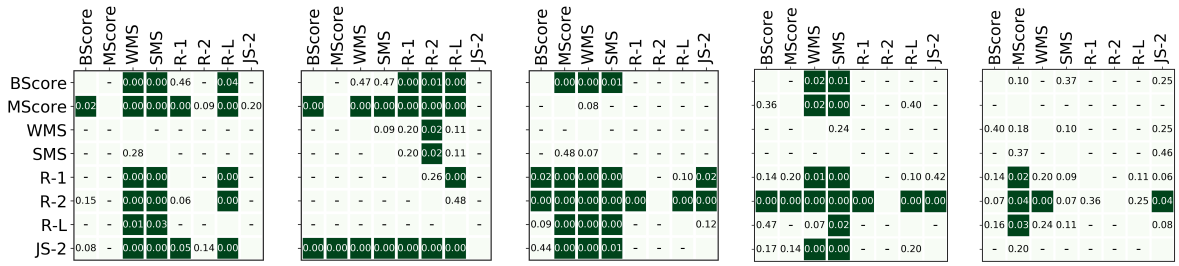
Automatic metrics are widely used to determine where a new system may rank against existing state-of-the-art systems. Thus, in meta-evaluation studies, calculating correlation of automatic metrics with human judgments at the system level is a commonly-used setting (Novikova et al., 2017; Bojar et al., 2016; Graham, 2015). We follow this setting and specifically, ask two questions: **Can metrics reliably compare different systems?** To answer this we observe the *Pearson* correlation between different metrics and human judgments in Fig. 2, finding that:

(1) MoverScore and JS-2, which were the best performing metrics on TAC, have poor correlations with humans in comparing CNNDM Ext systems.

(2) Most metrics have high correlations on the TAC-2008 dataset but many suffer on TAC-2009, especially ROUGE based metrics. However, ROUGE metrics consistently perform well on the collected CNNDM datasets.

Are some metrics significantly better than others in comparing systems? Since automated metrics calculated on the same data are not independent, we must perform the William's test (Williams, 1959) to establish if the difference in correlations between metrics is statistically significant (Graham and Baldwin, 2014). In Fig. 1 we report the p-values of William's test. We find that

[13]Dark cells with p-value = 0.05 have been rounded up.



(a) TAC-2008 (b) TAC-2009 (c) CNNDM Mix (d) CNNDM Abs (e) CNNDM Ext

Figure 1: p-value of William's Significance Test for the hypothesis "Is the system on left (y-axis) significantly better than system on top (x-axis)". 'BScore' refers to BERTScore and 'MScore' refers to MoverScore. A dark green value in cell (i, j) denotes metric m_i has a significantly higher *Pearson* correlation with human scores compared to metric m_j (p-value < 0.05).[13] '-' in cell (i, j) refers to the case when *Pearson* correlation of m_i with human scores is less that of m_j (Sec. 4.1).

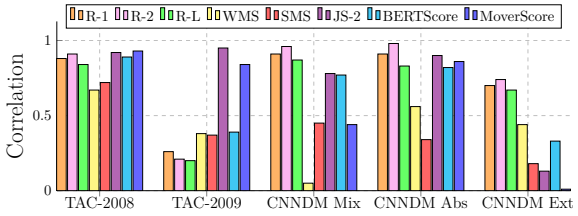


Figure 2: System-level *Pearson* correlation between metrics and human scores (Sec. 4.1).

(1) MoverScore and JS-2 are significantly better than other metrics in correlating with human judgments on the TAC datasets.
(2) However, on CNNDM Abs and CNNDM Mix, R-2 significantly outperforms all others whereas on CNNDM Ext none of the metrics show significant improvements over others.
Takeaway: These results suggest that metrics run the risk of overfitting to some datasets, highlighting the need to meta-evaluate metrics for modern datasets and systems. Additionally, there is no one-size-fits-all metric that can outperform others on all datasets. This suggests the utility of using different metrics for different datasets to evaluate systems e.g. MoverScore on TAC-2008, JS-2 on TAC-2009 and R-2 on CNNDM datasets.

