Near-Negative Distinction: Giving a Second Life to Human Evaluation Datasets

Philippe Laban **Chien-Sheng Wu** **Wenhao Liu** **Caiming Xiong**
Salesforce AI Research
{plaban, wu.jason, wenhao.liu, cxiong}@salesforce.com

Abstract

Precisely assessing the progress in natural language generation (NLG) tasks is challenging, and human evaluation to establish a preference in a model's output over another is often necessary. However, human evaluation is usually costly, difficult to reproduce, and non-reusable. In this paper, we propose a new and simple automatic evaluation method for NLG called Near-Negative Distinction (NND) that repurposes prior human annotations into NND tests. In an NND test, an NLG model must place a higher likelihood on a high-quality output candidate than on a near-negative candidate with a known error. Model performance is established by the number of NND tests a model passes, as well as the distribution over task-specific errors the model fails on. Through experiments on three NLG tasks (question generation, question answering, and summarization), we show that NND achieves a higher correlation with human judgments than standard NLG evaluation metrics. We then illustrate NND evaluation in four practical scenarios, for example performing fine-grain model analysis, or studying model training dynamics. Our findings suggest that NND can give a second life to human annotations and provide low-cost NLG evaluation.

1 Introduction

Pre-training of large language models has fueled recent progress in many natural language generation (NLG) tasks such as summarization (Zhang et al., 2020), question answering (Tafjord and Clark, 2021) (Ng et al., 2019), and question generation (Murakhovs'ka et al., 2022). However, quantifying this progress remains a challenge due to the open-ended nature of NLG.

The gold standard for NLG evaluation is manual expert annotation: it can be highly precise and fully customized to an NLG task, helping identify model limitations, and setting the direction of future work. The main limitation of manual expert annotation is the complexity and cost associated with running an evaluation. The cost often increases linearly or quadratically with the number of models compared, restricting evaluation to a small number of models.

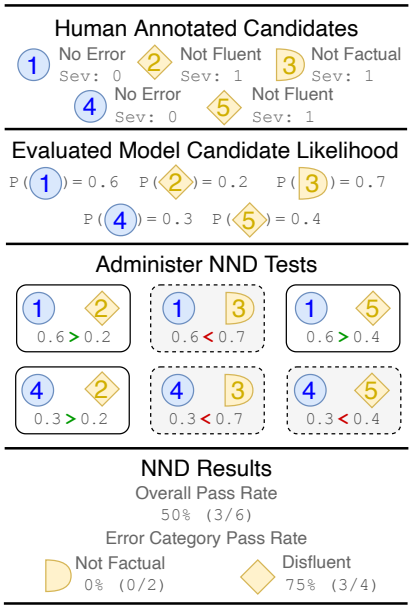


Figure 1: Stages of the **Near-Negative Distinction** framework for NLG evaluation. A pre-existing human evaluation is repurposed into a series of NND tests.

To circumvent the cost of expert evaluation, many in the field rely on automatic metrics such as BLEU (Papineni et al., 2002), which compute text quality based on n-gram overlap between model outputs and human references. Such metrics are easy to compute, and have been shown to moderately correlate with human judgments, but are limited in three ways: they only offer an aggregate score that is difficult to interpret, they do not offer a clear upper bound in performance, and they have limited generalized ability to some NLG tasks (Liu et al., 2016; Sulem et al., 2018a).

In this paper, we propose a new and simple automatic framework for the evaluation of NLG models which we call Near-Negative Distinction (NND). At a high level, the NND framework bridges the

gap between expert annotation and automated metrics by repurposing existing annotations into a series of automatic tests which assess how likely a model is to avoid previously annotated errors.

The first contribution of our work is the definition of the NND framework, illustrated in Figure 1. In NND, an existing human evaluation dataset D is repurposed to create a series of automated tests. For a given input context, D should contain annotations for several model outputs, some high-quality (candidates 1, 4) and some low-quality (candidates 2, 3, 5). A collection of NND tests is created containing candidate pairs of differing quality. Generation models pass an NND test if they assign a higher likelihood to the high-quality candidate than the low-quality one. NND evaluation produces an overall test pass rate, as well as a pass rate for each error category, which can be used to inspect model strengths and weaknesses.

The second contribution is the creation of NND datasets from existing human evaluations for three NLG tasks: question generation, generative question answering, and summarization. On these three tasks, verification experiments find that NND pass rates correlate better with human judgments than existing evaluation metrics, both n-gram-based metrics such as BLEU (Papineni et al., 2002), and more recent metrics such as BERTScore (Zhang et al., 2019) and QuestEval (Zhang et al., 2019).

The third contribution is a collection of practical experiments showcasing how to use NND. The experiments demonstrate the flexibility of the NND framework, showing it can be useful to extrapolate a model's performance in a user study, perform fine-grain model analysis, study scaling effects in model families, or discover trends during training.

Although we focus experiments on the English language, the NND framework is not English-specific, and we encourage the community to experiment with NND evaluation, helping to expand it to new NLG domains and languages.

We publicly release the NND datasets we generated as well as the code needed to create new NND datasets, and models used in experiments[1].

2 Near-Negative Distinction

We now detail the process of transforming pre-existing human annotations into an NND dataset and show how to perform NND evaluation.

[1] https://github.com/Salesforce/nnd_evaluation

2.1 NND Dataset Creation Procedure

A human annotation dataset D consists of `(context, candidate)` tuples that have been annotated typically with one or more labels from a discrete error categorization. Several properties are required from human annotation datasets to be compatible with the NND framework. First, several candidates should be annotated for each context, so that pairs of candidates can be formed into unit NND tests. Second, it should be possible to map error categories to varying quality levels. For instance in Figure 1, candidate 1 labeled `No Error` is of higher quality than candidate 2 labeled `Not Fluent`. If these properties are present in a human annotation dataset, an NND dataset can be created in three steps:

1. **Group By Context:** Group all annotated candidates for a given context, typically each candidate originates from an NLG model.

2. **Assign Quality**: Assign a quality to each candidate within a group based on its annotation.

3. **Generate Candidate Pairs:** For a given context, construct all pairs of candidates of differing quality (C_{high}, C_{low}).

The difference in quality between some error categories might not be known (e.g., the difference between "Not Fluent" and "Not Factual" candidates in Figure 1), preventing the ability to fully rank candidates. Because of this limitation, NND focuses on pairwise comparisons rather than ranking, analyzing each pair of candidates for which a quality differential is known.

2.2 Administering NND

The finalized NND dataset consists of `(context,` C_{high}`,` C_{low}`)` triplets we call *NND tests*. Most text generators are language models, which assign a probability to a sequence of tokens. Sequence probability can be used during generation to rank partial candidates such as in beam search generation, however most often a generated sequence's likelihood is discarded once generation is completed.

In NND, we make use of sequence likelihoods to assess whether models are likely to reproduce the mistakes of previous models, or if they can correctly assign lower likelihood to low-quality candidates. Formally, each candidate C is tokenized into a sequence of tokens: $w_1, ...w_N$, and a candidate's likelihood is computed in the following way:

$$LL(C) = \frac{\sum_{i=1}^{N} log(P(w_i|\texttt{ct}, w_1, ..., w_{i-1}))}{N}, \quad (1)$$

where $P(w_i|...)$ is the probability assigned by the model to the i-th token of the candidate, and $\texttt{ct}$ is the input context. We use log likelihood instead of likelihood, a standard step to improve numerical stability. We further choose to normalize the likelihood by the sequence length (N) to counterbalance the effect of sequence length on likelihood. An NND test is performed by computing the likelihood of both candidates $LL(C_{high})$ and $LL(C_{low})$ and comparing both. The model passes the test if

$$LL(C_{high}) > LL(C_{low}). \quad (2)$$

In cases where the model fails the test, the error category of C_{low} is recorded, allowing to compute NND pass rates for each category of error.

By administering an entire dataset of tests, the NND produces two outputs: first an overall pass rate which is the percentage of NND tests passed by the model, and the breakdown of pass rates for each error category. The two outputs complement each other: the former can be used to compare models, and the second can be used to inspect model performance and discover model limitations.

2.3 Verification of NND Quality

To gain an understanding of the quality of NND estimates, we run verification experiments assessing the level of correlation between NND estimates of model performance and human reference annotations. We run identical verification experiments with a set of standard NLG metrics.

We design two verification experiments based on desired properties for an evaluation metric: (1) **Rank Correlation**, an evaluation metric should rank NLG models similarly to rankings based on human annotation, (2) **Gap Correlation**, a metric's estimate of gaps in performance between pairs of models should correlate positively with gaps measured through human annotation (i.e., if human annotation reveals a large gap in performance between two models, the evaluation metric should similarly estimate a large gap).

For Rank Correlation, given a set of NLG models and a metric, we compute the Kendall rank correlation coefficient (τ) (Kendall, 1938) between the models' ranking according to the metric, and the ranking based on human annotation. Higher τ signifies that an evaluation metric is more accurate at predicting the ordinal ranks of models.

For Gap Correlation, for each pair of NLG models, we compute the difference in performance according to the metric and according to human annotation. The gaps of all pairs of models are assembled into two vectors of size $\binom{n}{2}$, and we compute the Pearson correlation of the two vectors. If a metric achieving a high Gap Correlation is well calibrated and can predict gaps in performance between two models accurately.

In Section 3, we introduce NND datasets for three NLG tasks, based on pre-existing human annotations. In Section 4, we perform the verification experiments in the three domains and confirm that NND correlates better with human opinion than well-established NLG metrics. Section 5 introduces practical use-cases of NND evaluation.

3 NND Datasets

3.1 NND For Question Generation

We first describe NND experiments for the task of Question Generation, based on Quiz Design (QD) dataset (Laban et al., 2022). For each context in QD, seven answer-aware QGen models generated up to seven questions. Ten teachers designing educational quizzes annotated 3,164 questions with one of four error types: *No Error, Disfluent, Off Target, Wrong Context*.

We generate NND tests by pairing *No Error* questions with any question with an error, producing 2,686 NND pairs in total. Examples in Table A1.

We run NND experiments with the seven models used in the original QD study (GPT2-{distil,base,med} (Radford et al., 2019), BART-{base,large} (Lewis et al., 2020), ProphetNet, and MixQG-Large (Murakhovs'ka et al., 2022)), as well as three newer models that were not released when the QD annotation was run: MixQG-3B, and Macaw-{3B-11B} (Tafjord and Clark, 2021).

3.2 NND For Question Answering

In generative QA, a QA model receives a question and must generate a potentially abstractive answer. We create an NND dataset by re-purposing the Challenge 300 annotations (Tafjord and Clark, 2021). Challenge 300 is a suite of 300 questions intended to challenge QA models (e.g., Can you sit and stand at the same time?). For each question, QA models must generate a free-text answer,

and candidate answers from five large QA models (including GPT3) were annotated with a credit of either 0 (incorrect), 0.5 (partially correct), or 1 (correct). Each question in Challenge 300 is further tagged into 20 categories, which we consolidate into 5 groups: Common Sense, Comparison, Entity, Creativity, and Science. We create NND test pairs out of (correct, incorrect) answer pairs and obtain 829 NND test pairs which we further organize according to category groups. Example NND tests for each category in Table A2.

We run NND experiments with three families of publicly available generative QA models: T5 fine-tuned on Natural Questions (Roberts et al., 2020), UnifiedQA (Khashabi et al., 2020), and Macaw (Tafjord and Clark, 2021), which achieved the highest performance during annotation.