4.2 Exp-II: Evaluating *Top-k* Systems

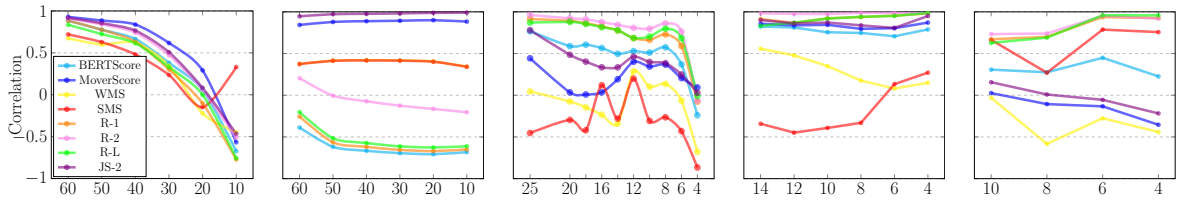
Most papers that propose a new state-of-the-art system often use automatic metrics as a proxy to human judgments to compare their proposed method against other top scoring systems. However, *can metrics reliably quantify the improvements that one high quality system makes over other competitive systems?* To answer this, instead of focusing on all of the collected systems, we evaluate the correlation between automatic metrics and human judgments in comparing the *top-k* systems, where *top-k* are chosen based on a system's mean human score (Eqn. 3).[14] Our observations are presented in Fig. 3. We find that:
(1) As k becomes smaller, metrics de-correlate with humans on the TAC-2008 and CNNDM Mix datasets, even getting negative correlations for small values of k (Fig. 8a, 8c). Interestingly, SMS, R-1, R-2 and R-L *improve* in performance as k becomes smaller on CNNDM Ext.
(2) R-2 had negative correlations with human judgments on TAC-2009 for $k < 50$, however it remains highly correlated with human judgments on CNNDM Abs for all values of k.
Takeaway: Metrics cannot reliably quantify the improvements made by one system over others, especially for the top few systems across all datasets. Some metrics, however, are well suited for specific datasets, e.g. JS-2 and R-2 are reliable indicators of improvements on TAC-2009 and CNNDM Abs respectively.

4.3 Exp-III: Comparing Two-Systems

Instead of comparing many systems (Sec. 4.1, 4.2) ranking *two* systems aims to test the discriminative power of a metric, i.e., the degree to which the metric can capture statistically significant differences between two summarization systems.

We analyze the reliability of metrics along a useful dimension: *can metrics reliably say if one system is significantly better than another*? Since we only have 100 annotated summaries to compare any two systems, sys_1 and sys_2, we use paired bootstrap resampling, to test with statistical sig-

[14] As a caveat, we *do not* perform significance testing for this experiment, due to the small number of data points.



(a) TAC-2008 (b) TAC-2009 (c) CNNDM Mix (d) CNNDM Abs (e) CNNDM Ext

Figure 3: System-level *Pearson* correlation with humans on top-k systems (Sec. 4.2).

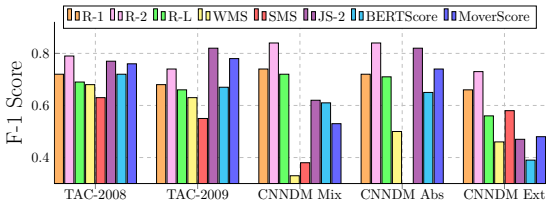
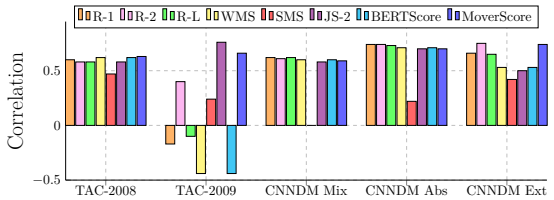
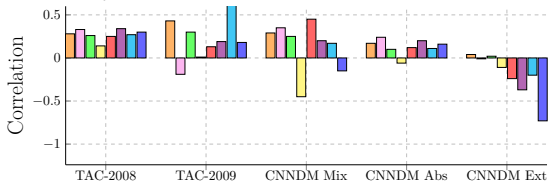


Figure 4: F1-Scores with bootstrapping (Sec. 4.3).



(a) Summary-level *Pearson* correlation with human scores.



(b) Difference between system-level and summary-level *Pearson* correlation.

Figure 5: *Pearson* correlation between metrics and human judgments across different datasets (Sec. 4.4).

nificance if sys_1 is better than sys_2 according to metric m (Koehn, 2004; Dror et al., 2018). We take all $\binom{J}{2}$ pairs of systems and compare their mean human score (Eqn. 3) using paired bootstrap resampling. We assign a label $y_{true} = 1$ if sys_1 is better than sys_2 with 95% confidence, $y_{true} = 2$ for vice-versa and $y_{true} = 0$ if the confidence is below 95%. We treat this as the ground truth label of the pair (sys_1, sys_2). This process is then repeated for all metrics, to get a "prediction", y^m_{pred} from each metric m for the same $\binom{J}{2}$ pairs. If m is a good proxy for human judgments, the F1 score (Goutte and Gaussier, 2005) between y^m_{pred} and y_{true} should be high. We calculate the weighted macro F1 score for all metrics and view them in Fig. 4.

We find that ROUGE based metrics perform moderately well in this task. R-2 performs the best on `CNNDM` datasets. While on the `TAC 2009` dataset, JS-2 achieves the highest F1 score, its performance is low on `CNNDM Ext`.

Takeaway: Different metrics are better suited for different datasets. For example, on the `CNNDM` datasets, we recommend using R-2 while, on the TAC datasets, we recommend using JS-2.

4.4 Exp-IV: Evaluating Summaries

In addition to comparing systems, real-world application scenarios also require metrics to reliably compare multiple summaries of a document. For example, top-scoring reinforcement learning based summarization systems (Böhm et al., 2019) and the current state-of-the-art extractive system (Zhong et al., 2020) heavily rely on summary-level reward scores to guide the optimization process.

In this experiment, we ask the question: *how well do different metrics perform at the summary level, i.e. in comparing system summaries generated from the same document?* We use Eq. 1 to calculate *Pearson* correlation between different metrics and human judgments for different datasets and collected system outputs. Our observations are summarized in Fig. 5. We find that:

(1) As compared to semantic matching metrics, R-1, R-2 and R-L have lower correlations on the `TAC` datasets but are strong indicators of good summaries especially for extractive summaries on the `CNNDM` dataset.

(2) Notably, BERTScore, WMS, R-1 and R-L have *negative* correlations on `TAC-2009` but perform moderately well on other datasets including `CNNDM`.

(3) Previous meta-evaluation studies (Novikova et al., 2017; Peyrard et al., 2017; Chaganty et al., 2018) conclude that automatic metrics tend to correlate well with humans at the system level but have poor correlations at the instance (here summary) level. We find this observation only holds on

TAC-2008. Some metrics' summary-level correlations can outperform system-level on the CNNDM dataset as shown in Fig. 7b (bins below $y = 0$). Notably, MoverScore has a correlation of only 0.05 on CNNDM Ext at the system level but 0.74 at the summary level.

Takeaway: Meta-evaluations of metrics on the old TAC datasets show significantly different trends than meta-evaluation on modern systems and datasets. Even though some metrics might be good at comparing summaries, they may point in the wrong direction when comparing systems. Moreover, some metrics show poor generalization ability to different datasets (e.g. BERTScore on TAC-2009 vs other datasets). This highlights the need for empirically testing the efficacy of different automatic metrics in evaluating summaries on multiple datasets.

5 Related Work

This work is connected to the following threads of topics in text summarization.