3.3 NND For Summarization

For summarization, we adapt two human annotation datasets to the NND framework: SummEval (Fabbri et al., 2021) and FRANK (Pagnoni et al., 2021). Example NND tests in Table A3.

SummEval consists of 100 documents each with 8 to 9 system-generated summaries annotated with 5-Point Likert scale ratings on four general attributes (Consistency, Coherence, Fluency, and Relevance). We treat each attribute independently, and normalize Likert scale annotations following the SummaC benchmark procedure (Laban et al., 2021c): for each attribute, a summary is of high quality if a majority of annotators gave the summary a score of 5, and is of low quality otherwise. The NND procedure yields 3,613 NND tests.

FRANK focuses annotation on the consistency attribute, offering more specialized error categories. The test portion of FRANK contains 350 news articles, each coupled with 4 or 5 summaries. Each summary has annotations that follow a hierarchical error categorization, breaking down consistency errors into four groups: *No Error, Semantic Frame, Discourse, and Verifiability* errors.[2] We treat *No Error* as high-quality, and any other error as low-quality, and generate 824 NND test pairs.

We run NND experiments with five summarization models in the SummEval evaluation (M9, M17, M20, M22, M23) and perform a fine-grain comparison of BART-large and PEGASUS (Zhang et al., 2020), two models that achieve very strong ROUGE performance on the CNN/DM dataset (Nallapati et al., 2016).

	QGen		Gen. QA		Summ.	
Metric	**Rank**	**Gap**	**Rank**	**Gap**	**Rank**	**Gap**
	τ	r	τ	r	τ	r
BLEU	0.65	0.35	0.40	0.63	0.45	0.74
R-1	0.64	0.31	0.40	0.57	0.55	0.84
R-2	0.65	0.34	0.27	0.56	0.55	**0.86**
R-L	0.65	0.41	0.40	0.57	0.55	0.85
METEOR	0.65	0.36	0.27	0.56	0.55	0.72
BERT	0.49	0.36	0.40	0.57	0.65	0.67
BARTScore	0.65	0.40	0.27	0.57	**0.75**	0.74
QuestEval	0.65	0.39	0.27	0.56	**0.75**	0.79
NND	**0.78**	**0.80**	**0.67**	**0.86**	0.70	**0.86**

Table 1: **Results for the Rank (τ) and Gap (r) Correlation experiments.** Experiments were performed for Question Generation, Generative QA and Summarization using scores from standard NLG evaluation metrics and NND. Each entry is the average of verification experiments run on the dataset.

4 NND Verification

We now present results from running the verification experiments of Section 2.3 on the three domains we study. In our analysis, we compare NND to standard n-gram based evaluation metrics: BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), METEOR (Banerjee and Lavie, 2005), as well as more recent Transformer-based metrics: BERTScore (Zhang et al., 2019), BARTScore (Yuan et al., 2021) and QuestEval (Scialom et al., 2021). For each verification experiment, we are limited to evaluating models present in the annotation datasets that have been open-sourced as the NND framework requires a running version of the model to compute candidate likelihoods.

For QGen, verification experiments used all seven models present in the annotations dataset, with separate verification experiments run on each of the three error types. For QA, verification experiments involved three of the four available models[3], and were run on each question category. For Summarization, verification experiments were run with five summarizers from SummEval (M9, M17, M20, M22, M23) with experiments run on each of the four summarization aspects. We do not run verification experiments on FRANK, as it contains fewer annotations of publicly released models.

Verification results summarized in Table 1. NND compares favorably across the board, achieving the

[2]We remove the "Other" category as it has few samples.

[3]The fourth model GPT3-DaVinci is not publicly released

highest correlation on five of the six assessments. Improvements in correlation are stronger on the QG and generative QA tasks than Summarization, on which ROUGE, BARTScore, and QuestEval achieve strong performance.

We note an important conceptual difference between NND and the metrics we compare to which are reference-based. Reference-based metrics score a generator by establishing a similarity between the model's candidate outputs and human-written references. In contrast, NND is reference-less and relies on human annotations of several model candidate outputs to evaluate models. We hypothesize that the use of near-negatives, and whether a model is likely to avoid them, provides a useful signal that leads to high-quality model evaluation.

We next turn to use the NND framework in practical situations and assume that NND pass rates provide quality estimates of model ranks and performance gaps between models.

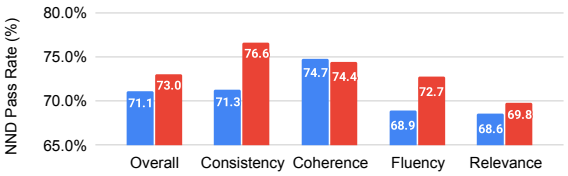
5 NND Applications

5.1 Extrapolating Model Performance

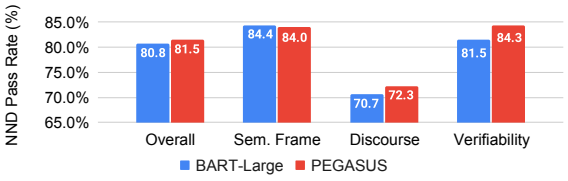
		Quiz Design NND			
		Overall	**Disfluent**	**Off Tgt**	**W. Ctxt**
#NND Tests		2686	711	890	1085
Model	**%A**	**NND Test Pass Rate (%)**			
Distil-GPT2	33.4	44.9	52.7	37.0	46.0
GPT2-base	40.9	52.3	60.3	49.7	49.3
GPT2-med	51.3	60.8	63.3	64.5	56.1
BART-Base	52.0	59.6	60.5	64.5	55.0
ProphetNet	53.5	67.7	58.1	79.8	64.1
BART-Large	58.4	64.2	63.3	70.8	59.4
MixQG-L	68.4	70.9	66.9	80.9	65.3
MixQG-3B	-	**72.9**	69.5	**81.7**	**67.8**
Macaw-3B	-	69.2	**70.3**	73.3	65.1
Macaw-11B	-	70.6	69.3	78.0	65.4

Table 2: **Extrapolation of QGen model's performance on the Quiz Design manual evaluation.** The first seven models (top) are part of the human evaluation (%A: original human acceptance rate), bottom three are only evaluated with NND.

In Quiz Design, the largest MixQG-3B model was not included in the annotations due to latency requirements for the interface (Laban et al., 2022). Further, new QGen models have been released since the study's conduct. We leverage NND's ability to provide category-specific estimates of performance to extrapolate how these unseen models would have performed in the Quiz Design Study.



(a) SummEval NND



(b) FRANK NND

Figure 2: **Fine-grain comparison of a pair of summarization models on based on two NND test sets.** SummEval estimates performance on four general aspects, and FRANK focuses on factual consistency errors.

We run NND experiments for each of the seven models included in the study, as well as the unseen models. Results are summarized in Table 2.

First, the three novel models all achieve strong performances, obtaining three of the best four overall NND pass rates. The MixQG-3B achieves the highest performance overall, seeing a total improvement of 2% when compared to MixQG-L, the best performer at the time of the study, with gains on all three error categories. The Macaw models achieve the strongest performance in *Disfluency*, but lower performance on *Off Target* and *Wrong Context* lead to lower performance overall.

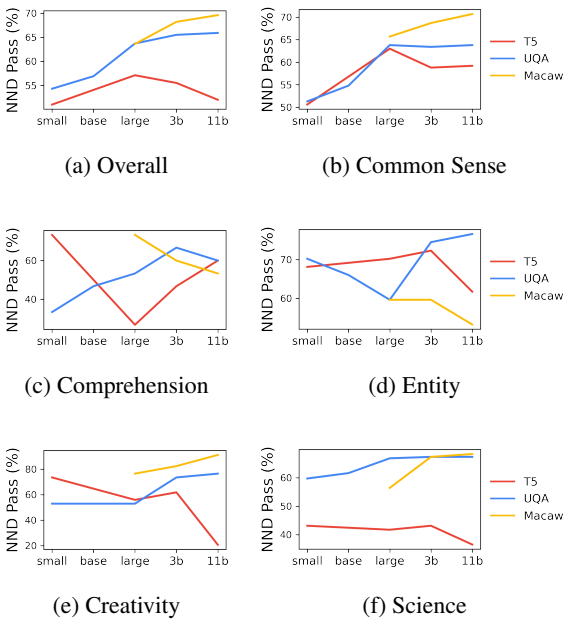
These results show that NND can be used to give a second life to human evaluation datasets by projecting model performance a posteriori.

5.2 Fine-Grained Model Comparison

Prior work has recognized the BART-Large and PEGASUS models as close contenders for top performance in summarization (Fabbri et al., 2021). The two models are virtually tied in terms of ROUGE-1 score on the CNN/DM test set with a variation of fewer than 0.1 points.

To gain specific insights into the differences between the models, we run NND experiments with both models using the general NND test set based on the SummEval annotations, as well as the factual consistency-focused FRANK annotations. Results are summarized in Figure 2.

On the SummEval test set, PEGASUS narrowly outperforms BART overall, owing to 4-5% gains in



(a) Overall (b) Common Sense

(c) Comprehension (d) Entity

(e) Creativity (f) Science

Figure 3: **Scaling experimental results for QA models.** Overall and category-specific NND pass rates are computed for varying model sizes from three model families: T5, UnifiedQA and Macaw.

the consistency and fluency aspects. Performance on the coherence and relevance aspects are narrower, with BART topping coherence, and PEGASUS with a slight edge in relevance.

The SummEval results are reaffirmed by the FRANK NND experiment, on which PEGASUS also outperforms BART overall, confirming that PEGASUS is better at avoiding factual errors than BART. However, on this more precise error categorization, PEGASUS does not win out entirely, with BART-Large achieving a higher pass rate on the Semantic Frame errors.

The NND results confirm that the two models' performance is close, with overall NND pass rates within 2% of each other, yet reveal some subtlety in the specific strengths and weaknesses of each model. Depending on the application, certain attributes might be of more or less importance, and NND could inform a user on which model to select.

5.3 Model Scaling Effects

The authors of the Challenge 300 dataset only annotated text outputs from the largest models available for each model family (Tafjord and Clark, 2021). This annotation strategy is understandable, as annotating smaller models' answers increases annotation cost, but it limits understanding of the effect of model size on performance.

We run NND experiments for all model sizes

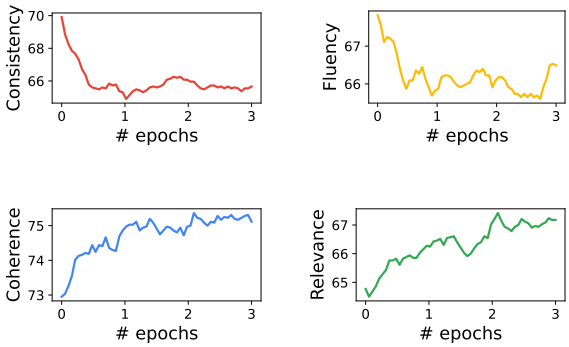


Figure 4: **NND Performance as a summarization model is trained.** By running SummEval NND evaluations on model checkpoints during training, model ability to detect consistency, fluency, coherence and relevance errors can be studied.

available for three families of QA models: T5 finetuned on Natural Questions (Small, Large, 3B, 11B) (Roberts et al., 2020), Unified-QA (Small, Base, Large, 3B, 11B) (Khashabi et al., 2020) and Macaw (Large, 3B, 11B) (Tafjord and Clark, 2021), with results summarized in Figure 3.