Human Judgment Collection Despite many approaches to the acquisition of human judgment (Chaganty et al., 2018; Nenkova and Passonneau, 2004; Shapira et al., 2019; Fan et al., 2018), *Pyramid* (Nenkova and Passonneau, 2004) has been a mainstream method to meta-evaluate various automatic metrics. Specifically, Pyramid provides a robust technique for evaluating content selection by exhaustively obtaining a set of Semantic Content Units (SCUs) from a set of references, and then scoring system summaries on how many SCUs can be inferred from them. Recently, Shapira et al. (2019) proposed a lightweight and crowdsourceable version of the original Pyramid, and demonstrated it on the DUC 2005 (Dang, 2005) and 2006 (Dang, 2006) multi-document summarization datasets. In this paper, our human evaluation methodology is based on the Pyramid (Nenkova and Passonneau, 2004) and LitePyramids (Shapira et al., 2019) techniques. Chaganty et al. (2018) also obtain human evaluations on system summaries on the CNNDM dataset, but with a focus on language quality of summaries. In comparison, our work is focused on evaluating content selection. Our work also covers more systems than their study (11 extractive + 14 abstractive vs. 4 abstractive).

Meta-evaluation with Human Judgment The effectiveness of different automatic metrics - ROUGE-2 (Lin, 2004), ROUGE-L (Lin, 2004), ROUGE-WE (Ng and Abrecht, 2015), JS-2 (Louis and Nenkova, 2013) and S3 (Peyrard et al., 2017) is commonly evaluated based on their correlation with human judgments (e.g., on the TAC-2008 (Dang and Owczarzak, 2008) and TAC-2009 (Dang and Owczarzak, 2009) datasets). As an important supplementary technique to meta-evaluation, Graham (2015) advocate for the use of a significance test, William's test (Williams, 1959), to measure the improved correlations of a metric with human scores and show that the popular variant of ROUGE (mean ROUGE-2 score) is sub-optimal. Unlike these works, instead of proposing a new metric, in this paper, we upgrade the meta-evaluation environment by introducing a sizeable human judgment dataset evaluating current top-scoring systems and mainstream datasets. And then, we re-evaluate diverse metrics at both system-level and summary-level settings. (Novikova et al., 2017) also analyzes existing metrics, but they only focus on dialog generation.

6 Implications and Future Directions

Our work not only diagnoses the limitations of current metrics but also highlights the importance of upgrading the existing meta-evaluation testbed, keeping it up-to-date with the rapid development of systems and datasets. In closing, we highlight some potential future directions: (1) The choice of metrics depends not only on different tasks (e.g, summarization, translation) but also on different datasets (e.g., TAC, CNNDM) and application scenarios (e.g, system-level, summary-level). Future works on meta-evaluation should investigate the effect of these settings on the performance of metrics. (2) Metrics easily overfit on limited datasets. Multi-dataset meta-evaluation can help us better understand each metric's peculiarity, therefore achieving a better choice of metrics under diverse scenarios. (3) Our collected human judgments can be used as supervision to instantiate the most recently-proposed *pretrain-then-finetune* framework (originally for machine translation) (Sellam et al., 2020), learning a robust metric for text summarization.

Acknowledgements

We sincerely thank all authors of the systems that we used in this work for sharing their systems' outputs.

References

Ondřej Bojar, Yvette Graham, Amir Kamran, and Miloš Stanojević. 2016. Results of the WMT16 metrics shared task. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 199–231, Berlin, Germany. Association for Computational Linguistics.

Florian Böhm, Yang Gao, Christian M. Meyer, Ori Shapira, Ido Dagan, and Iryna Gurevych. 2019. Better rewards yield better summaries: Learning to summarise without references.

Arun Tejasvi Chaganty, Stephen Mussman, and Percy Liang. 2018. The price of debiasing automatic metrics in natural language evaluation.

Yen-Chun Chen and Mohit Bansal. 2018. Fast abstractive summarization with reinforce-selected sentence rewriting. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 675–686, Melbourne, Australia. Association for Computational Linguistics.

Elizabeth Clark, Asli Celikyilmaz, and Noah A Smith. 2019. Sentence mover's similarity: Automatic evaluation for multi-sentence texts. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2748–2760.

Hoa Dang and Karolina Owczarzak. 2008. Overview of the tac 2008 update summarization task. In *Proceedings of the First Text Analysis Conference (TAC 2008)*, pages 1 – 16.

Hoa Dang and Karolina Owczarzak. 2009. Overview of the tac 2009 summarization track. In *Proceedings of the First Text Analysis Conference (TAC 2009)*, pages 1 – 16.

Hoa Trang Dang. 2005. Overview of duc 2005. In *In Proceedings of the Document Understanding Conf. Wksp. 2005 (DUC 2005) at the Human Language Technology Conf./Conf. on Empirical Methods in Natural Language Processing (HLT/EMNLP*.

Hoa Trang Dang. 2006. Overview of duc 2006. In *In Proceedings of HLT-NAACL 2006*.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. In *Advances in Neural Information Processing Systems*, pages 13042–13054.

Yue Dong, Yikang Shen, Eric Crawford, Herke van Hoof, and Jackie Chi Kit Cheung. 2018. BanditSum: Extractive summarization as a contextual bandit. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3739–3748, Brussels, Belgium. Association for Computational Linguistics.

Rotem Dror, Gili Baumer, Segev Shlomov, and Roi Reichart. 2018. The hitchhiker's guide to testing statistical significance in natural language processing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1383–1392, Melbourne, Australia. Association for Computational Linguistics.

Angela Fan, David Grangier, and Michael Auli. 2018. Controllable abstractive summarization. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 45–54, Melbourne, Australia. Association for Computational Linguistics.

Sebastian Gehrmann, Yuntian Deng, and Alexander Rush. 2018. Bottom-up abstractive summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4098–4109.

Cyril Goutte and Eric Gaussier. 2005. A probabilistic interpretation of precision, recall and f-score, with implication for evaluation. In *European Conference on Information Retrieval*, pages 345–359. Springer.

Yvette Graham. 2015. Re-evaluating automatic summarization with BLEU and 192 shades of ROUGE. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 128–137, Lisbon, Portugal. Association for Computational Linguistics.

Yvette Graham and Timothy Baldwin. 2014. Testing for significance of increased correlation with human judgment. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 172–176, Doha, Qatar. Association for Computational Linguistics.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, pages 1684–1692.

Chris Kedzie, Kathleen McKeown, and Hal Daume III. 2018. Content selection in deep learning models of summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1818–1828.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 388–395, Barcelona, Spain. Association for Computational Linguistics.

Klaus Krippendorff. 2011. Computing krippendorff's alpha-reliability.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. 2015. From word embeddings to document distances. In *International conference on machine learning*, pages 957–966.

W Alan Lee Rodgers. 1988. Thirteen ways to look at the correlation coefficient. *The American Statistician*, 42(1):59–66.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out*.

Chin-Yew Lin, Guihong Cao, Jianfeng Gao, and Jian-Yun Nie. 2006. An information-theoretic approach to automatic evaluation of summaries. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 463–470, New York City, USA. Association for Computational Linguistics.

Chin-Yew Lin and Franz Josef Och. 2004. ORANGE: a method for evaluating automatic evaluation metrics for machine translation. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 501–507, Geneva, Switzerland. COLING.

Yang Liu and Mirella Lapata. 2019a. Text summarization with pretrained encoders. 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 3730–3740, Hong Kong, China. Association for Computational Linguistics.

Yang Liu and Mirella Lapata. 2019b. Text summarization with pretrained encoders.

Annie Louis and Ani Nenkova. 2013. Automatically assessing machine summary content without a gold standard. *Computational Linguistics*, 39(2):267–300.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Ça glar Gulçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. *CoNLL 2016*, page 280.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Ranking sentences for extractive summarization with reinforcement learning. 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 1747–1759, New Orleans, Louisiana. Association for Computational Linguistics.

Ani Nenkova and Rebecca Passonneau. 2004. Evaluating content selection in summarization: The pyramid method. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 145–152, Boston, Massachusetts, USA. Association for Computational Linguistics.