Overall, increasing model size leads to gradual increases in performance for the UnifiedQA and Macaw models. Unexpectedly for T5, performance peaks with the T5-Large, however overall the T5 family underperforms UnifiedQA and Macaw.

Focusing on UnifiedQA and Macaw, model performance increases steadily in three question categories: Common Sense, Creativity, and Science, but surprisingly stagnates or decreases in the Comprehension and Entity categories.

The NND experiments reveal that although performance tends to improve with model size increase, the trends vary widely by question category: an end-user with a particular question category in mind might benefit from a smaller model size.

5.4 Evaluation During Training

So far, we ran NND to evaluate finalized models, performing comparisons across models. We now use NND to inspect a model during training.

We train a BART-base model on the CNN/DM dataset using teacher forcing with cross-entropy loss for three epochs. We perform an NND evaluation of the latest model checkpoint every 2,000 training steps, using the SummEval NND test pairs.

Results summarized in Figure 4. Surprisingly, the model's ability to detect consistency and fluency errors decreases during training, with NND pass rates decreasing by 2-4%. This finding mirrors

the analysis of training dynamics in summarization, which finds that models become less factual in later stages of the training process (Goyal et al., 2021). On the other hand, model performance on coherence and relevance errors steadily increases during training. These trends could be explained by the model becoming better at summarization-specific skills, such as content selection (relevance) and ordering (coherence) at the cost of factual consistency and general fluency.

6 Related Work

NLG Benchmarks. Following the success of benchmarks such as GLUE (Wang et al., 2018) for the evaluation of NLU models, some work has proposed benchmarks as a way to evaluate NLG models, such as GLGE (Liu et al., 2021) with 8 NLG tasks or the crowd-sourced BigBench (bench collaboration, 2021) with 209 NLG tasks. More recently, the GEM Workshop proposed the GEM Benchmark (Gehrmann et al., 2021), a living benchmark with rule-based challenge sets which can be updated with new models and reference-based metrics. Benchmarks are useful for broad comparison of model performance across tasks, for example with the evaluation of large language models in few-shot settings. We view NND as complementary to NLG benchmarks: a highly task-specific tool that can be used to assess a model's potential limitation on a particular task.

LM Likelihood Score. Language-modeling likelihood and perplexity (the exponentiation of log-likelihood) are commonly used to evaluate NLG models (Hashimoto et al., 2019). For example, test-set perplexity is the standard metric to compare unconditional language models (Chelba et al., 2013; Khandelwal et al., 2019). Model capacity and vocabulary size affect likelihoods, and careful normalization is required for model-to-model comparisons (Jelinek et al., 1977). In NND, likelihoods are not compared across models, circumventing normalization needs. Furthermore, likelihood and perplexity lack interpretability, whereas NND mirrors error categories of human evaluations.

External LM Likelihood. Besides the evaluated model's own likelihood, some work has used an external language model's likelihood for scoring. BARTScore (Yuan et al., 2021) uses a BART model's likelihood to evaluate generated texts on faithfulness, precision, and recall factors. Salazar et al. (2020) propose Masked-Language Model Scoring to repurpose BERT-style NLU models into producing pseudo-log likelihoods shown to measure textual fluency. Although large external language models can be useful for measuring general language quality, it is challenging for a single model to assess the task-specific quality of generated text. In NND, test pairs are targeted at evaluating model performance on specific task skills.

Contrastive Learning. The use of negative candidates in NLG has been explored with recent interest in applying contrastive learning (Chopra et al., 2005) methods to NLG training (He and Glass, 2020; Liu and Liu, 2021; Cao and Wang, 2021). In contrastive learning, a model being trained receives both positive and negative candidates and has a two-sided objective of increasing the likelihood of positive candidates, while reducing the likelihood of negative candidates.

Similarly, **Self-Critical Sequence Training** (Rennie et al., 2017; Laban et al., 2021b) is an RL training method in which models generate several candidates which are scored and contrasted. NND relies on pairs of candidates of differing quality as well, however, the framework is focused on evaluation and not training. Further, SCST relies on automatic metrics to score negative candidates, whereas NND is based on human annotations. When a large number of NND tests are available, NND could be compatible with contrastive learning: a portion of the tests can be for model training, while a portion is reserved for evaluation.

Language Model Behavioral Analysis. Recent work has built behavioral analysis corpora (Isabelle et al., 2017; Naik et al., 2018; Vig et al., 2020) to evaluate model behavior and bias. For example, in: "The nurse said that _ is fine", a biased model assigns a higher likelihood to a stereotypical "she" pronoun than an anti-stereotypical pronoun ("he", "it"). Behavioral analysis corpora rely on unit tests, and models are evaluated by the percentage of passed tests. Unlike NND, behavioral analysis often relies on rules or a lexicon to construct tests and is focuses on the effect of a single word or phrase, whereas NND relies on model-generated candidates with human annotations.

Datasets Repurposing is common in machine learning and NLP (Koch et al., 2021; Koesten et al., 2020), particularly in cases where data access is limited or noisy. Common datasets, such as the Penn Treebank for syntax parsing (Marcinkiewicz, 1994), CNN/DM for summarization(Nallapati et al.,

2016), or PPDB for paraphrase detection and generation(Pavlick et al., 2015). However, there are known limitations to fixed leaderboards, and some work has proposed evolving evaluation sets to accompany model improvements (Ma et al., 2021; Khashabi et al., 2021; Kasai et al., 2021). With NND evaluation, we propose to repurpose the annotations of model-generated texts, both enabling to learn from prior model's errors, as well as adapt to more recent model performance.

7 Discussion

7.1 Other Domains

Although we focus on three NLG tasks, annotations from human evaluation in other NLG tasks could be used to expand the framework further in future work, for example with the WMT MQM (Freitag et al., 2021) annotations for translation, the SAMSA (Sulem et al., 2018b) annotations for text simplification, or the HLGD for news headline generation (Laban et al., 2021a).

7.2 Benefits of NND

Flexibility of Framework. NND relies on pre-existing human annotations to generate NND test pairs. However, the required annotation format is flexible, our experiments show that NND is compatible with single-error categorizations (e.g., the Quiz Design in Section 3.1), hierarchical categorizations (e.g., FRANK in Section 3.3), or Likert-scale ratings (e.g., SummEval in Section 3.3). NND results adopt the shape of the repurposed human evaluation, for instance, results in Section 5.2 are broken down both by general summarization aspects using the SummEval NND, and further refined to detailed categories with the FRANK NND.

Direct Language Model Evaluation. In a typical NLG evaluation, a decoding strategy is used to generate a candidate which is evaluated. Often, authors of a model recommend a decoding strategy to pair with the model, which creates an additional confounding factor in the evaluation, as a better decoding strategy (e.g., Nucleus Sampling Holtzman et al. (2019)) can lead to improvements regardless of model quality. NND avoids this problem by evaluating a model directly through its likelihood and by-passing the use of a decoding strategy.

Computationally Inexpensive. Computing candidate likelihood requires a single model forward pass, through teacher forcing, whereas other automated NLG evaluations often require full candidate generation, which is computationally expensive. The low computational cost of NND enables rapid evaluation during training (Section 5.4).

Limitations of NND are discussed in Section 9.

8 Conclusion

We introduce the Near-Negative Distinction (NND) framework for the evaluation of NLG models. In the NND framework, a pre-existing human evaluation dataset is repurposed to create NND test pairs comprised of text candidates of differing quality. Models are evaluated on their ability to assign a higher likelihood to high-quality candidates, giving an estimate of whether models would avoid the errors of previously evaluated models. We apply the NND framework to three NLG tasks: question generation, question answering, and summarization, and show that NND results correlate better with human preference than prior NLG evaluation methods. The NND framework allows the breaking down of model performance by error category, and we illustrate how the framework's flexibility can be used to understand model strengths and weaknesses, for instance extrapolating how newer models would perform in an existing human study or how a summarization model can lose factual consistency ability during training. NND is a simple, automatic, and versatile evaluation method that we hope can accelerate NLG research.

9 Limitations

Reliance on Likelihood. Not all NLG models are language models capable of producing candidate likelihoods. For instance, black-box models such as GPT-3 (Brown et al., 2020) or an extractive summarizer (Mihalcea and Tarau, 2004) cannot be evaluated through NND out-of-the-box as there is no way to administer NND tests. Furthermore, NND relies on models being well-calibrated. If a model is poorly calibrated, it could generate a single quality candidate, but a poor judge of quality on other candidates, leading to low performance on NND tests. However, prior work has argued that model calibration is important: it enables models to generate diverse candidates and is important in gaining a user's trust in practical applications (Guo et al., 2017).

Reliance on Prior Errors. NND relies on annotated errors of previous models to evaluate a new model, which assumes errors made by models remain constant over time. This is limiting, as each

generation of models has specific strengths and weaknesses, with new categories of errors emerging over time. We recommend that NND be used as a temporary extension to a human evaluation, allowing for a few generations of models to be evaluated on the same benchmark. However, the gold standard of NLG evaluation remains human evaluation, and it should still be performed frequently, and repurposed into updated NND test sets.

NND Requirements. Not all human annotations of generated texts can be repurposed for NND evaluation, and the two requirements – outlined in §2.1) – limit usability of the evaluation methodology. More precisely, annotations can be repurposed only if several model outputs are labeled for a given input, and if a partial ordering of quality over the labels is known. We however show in the paper that these requirements are common amongst existing annotation collections.

Sensitivity to Normalization. A complication of the NND framework is that it relies on inputting the prior model's outputs into the evaluated model to obtain a likelihood. NLG models use different norms for punctuation and capitalization, making the exchange of generated text across models delicate. Other NLG evaluation metrics are also sensitive to un-normalized texts (Post, 2018), and for NND it falls on the creator of the dataset to verify that NND test pairs are well-framed and do not contain noise that might affect result validity.

10 Ethical Considerations

We focused our experiments on models and datasets for the English language, and even though we expect the NND framework to be adaptable to other languages and settings, we have not verified this assumption experimentally and limit our claims to the English language.

The models and datasets utilized primarily reflect the culture of the English-speaking populace. Gender, age, race, and other socio-economic biases may exist in the dataset, and models trained on these datasets may propagate these biases. Question-answering and summarization tasks in particular have previously been shown to contain these biases.

We selected question generation, question answering, and summarization as the three NLG domains on which we assessed the NND framework. We expect that the framework will be beneficial in other NLG tasks such as data-to-text, image captioning, or simplification, but have not created NND test sets for these domains and limit our claims to the three tasks we ran experiments for.

We note that NND datasets are not novel datasets. Still, transformations of pre-existing human annotation datasets and proper permission to reuse underlying datasets should be granted before usage in the NND framework. Our experiments all relied on publicly released human evaluation annotations with explicit permission for research re-use.

Acknowledgement

We thank Alexander Fabbri, Jesse Vig, Greg Durrett, Jiacheng Xu and Shafiq Joty for helpful feedback on the manuscript.

References

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

BIG bench collaboration. 2021. Beyond the imitation game: Measuring and extrapolating the capabilities of language models.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Shuyang Cao and Lu Wang. 2021. Cliff: Contrastive learning for improving faithfulness and factuality in abstractive summarization. In *EMNLP*.

Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. 2013. One billion word benchmark for measuring progress in statistical language modeling. *arXiv preprint arXiv:1312.3005*.

Sumit Chopra, Raia Hadsell, and Yann LeCun. 2005. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 539–546. IEEE.