Jun-Ping Ng and Viktoria Abrecht. 2015. Better summarization evaluation with word embeddings for ROUGE. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1925–1930, Lisbon, Portugal. Association for Computational Linguistics.

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, Copenhagen, Denmark. Association for Computational Linguistics.

Maxime Peyrard. 2019. Studying summarization evaluation metrics in the appropriate scoring range. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5093–5100, Florence, Italy. Association for Computational Linguistics.

Maxime Peyrard, Teresa Botschen, and Iryna Gurevych. 2017. Learning to score system summaries for better content selection evaluation. In *Proceedings of the Workshop on New Frontiers in Summarization*, pages 74–84, Copenhagen, Denmark. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer.

Peter A. Rankel, John M. Conroy, Hoa Trang Dang, and Ani Nenkova. 2013. A decade of automatic content evaluation of news summaries: Reassessing the state of the art. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 131–136, Sofia, Bulgaria. Association for Computational Linguistics.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Thibault Sellam, Dipanjan Das, and Ankur P Parikh. 2020. Bleurt: Learning robust metrics for text generation. *arXiv preprint arXiv:2004.04696*.

Ori Shapira, David Gabay, Yang Gao, Hadar Ronen, Ramakanth Pasunuru, Mohit Bansal, Yael Amsterdamer, and Ido Dagan. 2019. Crowdsourcing lightweight pyramids for manual summary evaluation. 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 682–687, Minneapolis, Minnesota. Association for Computational Linguistics.

Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, CJ Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake Vand erPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1. 0 Contributors. 2020. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272.

Danqing Wang, Pengfei Liu, Yining Zheng, Xipeng Qiu, and Xuanjing Huang. 2020. Heterogeneous graph neural networks for extractive document summarization. *arXiv preprint arXiv:2004.12393*.

Evan J. Williams. 1959. Regression analysis. *Wiley, New York*, 14.

Wonjin Yoon, Yoon Sun Yeo, Minbyul Jeong, Bong-Jun Yi, and Jaewoo Kang. 2020. Learning by semantic similarity makes abstractive summarization better. *arXiv preprint arXiv:2002.07767*.

Haoyu Zhang, Yeyun Gong, Yu Yan, Nan Duan, Jianjun Xu, Ji Wang, Ming Gong, and Ming Zhou. 2019a. Pretraining-based natural language generation for text summarization. *arXiv preprint arXiv:1902.09243*.

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*.

Xingxing Zhang, Mirella Lapata, Furu Wei, and Ming Zhou. 2018. Neural latent extractive document summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 779–784, Brussels, Belgium. Association for Computational Linguistics.

Xingxing Zhang, Furu Wei, and Ming Zhou. 2019b. HIBERT: Document level pre-training of hierarchical bidirectional transformers for document summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5059–5069, Florence, Italy. Association for Computational Linguistics.

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, Hong Kong, China. Association for Computational Linguistics.

Ming Zhong, Pengfei Liu, Yiran Chen, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. 2020. Extractive summarization as text matching. *arXiv preprint arXiv:2004.08795*.

Ming Zhong, Pengfei Liu, Danqing Wang, Xipeng Qiu, and Xuan-Jing Huang. 2019. Searching for effective neural extractive summarization: What works and what's next. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*, pages 1049–1058.

Deyu Zhou, Linsen Guo, and Yulan He. 2018. Neural storyline extraction model for storyline generation from news articles. 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 1727–1736, New Orleans, Louisiana. Association for Computational Linguistics.

A Appendices

A.1 Sampling Methodology

Please see Algorithm 1.