Alexander R Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. 2021. Summeval: Re-evaluating summarization evaluation. *Transactions of the Association for Computational Linguistics*, 9:391–409.

Markus Freitag, George Foster, David Grangier, Viresh Ratnakar, Qijun Tan, and Wolfgang Macherey. 2021. Experts, errors, and context: A large-scale study of human evaluation for machine translation.

Sebastian Gehrmann, Tosin Adewumi, Karmanya Aggarwal, Pawan Sasanka Ammanamanchi, Anuoluwapo Aremu, Antoine Bosselut, Khyathi Raghavi Chandu, Miruna-Adriana Clinciu, Dipanjan Das, Kaustubh Dhole, et al. 2021. The gem benchmark: Natural language generation, its evaluation and metrics. In *Proceedings of the 1st Workshop on Natural Language Generation, Evaluation, and Metrics (GEM 2021)*, pages 96–120. Association for Computational Linguistics.

Tanya Goyal, Jiacheng Xu, Junyi Jessy Li, and Greg Durrett. 2021. Training dynamics for text summarization models. *arXiv preprint arXiv:2110.08370*.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. 2017. On calibration of modern neural networks. In *International Conference on Machine Learning*, pages 1321–1330. PMLR.

Tatsunori B Hashimoto, Hugh Zhang, and Percy Liang. 2019. Unifying human and statistical evaluation for natural language generation. *arXiv preprint arXiv:1904.02792*.

Tianxing He and James Glass. 2020. Negative training for neural dialogue response generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2044–2058.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Pierre Isabelle, Colin Cherry, and George Foster. 2017. A challenge set approach to evaluating machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2486–2496.

Fred Jelinek, Robert L Mercer, Lalit R Bahl, and James K Baker. 1977. Perplexity—a measure of the difficulty of speech recognition tasks. *The Journal of the Acoustical Society of America*, 62(S1):S63–S63.

Jungo Kasai, Keisuke Sakaguchi, Ronan Le Bras, Lavinia Dunagan, Jacob Morrison, Alexander R Fabbri, Yejin Choi, and Noah A Smith. 2021. Bidimensional leaderboards: Generate and evaluate language hand in hand. *arXiv preprint arXiv:2112.04139*.

M. G Kendall. 1938. A new measure of rank correlation. In *Biometrika*.

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2019. Generalization through memorization: Nearest neighbor language models. In *International Conference on Learning Representations*.

Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. 2020. Unifiedqa: Crossing format boundaries with a single qa system. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1896–1907.

Daniel Khashabi, Gabriel Stanovsky, Jonathan Bragg, Nicholas Lourie, Jungo Kasai, Yejin Choi, Noah A Smith, and Daniel S Weld. 2021. Genie: A leaderboard for human-in-the-loop evaluation of text generation. *arXiv preprint arXiv:2101.06561*.

Bernard Koch, Emily Denton, Alex Hanna, and Jacob G Foster. 2021. Reduced, reused and recycled: The life of a dataset in machine learning research. *arXiv preprint arXiv:2112.01716*.

Laura Koesten, Pavlos Vougiouklis, Elena Simperl, and Paul Groth. 2020. Dataset reuse: Toward translating principles to practice. *Patterns*, 1(8):100136.

Philippe Laban, Lucas Bandarkar, and Marti A Hearst. 2021a. News headline grouping as a challenging nlu task. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3186–3198.

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2021b. Keep it simple: Unsupervised simplification of multi-paragraph text. In *ACL/IJCNLP*.

Philippe Laban, Tobias Schnabel, Paul N Bennett, and Marti A Hearst. 2021c. Summac: Re-visiting nli-based models for inconsistency detection in summarization. *arXiv preprint arXiv:2111.09525*.

Philippe Laban, Chien-Sheng Wu, Lidiya Murakhovs'ka, Wenhao Liu, and Caiming Xiong. 2022. Quiz design task: Helping teachers create quizzes with automated question generation. In *Findings of the North American Chapter of the Association for Computational Linguistics: NAACL 2022*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

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

Dayiheng Liu, Yu Yan, Yeyun Gong, Weizhen Qi, Hang Zhang, Jian Jiao, Weizhu Chen, Jie Fu, Linjun Shou, Ming Gong, et al. 2021. Glge: A new general language generation evaluation benchmark. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 408–420.

Yixin Liu and Pengfei Liu. 2021. Simcls: A simple framework for contrastive learning of abstractive summarization. *arXiv preprint arXiv:2106.01890*.

Zhiyi Ma, Kawin Ethayarajh, Tristan Thrush, Somya Jain, Ledell Wu, Robin Jia, Christopher Potts, Adina Williams, and Douwe Kiela. 2021. Dynaboard: An evaluation-as-a-service platform for holistic next-generation benchmarking. *Advances in Neural Information Processing Systems*, 34:10351–10367.

Mary Ann Marcinkiewicz. 1994. Building a large annotated corpus of english: The penn treebank. *Using Large Corpora*, 273.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 404–411.

Lidiya Murakhovs'ka, Chien-Sheng Wu, Philippe Laban, Tong Niu, Wenhao Liu, and Caiming Xiong. 2022. Mixqg: Neural question generation with mixed answer types. *Findings of the North American Chapter of the Association for Computational Linguistics: NAACL 2022*.

Aakanksha Naik, Abhilasha Ravichander, Norman Sadeh, Carolyn Rose, and Graham Neubig. 2018. Stress test evaluation for natural language inference. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2340–2353.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Caglar Gulcehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290.

Nathan Ng, Kyra Yee, Alexei Baevski, Myle Ott, Michael Auli, and Sergey Edunov. 2019. Facebook fair's wmt19 news translation task submission. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 314–319.

Artidoro Pagnoni, Vidhisha Balachandran, and Yulia Tsvetkov. 2021. Understanding factuality in abstractive summarization with frank: A benchmark for factuality metrics. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4812–4829.

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.

Ellie Pavlick, Pushpendre Rastogi, Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2015. Ppdb 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 425–430.

Matt Post. 2018. A call for clarity in reporting bleu scores. *WMT 2018*, page 186.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog 1.8 (2019): 9*.

Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2017. Self-critical sequence training for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7008–7024.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5418–5426.

Julian Salazar, Davis Liang, Toan Q Nguyen, and Katrin Kirchhoff. 2020. Masked language model scoring. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2699–2712.

Thomas Scialom, Paul-Alexis Dray, Sylvain Lamprier, Benjamin Piwowarski, Jacopo Staiano, Alex Wang, and Patrick Gallinari. 2021. Questeval: Summarization asks for fact-based evaluation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6594–6604.

Elior Sulem, Omri Abend, and Ari Rappoport. 2018a. BLEU is not suitable for the evaluation of text simplification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 738–744, Brussels, Belgium. Association for Computational Linguistics.

Elior Sulem, Omri Abend, and Ari Rappoport. 2018b. Semantic structural evaluation for text simplification. 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 685–696.

Oyvind Tafjord and Peter Clark. 2021. General-purpose question-answering with macaw. *arXiv preprint arXiv:2109.02593*.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Simas Sakenis, Jason Huang, Yaron Singer, and Stuart Shieber. 2020. Causal mediation analysis for interpreting neural nlp: The case of gender bias. *arXiv preprint arXiv:2004.12265*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. *arXiv preprint arXiv:2106.11520*.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Appendix

A NND Examples

We provide example NND tests from each of the datasets used in experimentation, with question generation examples in Table A1, generative QA in Table A2, summarization in Table A3. The elements were hand-picked to illustrate a diversity of cases and error categories present in the NND test sets.

Selected NND Tests - Question Generation

Like all catalysts, enzymes **increase the reaction rate by lowering its activation energy**. Some enzymes can make their conversion [...]

What do enzymes do?	What does enzyme do?
`No Error`	`Disfluent`

Californium [...] The element was named after the **university and the U.S. state of California**. Two crystalline forms exist for californium [...]

What is Californium named after?	What is the state of California?
`No Error`	`Off Target`

The Palazzo Pitti [...] Giorgio Vasari proposed that Brunelleschi was the palazzo's architect, and that his pupil **Luca Fancelli** was merely his assistant in the task, but today it is Fancelli who is generally credited.

Who is generally credited with the design of the Palazzo Pitti?	Who was the pupil of Brunelleschi?
`No Error`	`Wrong Context`

Table A1: **Three selected examples from the NND QGen dataset.** For a given context with the target answer in bold, two candidates are provided: No Error (left) and Error (right).

Selected NND Tests - Generative Question Answering

Common Sense

If plastic was a conductor, then would a plastic spoon conduct electricity?

Yes, but it would be a very poor conductor.	No. Plastic is a non-conductor of electricity.
`Credit: 1`	`Credit: 0`

Comparison

What is the difference between a noun and a verb?

a verb expresses action, a noun describes things	A noun is a person, place, or thing. A verb is a person, place, or thing.
`Credit: 1`	`Credit: 0`

Entity

Imagine an empty cup. Now put a coin in the cup. Now put another coin in the cup. Now put a pen in the cup. How many coins are in the cup now?

2 coins and a pen.	three
`Credit: 1`	`Credit: 0`

Creativity

How can you sit and stand at the same time?

you can't	It's easy. You just sit down and stand up.
`Credit: 1`	`Credit: 0`

Science

Why does the sky reflect blue light?

The sky is blue because of Rayleigh scattering.	Because God hates you.
`Credit: 1`	`Credit: 0`

Table A2: **Three selected examples from the NND QGen dataset.** For a given context with the target answer in bold, two candidates are provided: No Error (left) and Error (right).

Selected NND Tests - Summarization

Uber has poached Facebook's security chief Joe Sullivan in an attempt to double down on rapidly escalating safety concerns. The $40 billion taxi service has been plagued by serious accusations of failing to vet its drivers. Lawsuits have been brought against Uber in San Francisco and Los Angeles. A New Delhi driver was accused of raping a passenger in December. This week in Denver, a driver tried and failed to break into a passenger's home. And in London, [...]

Facebook's security chief Joe Sullivan will leave his role as Facebook's security chief to help Uber defend safety concerns. Lawsuits have been brought against Uber in San Francisco and Los Angeles. There were three high-profile assault cases involving Uber drivers in December 2014.	The $40 billion taxi service has been plagued by serious accusations. The $40 billion taxi service has been plagued by serious accusations. It comes days after a driver tried and failed to break into a passenger's home.
`No Error`	`Coreference Error`

One of the biggest TV events of all time is being reimagined for new audiences. "Roots," the epic miniseries about an African-American slave and his descendants, had a staggering audience of over 100 million viewers back in 1977. Now A&E networks are remaking the miniseries, to air in 2016. A&E, Lifetime and History (formerly the History Channel) announced Thursday that the three networks would simulcast a remake of the saga [...]

A&E, lifetime and history will simulcast a new "roots" in 2016. The original miniseries drew more than 100 million viewers in 1977. Levar Burton, who played Kunta Kinte in the original, will co-executive produce.	"Roots," the epic miniseries about an african-american slave and his descendants , had a staggering audience of over 100 million viewers back in 1977. Now A&E, lifetime and history (formerly the history channel) announced Thursday. Producers will consult scholars in african and african-american history for added authenticity.
`No Error`	`Incoherent`

Table A3: **Two selected examples from the NND Summarization datasets.** For a given document, two candidates are provided: No Error (left) and Error (right). The top example is from the FRANK NND, and the bottom from the SummEval NND.