Algorithm 1: Sampling Methodology

Data: $(d_i, r_i, S_i) \in D$ where D is CNNDM test set, d_i is source document, r_i is reference summary, and S_i is a set of individual system summaries $s_{ij} \in S_i$. M = [ROUGE-1, ROUGE-2, ROUGE-L, BERTScore, MoverScore]

Output: D_{out}: sampled set of documents

$\mu_{m,i} := mean(\{m(r_i, s_{ij}) \forall s_{ij} \in S_i\}), m \in M$
$\sigma_{m,i} := std.dev(\{m(r_i, s_{ij}) \forall s_{ij} \in S_i\}), m \in M$
$D_{out} := \{\}$
$n_1 := |D|/5$
for $m \in M$ **do**
  $D' :=$ $[d_i : d_i \in D]$ sorted by $\mu_{m,i}$
  for $k \in [0, 1, 2, 3, 4]$ **do**
    $D'_k = D'[i * n_1 : (i + 1) * n_1]$
    $D''_k = [d_i : d_i \in D'_k]$ sorted by $\sigma_{m,i}$
    $n_2 = |D''_k|/4$
    for $l \in [0, 1, 2, 3]$ **do**
      $D''_{kl} = D''_k[l * n_2 : (l + 1) * n_2]$
      Randomly sample d_i from D''_{kl}
      $D_{out} = D_{out} \cup d_i$
    end
  end
end

A.2 Exp-I using Kendall's tau correlation

Please see Figure 6 for the system level Kendall's tau correlation between different metrics and human judgements.

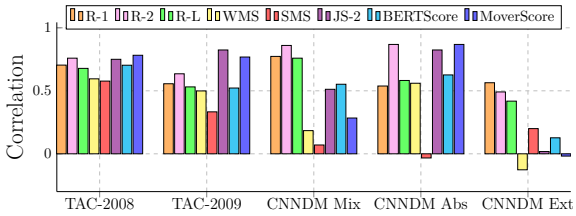


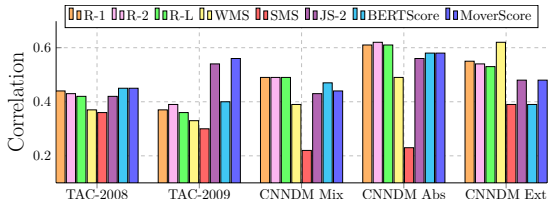
Figure 6: System-level *Kendall* correlation between metrics and human scores.

A.3 Exp-II using Kendall's tau correlation

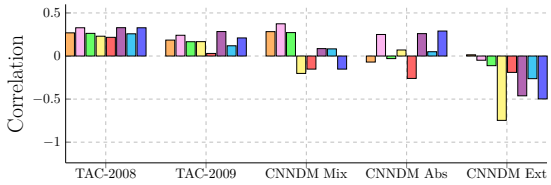
Please see Figure 8 for the system level Kendall's tau correlation on top-k systems, between different metrics and human judgements.

A.4 Exp IV using Kendall's tau correlation

Please see Figure 7 for the summary level Kendall's tau correlation between different metrics and human judgements.

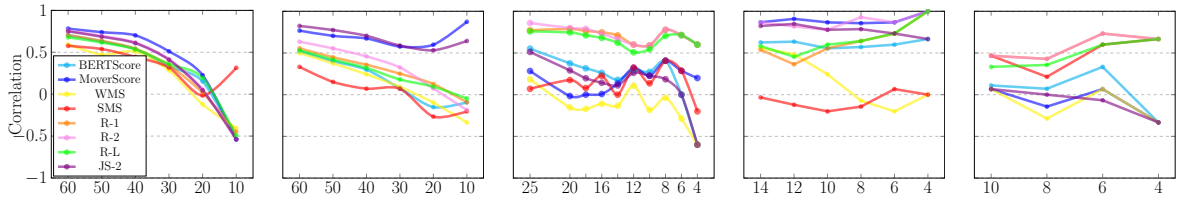


(a) Summary-level *Kendall* correlation with human scores.



(b) Difference between system-level and summary-level *Kendall* correlation.

Figure 7: *Kendall* correlation between metrics and human judgements across different datasets.



(a) TAC-2008 (b) TAC-2009 (c) CNNDM Mix (d) CNNDM Abs (e) CNNDM Ext

Figure 8: System-level *Kendall* correlation with humans on top-k systems.